A Deep Dive into Node-level Analysis with Fusion RNN Model for Smart LTE Network Monitoring

by

Md Rashidul Islam
20366008

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
M.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
September 2023

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Md Rashidul Islam
20366008

Approval

The thesis/project titled “Deep Dive into node-level analysis with Machine Learning Models for Smart LTE Network Management” submitted by

1. Md Rashidul Islam (20366008)

Of Summer, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of M.Sc. in Computer Science on September 12, 2023.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Examiner:
(External)

Prof. Mohammad Zahidur Rahman, Ph.D.
Professor
Department of Computer Science and Engineering
Jahangirnagar University, Savar, Dhaka

Examiner:
(Internal)

Dr. Amitabha Chakrabarty
Professor
Department of Computer Science and Engineering
Brac University

Examiner:
(Internal)

Dr. Md. Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Amitabha Chakrabarty
Professor
Department of Computer Science and Engineering
Brac University

Chairperson:
(Chair)

Sadia Hamid Kazi, Ph.D
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Predicting and understanding traffic patterns have become important objectives for maintaining the Quality of Service (QoS) standard in network management. This change stems from analyzing the data usage on cellular internet networks. Cellular network optimiser frequently employ a variety of data traffic prediction algorithms for this reason. Traditional traffic projections are often made at the high-level or generously large regional cluster level and therefore has the lacking in precised forecation. Furthermore, it is difficult to obtain information on eNodeB-level utilisation with regard to traffic predictions. As a result, using the conventional approach causes user experience degradation or unnecessary network expansion. Developing a traffic forecasting model with the aid of multivariate feature inputs and deep learning techniques was one of the objective of this research. It deals with extensive 6.2 million real network time series LTE data traffic and other associated characteristics, including eNodeB-wise PRB utilisation. A cutting-edge fusion model based on Deep Learning algorithms is suggested. Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), and Gated Recurrent Unit (GRU) are three deep learning algorithms that when combined allow for eNodeB-level traffic forecasting and eNodeB-wise anticipated PRB utilisation.The proposed fusion model's R^2 score is 0.8034, outperforms the conventional state-if-the-art models. This study also proposed a unique method that thoroughly examines individual nodes for the Smart Network Monitor. This approach follows adjustments made to soft capacity parameters at the eNodeB level, aiming for immediate improvement or long-term network growth to meet a consistent QoS standard. The algorithm relies on expected PRB utilization.

Keywords: LTE Networks, Machine Learning in Networking, Traffic Prediction, Deep Learning, Mobile Network Capacity, Physical Resource Block, Resource Management

Acknowledgement

First and foremost, glory be to the Great Allah, with whose help me and able to finish writing our thesis without too many setbacks.

Second, I want to thank my mentor, Md. Golam Rabiul Alam, for his kind assistance and counsel with my work. He was there for me anytime I needed him.

Thirdly, IEEE Access 2023 and the judging panel where the paper ”Deep Learning Based Fusion Model for Multivariate LTE Traffic Forecasting and Optimized Radio Parameter Estimation” got accepted. DOI: 10.1109/ACCESS.2023.3242861

And ultimately, without my parents’ ongoing support, it might not be possible. With their kind support and prayer I am now on the verge of my graduation.

Table of Contents

Declaration i

Approval ii

Abstract iv

Acknowledgment v

Table of Contents vi

List of Figures viii

List of Tables ix

Nomenclature ix

1 Introduction 1

1.1 Motivation . . . 1

1.2 Research Problem . . . 2

1.3 Research Methodology . . . 4

1.3.1 Research Approach . . . 4

1.3.2 Data Collection . . . 4

1.3.3 Machine Learning Model Development . . . 4

1.3.4 Ethical Considerations . . . 5

1.3.5 Scope and Limitation . . . 5

1.4 Research Contributions . . . 6

1.5 Thesis Organization . . . 6

2 Theoretical Background 7

2.1 LTE Air Interface . . . 7

2.2 LTE QoS Parameter Understanding . . . 10

2.3 Knowledge Discovery and EDA . . . 11

2.4 Deep learning algorithm in time series forecasting . . . 12

2.4.1 Long Short-Term Memory (LSTM) . . . 13

2.4.2 Bidirectional LSTMs (BiLSTM) . . . 14

2.4.3 Gated Recurrent Unit (GRU) . . . 15

2.4.4 Deep Regression . . . 17

3 System Model 18
3.1 Problem Formulation 18
3.2 System Model 20
3.3 Datasets Collection 21
3.4 Exploration of Patterns and Correlation in Datasets 23
3.5 Multivariate Deep Learning Algorithms 26
3.5.1 Architecture for Multivariate Time Series Prediction 26
3.6 Introduction of Fusion Model 28
3.7 Architecture and Strategy of Fusion Model with RNN 28
3.8 Comprehensive RNN and Deep Regression 29

4 Experimental Results 30
4.1 Environment setup and Sequential model design 30
4.2 Evaluation and Performance Metrics 32
4.2.1 Regression Metrics 32
4.3 Experimental Model Outcome 33
4.4 Performance of the Fusion Model 33
4.5 Discussion of Performance Evaluation 34

5 Smart Network Management 35
5.1 LTE QoS plaining and Over utilization solution 35
5.2 Capacity Monitoring after Planning and Deployment 35
5.3 Problem Statement in Legacy Capacity Management 36
5.4 Traditional Network Management 36
5.5 Smart LTE Node monitoring 39

6 Conclusion 41
6.1 Conclusion 41
6.2 Future Research 41

Bibliography 45

List of Figures

2.1 LTE Architecture and interfaces 8
2.2 Physical Resource Block the LTE air interface 9
2.3 LSTM architecture for predicting future traffic 14
2.4 BiLSTM architecture 15
2.5 GRU architecture for predicting future traffic 16
2.6 Single layer regression with deep neural network 17

3.1 Proposed System Model of Cellular Network Traffic Prediction, PRB Utilization and eNodeB Capacity Based Smart Network management. 20
3.2 Data structure of a sample Node 23
3.3 Month by Month Average traffic (GB) per eNodeB 24
3.4 Boxplot of Month-by-Month Average traffic (GB) 24
3.5 Weekday vs. Weekend hourly traffic Pattern 25
3.6 Daily Average traffic (GB) 25
3.7 Correlation Plot from different features 26
3.8 Architecture of the proposed multivariate Deep Neural Network for multiple parallel time-series prediction 27
3.9 Fusion Model with blended approach 29

4.1 Sequential model for LSTM, BiLSTM and GRU 31
4.2 Training vs Validation loss of a sample eNodeB 31
4.3 Regression plots of the models at the training phase 33
4.4 Regression plots of the Fusion Model 34
4.5 Actual vs. Predicted Traffic based on different Model 34

5.1 Characteristics of two LTE 10Mhz BW Nodes 37
5.2 Correlation Heatmap of LTE KPI 38
5.3 Network Busy Hours Samples eNodesB 38
5.4 Capacity Utilization vs PRB Utilization from NBH Sample 39
5.5 Outliers detection and remove for one sample eNodeB 39
5.6 Capacity Utilization vs PRB Utilization with Deep-Dive Approach . . 40

List of Tables

4.1 Performance of the model in the testing phase 33
4.2 Performance of the Fusion Model 34

Chapter 1

Introduction

The usage of mobile internet data traffic has seen a significant increase in recent decades. According to the Ericsson Mobility Report released in November 2021, it is predicted that by 2027, global mobile network data traffic will approach 300 exabytes per month [36]. Although 5G and beyond technology are still under development, at present and over the next 2-3 years, LTE will carry most of the data traffic, especially in developing countries, as it is already established and covers 84% of the global population as of 2020, The report also highlighted the importance of sustainable business practices in the mobile industry and emphasized the need for more energy-efficient mobile networks, with the goal of reducing carbon emissions and achieving a net-zero carbon footprint by 2030. With the growth of mobile data traffic, new challenges are emerging, with the topmost being the eNodeB-wise utilization prediction, as the quality of network services, such as speed and latency, relies on this parameter. The average monthly internet usage per smartphone is approximately 11.4 GB as of the Report 2021, which is expected to increase almost four times by 2027. Video traffic currently accounts for nearly 70% of all mobile data traffic and is predicted to reach around 79% by 2027. With the variation in user behavior based on location, the LTE network or cell may experience additional loads. Therefore, it is imperative to predict network utilization accurately in advance to ensure that Mobile Network Operators (MNO) can handle the user demand on time and avoid QoS degradation. The report also highlighted that QoS requirements varied depending on the application and user behavior.

For example, real-time applications, such as online gaming and video conferencing, required low latency and high reliability. In contrast, applications such as web browsing and social media could tolerate higher latency but required fast download speeds. So, QoS would continue to be a critical factor for mobile network operators to differentiate themselves in a highly competitive market and provide superior user experience.

1.1 Motivation

The most complex aspect of network dimensioning is traffic forecasting, according to [23]. Investors are constantly seeking for the optimal Capital Expenditure (CAPEX) in the right cell/site/location and minimizing Operational Expenditure (OPEX) to increase the profitability of the cellular network company. Inaccurate traffic predictions might lead to incorrect network sizing, which would increase CAPEX and OPEX costs and degrade QoS.

In recent times, deep learning-based approaches have gained attention for their ability to identify patterns in sequential data and group similar data types together [18]. Researchers have explored various algorithms based on Recurrent Neural Networks (RNN) to forecast multiple types of sequential time series data. Recognizing the immense potential of deep learning algorithms in predicting future trends, the authors of the study were particularly interested in leveraging this technology to tackle one of the most critical challenges in cellular network dimensioning: traffic forecasting [22]. To achieve this, modern graphics processing units (GPUs) were utilized to efficiently execute complex deep learning algorithms with a diverse range of features.

The significance of accurate traffic and user demand information lies in the network's ability to effectively manage resource allocation among connected users, ultimately resulting in an improved quality of user experience [22]. This study aims to enhance our understanding of traffic behavior in mobile networks and provide valuable recommendations for triggering the expansion of eNodeBs. By leveraging deep learning algorithm-based traffic forecasts and utilization correlation charts, network operators can make informed decisions regarding capacity expansion and ensure optimal resource allocation in response to changing user demands.

In summary, deep learning has the unleashed potential to play a crucial role in empowering mobile network planning. By leveraging deep learning algorithms for traffic forecasting, resource allocation, and network optimization, MNOs can overcome the challenges of maintaining quality of service, efficient resource utilization, and network scalability. The utilization of deep learning techniques not only enhances network performance and user experience but also enables a more democratized and inclusive approach to mobile network planning.

1.2 Research Problem

Forecasting traffic in cellular networks is crucial for anticipating network conditions, understanding user usage patterns, and estimating important parameters related to quality-of-service and resource allocation [30]. Previous research by Fang, Ergüt, and Patras [37] employed a cell handover-aware graph neural network for city-scale traffic forecasting. Xu, Lin, Huang, *et al.* [13] analyzed time series data to create geographical distribution maps of forecasted traffic heatmaps for a specific city. Similarly, Kirmaz et al. from Nokia Bell Lab [26] conducted research on traffic prediction by dividing the geographic area into pixels.
While these works [37] [13] [26] focused on predicting traffic at a geographical level, our research concentrates on predicting traffic at each eNodeB or cell level. Trinh, Giupponi, and Dini [16] and Sun, Wang, Zhao, *et al.* [39] explored mobile traffic forecasting and network-level mobile data estimation, respectively, at different time scales. However, our developed model specifically predicts traffic at an hourly level, offering more insights into time series data and easy conversion to daily level predictions [16].

Lo Schiavo, Fiore, Gramaglia, *et al.* [38] developed a hybrid approach using Thresholded Exponential Smoothing and Recurrent Neural Network (TES-RNN) to manage traffic anomalies at a particular time. Yu, Wang, Li, *et al.* [40] employed Graph Attention Network (GAT) and Temporal Convolutional Network (TCN) to predict traffic overload with large amounts of small-scale redundant data. In contrast, our focus lies on cellular network traffic forecasting at a granular cell level, treating each cell as a separate eNodeB. This approach allows easy conversion of cell-level forecasts into city or province-level predictions by aggregating all eNodeB traffic within the geographic area. Furthermore, our research differentiates itself by addressing two major factors: hourly granularity in time and eNodeB-level or geographical granularity. This emphasis on granular data traffic prediction aligns with the real-life challenges faced in network planning. By considering these factors, we aim to develop an hourly traffic forecast model that meets the specific needs of network planning.

In summary, the distinctive features of our research compared to other similar works are the granularity of the traffic prediction in terms of both time (hourly) and network or geography (eNodeB level). These factors contribute to the unique contributions of our research in the field of cellular network traffic forecasting.

The central objective of this research was to concentrate on network PRB Utilization using forecasted traffic data. Consequently, we have provided a sequential examination of the latest network traffic forecasting methods and the approach for estimating PRB Utilization. While there are some prior works related to cellular network traffic forecasting that employ different techniques, the Deep-Dive based smart network Management from forecasted traffic has received limited attention from researchers. In this section.

The second phase of our research focuses on predicting future network utilization using forecasted traffic and introduces an algorithm to handle anticipated traffic by estimating utilization. While some scattered research exists on radio capacity analysis at various times, Jang, Lee, Kwon, *et al.* [31] developed a model to estimate the resource block usage rate, addressing the fixed-length input problem in traditional RNN models. However, this research [31] does not address how Fusion model can be utilized to estimate RB usage rate (RBUR). Similarly, Hasan, Kwon, and Na [15] proposed deep-dive approach for adaptive network management to maintain throughput in LTE Small-Cell networks, but it primarily follows a reactive approach without defining proactive measures [15]. Importantly, none of these research works devise a Fusion model and smart network monitoring based on predicted future traffic.

In contrast, our research tackles this specific issue and proposes an algorithm and method that triggers actual high capacity utilized site based on forecasted traffic. This approach aims to proactively optimize and monitor in response to predicted traffic conditions.

1.3 Research Methodology

This section refers to the systematic approach and techniques used to conduct research. Also encompasses the overall strategy and procedures employed to acquire, organize, analyze, and interpret data in order to answer research questions or investigate a specific problem.

1.3.1 Research Approach

The research design specifies the overall approach that was used in the study as a quantitative research approach. The research relies on empirical evidence and aims to be as objective as possible by minimizing biases and subjective interpretations.

1.3.2 Data Collection

The collected LTE 4G datasets from the Operations Support System (OSS) of one of the MNO were preprocessed to ensure data quality and relevance. This study relies on authentic network-generated data and the MNO's configuration as the fundamental pillars for its investigation. Here also steps involved handling missing values, data normalization, feature engineering, and outlier detection. The preprocessing steps were carefully applied to avoid any bias in the subsequent analysis.

1.3.3 Machine Learning Model Development

Algorithm Selection

In the subsequent stage of the research, the focus shifted towards the selection of appropriate deep learning models for mobile network planning tasks. Thorough investigation and experimentation were carried out to determine the most suitable models. Given the intricacy and characteristics of the data, three widely recognized deep learning architectures were ultimately chosen: Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), and Gated Recurrent Unit (GRU). These models have shown remarkable proficiency in capturing temporal dependencies and have been successfully employed in diverse time series prediction tasks.

Model Training and Evaluation

After the evaluation of the deep learning models, they were put into action and trained using the gathered data. The implementation was carried out utilizing a widely used time series forecasting algorithm, leveraging its rich functionalities and optimization capabilities. The training process entailed feeding the models with historical data and iteratively refining the model parameters to minimize prediction errors. Particular emphasis was placed on hyperparameter tuning and regularization techniques to guarantee the attainment of optimal model performance. The evaluation of the deep learning models' performance involved the use of suitable assessment metrics, including but not limited to Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE) and squared correlation (R2). The chosen evaluation metrics were carefully selected to offer a comprehensive insight into the models' predictive accuracy and their capability to capture underlying patterns within the mobile network data. Throughout the research, ethical considerations were prioritized. Any potential biases in the data or models were addressed, and the limitations of the research were openly acknowledged.

1.3.4 Ethical Considerations

Ethical considerations were given utmost importance during the entire research process. The data utilized in this study were anonymized and managed in adherence to privacy regulations and best practices. The research was conducted with great respect for ethical guidelines, safeguarding individuals' privacy and confidentiality. This chapter has offered an outline of the research methodology utilized in this study. The subsequent chapter will present the findings and analysis obtained from applying deep learning models to tackle mobile network management challenges.

1.3.5 Scope and Limitation

Scope

The primary aim of this research is to showcase the revolutionary capabilities of deep learning algorithms in the field of network management. The study centers around a particular top mobile network operator (MNO) with an extensive subscriber base exceeding 50 million users nationwide. Real network-generated data and the MNO's configuration are fundamental to this research. Numerous machine learning fusion models have been developed to cater to the network management objectives, tailored to suit the unique characteristics and needs of the selected MNO. Through the utilization of available data, these proposed models seek to optimize network performance and improve the overall user experience.

Limitation

The suggested model does have some limitations as well. Under normal day-to-day circumstances, the model performs well. However, its performance may experience a slight decline during social gatherings or events where the number of users in a particular location significantly exceeds the typical load. Additionally, during the development of the algorithm for soft parameter tuning, our primary focus was on enhancing LTE capacity. As a result of this dynamic resource sharing technique from GSM to LTE, GSM networks might occasionally encounter resource constraints. This challenge can be addressed through optimization techniques. Furthermore, there is a need to improve computational power to accommodate the increasing number of eNodeBs. In the upcoming chapter, we will provide the outcomes and detailed examination resulting from the implementation of the developed machine learning models to tackle the network planning challenges encountered by the selected MNO.

1.4 Research Contributions

This research delves into the untapped potential of intelligent mobile network management, aiming to contribute valuable insights to the academic and practical realms. Our primary objective is to propose a methodology that leverages deep learning algorithms to predict key network components and to address prevalent challenges in mobile network operations through in-depth node-level analysis.
The major contributions of this research paper are as follows:

- **Advanced Fusion Model:** We introduce a sophisticated fusion model that amalgamates deep learning algorithms, including LSTM, BiLSTM, and GRU, to enhance the precision and efficacy of network traffic forecasting. This approach factors in multiple variables to model and predict data traffic within mobile networks. Our specific focus lies in predicting the utilization or cell load at the eNodeB level, based on forecasted traffic. This prediction holds immense significance for Mobile Network Operators (MNOs), empowering them to make well-informed decisions about network expansion and ensuring the maintenance of top-tier service quality.
- **Node-Level Analysis Innovation:** We present an innovative technique involving a deep dive into node-level analysis to overcome capacity management challenges. Leveraging the XGBoost Regression method, we identify consistent patterns of high resource utilization across nodes while filtering out outliers and sporadic spikes. This approach emphasizes the significance of monitoring each node individually to uphold QoS benchmarks effectively.
- **End-to-End System Architecture:** We propose a comprehensive end-to-end system architecture for intelligent mobile network maintenance. This architecture not only contributes to cost reduction in terms of Capital Expenditure (CapEx) and Operational Expenditure (OpEx) but also ensures the continual adherence to QoS benchmarks.

1.5 Thesis Organization

Chapter 2 will delve into the theoretical foundation related to this study, with a particular emphasis on the knowledge discovery process, machine learning techniques, machine translation, and technical documentation. Subsequently, in Chapter 3, the chosen methodology will be outlined. Chapter 4 will present the empirical data and the corresponding results obtained from the experiments. Moving forward, Chapter 5 conduct a critical identification of the approaches and methods employed, and assess the validity and reliability of Smart Network Management. Finally, Chapter 6 will offer a comprehensive summary of the work and provide insights into potential future research and expansions on this subject.

Chapter 2

Theoretical Background

This section provides an overview of the relevant theoretical foundations pertinent to the experiment's framework. It aims to address the following questions:

- How LTE Architecture work and elements of LTE Air interface.
- Importance of LTE QoS Parameter.
- What are the advantages of Exploratory Data Analysis (EDA) and which algorithms are promising in the field of time series forecasting?

To answer these questions, the first part of this section delves into the knowledge discovery process in databases. This is followed by an exploration of Exploratory Data Analysis and a comprehensive discussion on deep learning principles and relevant algorithms. Subsequently, the section presents the current state-of-the-art fusion models and their quality evaluation. Finally, the section concludes by characterizing traffic patterns and examining the distinct features associated with them.

2.1 LTE Air Interface

LTE Architecture: Long-Term Evolution or LTE, is a wireless high-speed data communication standard for mobile devices and data terminals. It is a technology that cellular networks use to provide high-speed internet access to portable devices like smartphones, tablets, laptops, and other connected devices. LTE is designed to offer significantly faster data speeds, lower latency, and better overall performance than previous generations of cellular technology. Its foundations are the GSM/EDGE and UMTS/HSPA network technologies, with modifications to increase capacity and speed through the use of a different radio interface and a more straightforward core network.

The components of the LTE network: A standard LTE system architecture consists of an E-UTRAN (Evolved UMTS Terrestrial Radio Access Network) and main component is the Evolved Packet Core, also known as an EPC.
Below is the components:

- **User Equipment (UE):** The user equipment refers to the mobile devices or data terminals used by individuals. These devices include smartphones, tablets, laptops, and other wireless devices capable of connecting to LTE networks.

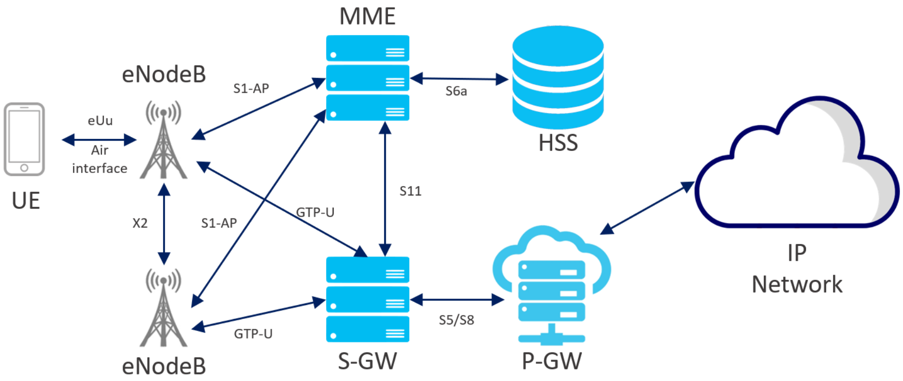


Figure 2.1: LTE Architecture and interfaces

- **Evolved NodeB (eNodeB):** The eNodeB, often referred to simply as "base station" or "cellular tower," is a key component of the LTE network. It connects directly to the UE and provides the wireless air interface. It's responsible for transmitting and receiving data to and from the user equipment. Each eNodeB covers a certain geographical area known as a cell.

- **E-UTRAN (Evolved UMTS Terrestrial Radio Access Network):** E-UTRAN consists of multiple eNodeBs and the interfaces that connect them. It handles the radio access part of the network and ensures that the data is transmitted efficiently between the user equipment and the core network.

- **Evolved Packet Core (EPC):** The EPC is the core network component of LTE architecture. It is responsible for handling tasks related to mobility, security, and various services. The EPC is composed of several key components:

  - **Mobility Management Entity (MME):** The MME is responsible for tracking the location of the user equipment, handling authentication, and managing handovers as the UE moves between cells.
  - **Serving Gateway (S-GW):** The SGW routes and forwards user data packets between the eNodeB and the packet data network (PDN). It also manages user-plane mobility within the E-UTRAN.
  - **PDN Gateway (P-GW):** The PGW connects the LTE network to external packet data networks (such as the internet) and performs tasks like IP address allocation, policy enforcement, and packet filtering.
  - **Home Subscriber Server (HSS):** The HSS stores subscriber information, including user profiles and authentication information. It's used for authentication, authorization, and mobility management.
  - **Policy and Charging Rules Function (PCRF)** The PCRF manages policy enforcement and quality of service (QoS) rules. It helps determine how different types of traffic are treated on the network and how they are charged.

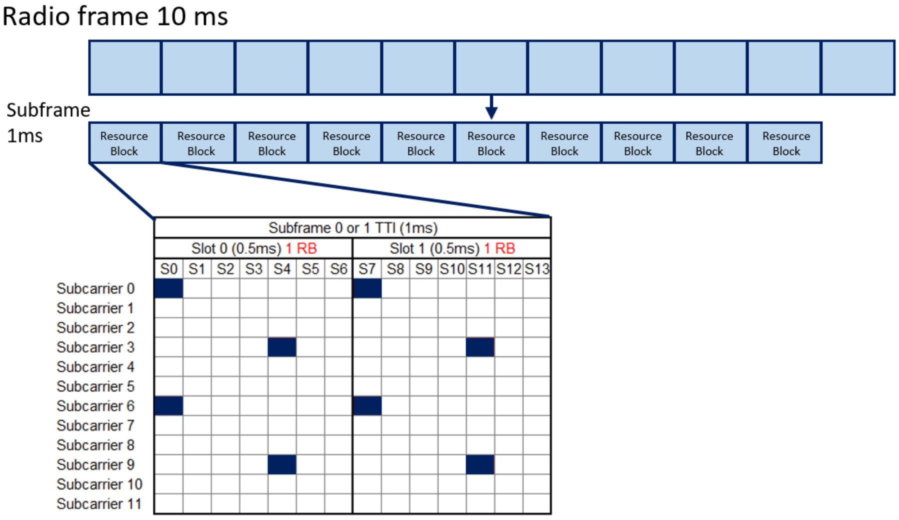


Figure 2.2: Physical Resource Block the LTE air interface

LTE Air Interface: In LTE, the wireless channel used for data transmission and reception between user equipment (UE) and the LTE base station (eNodeB) is referred to as the air interface. LTE divides the available frequency and temporal resources into smaller units known as Resource Blocks (RBs) and Physical Resource Blocks (PRBs) in order to effectively manage the scarce radio spectrum.

> **Resource Block** known as the The fundamental radio resource unit for LTE. Within a specified bandwidth, it represents a certain portion of the frequency spectrum. Each RB in the LTE standard has 12 subcarriers and covers a frequency range of 180 kHz [18], [12]. An RB's time duration varies depending on the type of specified subframe, but it normally corresponds to one slot, which lasts for 0.5 ms.
>
> > 1 RB = 12(Sub-carriers) x 7 (Symbols) = 84 Resource Elements. (For Normal CP: 7 symbols)
> >
> > 1 RB = 12(Sub-carriers) x 6 (Symbols) = 72 Resource Elements. (For Extended CP: 6 symbols)
>
> **Physical Resource Block (PRB)** are allotted collectively in A particular group of RBs that are in both the time and frequency domains. A PRB is, in other words, a two-dimensional unit for resource allocation. The PRB occupies a frequency range of 180 kHz and is made up of 12 sub-carriers that follow one another throughout a period of time (0.5 ms).

The e-NodeB assigns PRBs to particular UEs in the downlink (from the base station to the UE) so that they can transmit data. Similar to downlinks, PRBs are assigned to UEs for data transmission in uplinks (from the UE to the base station). The LTE scheduler manages the PRB distribution, dynamically allocating PRBs to UEs according to their quality of service specifications and channel conditions.

LTE can efficiently manage the distribution of radio resources and guarantee that various users receive enough bandwidth to satisfy their communication needs by dividing the available spectrum into PRBs. The system may modify the data rate dependent on the channel circumstances for each UE with the use of adaptive modulation and coding methods made possible by the usage of PRBs.

In a nutshell the LTE Resource Block and Physical Resource Block are essential components of the air interface because they allow for the efficient and adaptable use of radio resources to provide customers with high-speed data and a variety of services.

2.2 LTE QoS Parameter Understanding

Quality of Service (QoS) mechanisms are provided by LTE (Long-Term Evolution) networks to guarantee that various types of traffic receive adequate levels of service based on their unique requirements. Network operators may properly manage network resources and provide different services the attention they deserve based on their value and performance standards thanks to these QoS factors. Here are some significant LTE QoS factors and how to interpret them:

QoS Class Identifier (QCI) for a particular kind of traffic or service, the QCI is a numeric value (ranging from 1 to 9) that represents the priority level and the QoS features related to it. Services including telephony, video streaming, best-effort data, and others are given different QCIs. Packet latency, packet loss rate, and data rate are all determined by specified parameters for each QCI. The priority is higher and the QoS is better the higher the QCI value [18], [9].

Bit Rate is the highest data rate that a certain QoS flow is capable of. It is measured in bits per second (bps) and connected to several QCIs. For instance, compared to streaming high-definition video, speech services could have a lower bit rate demand.

Block Error Rate (BLER) is calculated as the ratio of incorrect blocks to all blocks. CRC is the method used to find errors in the transport block. The receiver will ask for HARQ NACK for re-transmission if the calculation does not produce the desired results. 90% effective transmission at the receiver end suggests that the standard BLER objective should be 10% in order to guarantee service quality [4]. If the BLER target is not met, further re-transmissions may be needed, increasing the radio resource consumption. To maintain the desired QoS benchmark of maximising throughput, ensuring user equality, and minimising Block Error Ratio, optimised and valuable resource scheduling strategies are necessary [11].

Resource Allocation based on QoS requirements, LTE dynamically distributes radio resources like PRBs (Physical Resource Blocks). Critical services are given the resources required for dependable performance when flows with higher priorities or QCI require more resources. The scheduler continuously modifies the PRB allocation as the network load and environmental factors change. When allocating resources, flows with higher priorities or stricter QoS specifications are given preference. This makes sure that vital services, like voice calls or real-time video streaming, get the resources they need to keep up their quality.

PRB Utilization is measured the efficiency with which the assigned PRBs are being used to transmit data, control signals, and other communication components within the network. It shows how effectively the radio spectrum is being used to meet the communication needs of user equipment (UE) and guarantee top network performance. Low PRB utilisation may imply that resources are underutilised or maybe wasted, whereas high PRB utilisation shows that a large amount of the given resources are being used. Several factors influence PRB utilization like *Traffic Load, Service Types, QoS Requirements, Channel Conditions, Scheduling Algorithms, Network Congestion.*

Capacity Utilization is how effectively the available resources within an LTE network are being used to handle user traffic and data demands. It's an important metric for ensuring a high-quality user experience and guiding network management decisions. Capacity utilization is not like PRB utilization and its depend on the *Spectral Efficiency (SE)*, SE is usually expressed in units of bits per second per hertz (bps/Hz). The higher the SE the more volume traffic can carry with in the limited spectrum.

2.3 Knowledge Discovery and EDA

Knowledge Discovery in the Dataset process, commonly referred to as KDD, is a systematic approach to extract valuable and previously unknown information from extensive datasets. It is a part of data mining, which is an essential step within the broader KDD process. The KDD process encompasses various stages, including data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation, and knowledge representation. This iterative and interactive process involves multiple iterations and feedback loops to uncover accurate and valuable insights from the data. The ultimate goal of the KDD process is to address real-world challenges by employing data analysis techniques and tools. Numerous applications of KDD exist, ranging from market basket analysis, network intrusion detection, customer segmentation, fraud detection, to recommendation systems

Exploratory Data Analysis (EDA) offers several advantages that make it an essential step in the data analysis process. Here are some of the key advantages of EDA:

1. Data Understanding: EDA helps researchers and analysts to get a better understanding of the dataset they are working with. It allows them to explore the structure, patterns, and relationships within the data, providing valuable insights into the nature of the variables and their distributions.

2. Data Cleaning and Preprocessing: During EDA, data inconsistencies, missing values, and outliers are identified. This helps in cleaning and preprocessing the data before performing any statistical modeling or machine learning tasks, ensuring that the subsequent analyses are based on accurate and reliable data.

3. Feature Selection: EDA assists in identifying the most relevant and informative features (variables) for the analysis. By understanding the relationship between variables and their impact on the target variable, researchers can make informed decisions about which features to include in their models.

4. Identification of Patterns and Trends: EDA helps in detecting patterns, trends, and anomalies in the data. It allows analysts to uncover potential insights and hidden relationships that may not be evident through summary statistics alone.

5. Hypothesis Generation: Exploratory data analysis often leads to the generation of hypotheses for further investigation. By observing patterns or trends, researchers can form initial hypotheses about relationships between variables, which can be formally tested in subsequent inferential analyses.

6. Visual Representation: EDA involves the use of various data visualization techniques such as scatter plots, histograms, box plots, and heatmaps. These visual representations make it easier to grasp the data's characteristics and convey complex information in an accessible manner.

7. Decision-Making Support: EDA provides a solid foundation for decision-making in various domains. Business leaders, policymakers, and researchers can use the insights gained from EDA to make informed decisions and take appropriate actions based on data evidence.

8. Efficient Resource Allocation: By understanding the distribution and characteristics of the data, organizations can allocate their resources effectively. For example, in marketing, EDA can help identify target customer segments, leading to optimized marketing strategies.

9. Early Detection of Data Anomalies: EDA allows the identification of data anomalies and errors at an early stage. Detecting and resolving these issues early on saves time, resources, and potential complications during later stages of the analysis.

10. Communication and Collaboration: EDA facilitates communication and collaboration among data analysts, domain experts, and stakeholders. Visualizations and insights from EDA can be shared and discussed, fostering a deeper understanding of the data and its implications.

Overall, Exploratory Data Analysis serves as a powerful and essential tool in the data analysis process, providing valuable insights, aiding in decision-making, and guiding subsequent steps in research and analysis. It helps analysts gain a comprehensive understanding of the data, paving the way for more accurate and meaningful results.

2.4 Deep learning algorithm in time series forecasting

According to Singh and Chauhan [5], Artificial Neural Networks (ANNs) are a mathematical model inspired by biological neural networks, emulating their functionality.

When compared to conventional algorithms, neural networks demonstrate the capability to handle significantly complex problems with relatively simpler algorithmic complexity. The primary advantage of using Artificial Neural Networks lies in their straightforward structure and self-organizing nature, enabling them to tackle a broad spectrum of problems without requiring additional intervention from the programmer. For instance, a neural network could be trained on customer behavior data in an online shop and predict whether a person is likely to make a purchase or not. An Artificial Neural Network is composed of nodes, also known as neurons, interconnected by weighted connections that can be adjusted during the network's learning process. Each node's output value is determined by an activation function based on its input values. Neural networks are organized into different layers: the input layer receives information from external sources, such as attribute values of the corresponding data entry, the output layer generates the network's final output, and hidden layers facilitate connections between the input and output layers. The input value of each node in every layer is computed by summing all incoming nodes' values multiplied by their respective interconnection weights [1]. Additionally, neural networks can be categorized into two primary types [5].

- Feedforward Networks encompass all networks that do not receive feedback from the network itself. In this type of network, input data flows in a unidirectional manner, starting from the input nodes, passing through 0 to n hidden nodes, and finally reaching the output nodes. The absence of feedback means that there is no information sent backward to readapt the system.

- Recurrent Networks comprise all networks that include a feedback mechanism, enabling them to reuse data from later stages during the learning process in earlier stages.

2.4.1 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture that addresses the limitations of traditional RNNs in capturing long-term dependencies in sequential data. Regular RNNs struggle with long sequences due to the vanishing or exploding gradient problem, which makes it challenging for them to retain information over long periods. LSTMs were introduced by Sepp Hochreiter and Jürgen Schmidhuber in 1997 to overcome this issue. They introduced additional components called "gates" within the LSTM units to control the flow of information. These gates, namely the input gate, forget gate, and output gate, work together to selectively allow information to be stored or discarded in the memory cell [2].

Some explanation of how LSTMs work:
Input Gate: It controls which information from the current input should be stored in the memory cell.
Forget Gate: It determines what information should be discarded from the memory cell from the previous time step.
Output Gate: It decides what information from the memory cell should be used as the output of the LSTM unit.

The ability to control information flow through these gates enables LSTMs to retain important information over longer sequences and efficiently capture long-term dependencies in the data. This makes LSTMs particularly well-suited for various sequential tasks such as natural language processing, speech recognition, time series analysis, and more.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{2.1}$$

Here h_{t-1} is the output value of the previous time, as well as x_t, denoting the input value of the present time. f_t is the output gate whose value range is (0,1). The weight of the forget gate is represented as W_f, where b_i is the bias of that forget gate.

In addition of that, input to input gate, output value and condition of candidate cell at input gate can also be calculated through output value of previous time and the input value of present time, which can be calculated through the below equations –

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{2.2}$$

$$\tilde{C}_t = \tanh\left(W_c \cdot [h_{t-1}, x_t] + b_c\right) \tag{2.3}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{2.4}$$

$$O_t = \sigma\left(W_0 \cdot [h_{t-1}, x_t] + b_o\right) \tag{2.5}$$

$$h_t = O_t * \tanh\left(C_t\right) \tag{2.6}$$

This LSTM is used as a sequential layer for building traffic forecasting model. This LSTM architecture is modified from [3], [35]. From the above equation (2.1),(2.2),(2.3) and (2.5) information transfer is based on dot product outcome. If the dot product result is zero, it means information is not transferred [32]. Information will transfer, in case of dot product outcome is one.

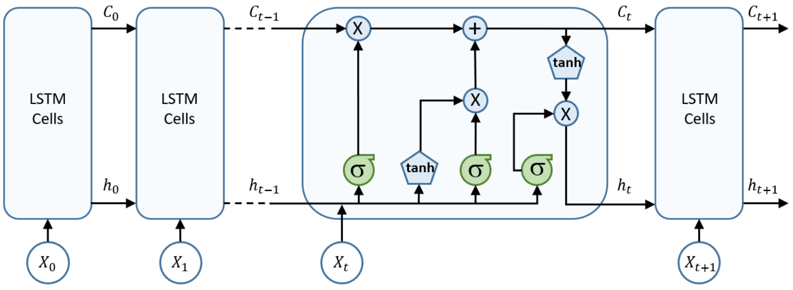


Figure 2.3: LSTM architecture for predicting future traffic

In these equations (2.3),(2.4) and (2.6), C_t is the cell state of the candidate cell in t time, which value ranges (0,1). O_t denotes the output gate, i_t is the input gate, and h_t is the hidden layers in the cell. Here, x_t is the cellular network data traffic. The bias of the network indicates by b function.

2.4.2 Bidirectional LSTMs (BiLSTM)

Bidirectional Long Short-Term Memory (BiLSTM) is an extension of the Long Short-Term Memory (LSTM) architecture. While traditional LSTMs process sequential data in a unidirectional manner, i.e., from past to future, BiLSTMs process the data in both directions simultaneously.

In a BiLSTM, the input sequence is fed into two separate LSTM layers: one layer processes the data from the beginning of the sequence to the end (forward direction), while the other layer processes the data from the end of the sequence to the beginning (backward direction). This bidirectional processing allows the BiLSTM to capture information from both past and future contexts, making it more capable of understanding the overall context and dependencies in the data.

The hidden states from both the forward and backward LSTM layers are typically concatenated to obtain the final output of the BiLSTM. This final output incorporates information from both directions, providing a more comprehensive representation of the input sequence.

BiLSTMs are particularly beneficial for tasks that require a deep understanding of the context, such as natural language processing tasks like named entity recognition, part-of-speech tagging, sentiment analysis, and machine translation. By capturing information from both past and future contexts, BiLSTMs can effectively model complex patterns and dependencies in the data, leading to improved performance in various sequential tasks.
In Fig 2.4 *li* represent the forward LSTM, *li'* represent the reverse directional LSTM, s_i and s_i' is the time series information delivering in LSTM cells

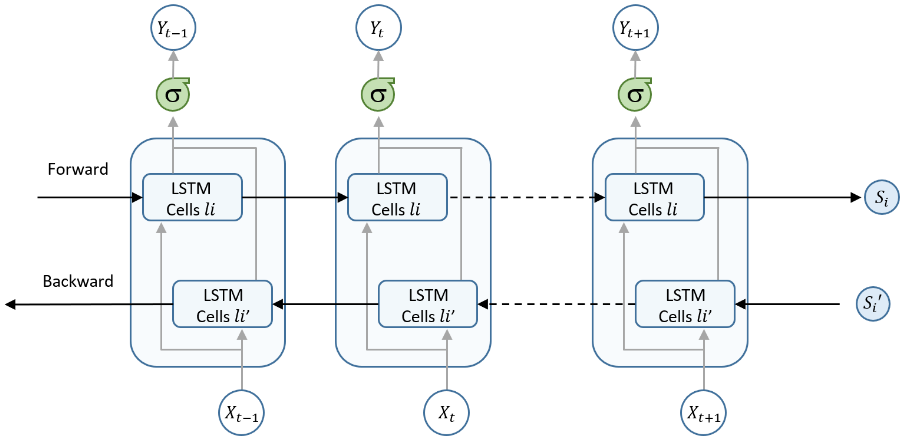


Figure 2.4: BiLSTM architecture

2.4.3 Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) is a variant of the traditional Long Short-Term Memory (LSTM) architecture and is used in recurrent neural networks (RNNs). Like LSTM, GRU is designed to address the vanishing gradient problem in traditional RNNs, allowing it to effectively capture long-term dependencies in sequential data. GRU was introduced by Kyunghyun Cho et al. in 2014 as a simplified version of LSTM. It achieves similar performance to LSTM but with fewer parameters, making it computationally more efficient and easier to train. The key components of a GRU unit are as follows:

Reset Gate (r): This gate determines which information from the previous time

step should be discarded or ”forgotten.” It controls how much of the past information is relevant for the current time step.
Update Gate (z): The update gate regulates the flow of new information into the current memory cell. It decides which parts of the new input should be considered for updating the memory.
Candidate Activation (h): The candidate activation computes the new candidate value that could potentially be added to the memory cell.
Hidden State (h): The hidden state at each time step is the output of the GRU unit and represents the information that is passed on to the next time step or to the output layer of the RNN.

In GRU, hidden state output at time t can be calculated as below general expression:

$$h_t = f\left(h_{t-1}, x_t\right) \tag{2.7}$$

In equation (2.7), h_{t-1} is the hidden state status in $t-1$ time and x_t input time series value at t time. For explaining to the GRU NN model as shown in architecture (Fig. 2.5) below equation can be used –

$$r_t = \sigma\left(W_r \cdot \left[h_{t-1}, x_t\right]\right) \tag{2.8}$$

$$z_t = \sigma\left(W_z \cdot \left[h_{t-1}, x_t\right]\right) \tag{2.9}$$

$$\tilde{h_t} = \tanh\left(W_{\tilde{h}}\left[r_t * h_{t-1}, x_t\right]\right) \tag{2.10}$$

$$h_t = \left(1 - z_t\right) * h_{t-1} + z_t * \tilde{h_t} \tag{2.11}$$

$$y_t = \sigma\left(W_o \cdot h_t\right) \tag{2.12}$$

In these equations (2.8),(2.9) and (2.12), Sigmoid function is represented as σ, which output is (0,1). r_t is the updated, which works for determining stored information quantity from one movement to another. Reset gate z_t determines the status of information of the last state, whether the information is kept or erased. The parameter which needs to train are denoted as W_r, W_z, W_h, W_o [22], [21], [19] & [23]. During the computation of the GRU, the reset and update gates are determined

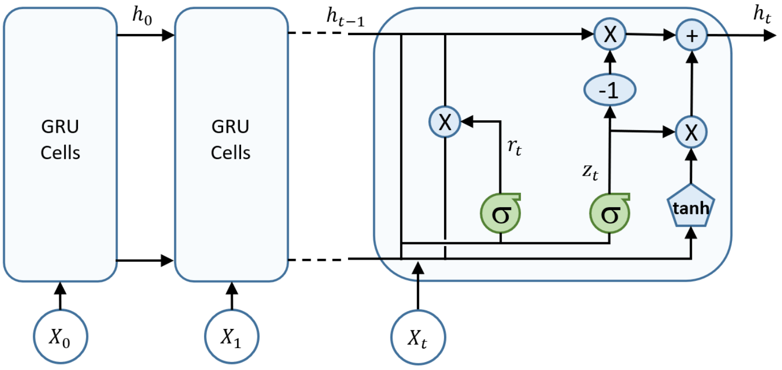


Figure 2.5: GRU architecture for predicting future traffic

by the input data and the hidden state from the previous time step. These gates allow the GRU to control the flow of information and effectively learn long-term dependencies in the sequential data.

2.4.4 Deep Regression

The essential challenge of predicting a continuous value based on input may be solved using the regression approach. [20], [33] After receiving the expected traffic based on the deep learning model, our study concentrated on utilisation prediction. In this situation, eNodeB-wise predicted traffic may be used to anticipate utilisation using deep regression (as indicated in the system model, Fig. 3.1) from the equation (2.13)

$$\hat{y} = w_1x_1 + w_2x_2 + \cdots + w_dx_d + b \tag{2.13}$$

Here, w is the weight of input traffic x_1 to x_d, and b is known as bias or offset. Weight determines the influence of features in the model. [20], [33] & [34].

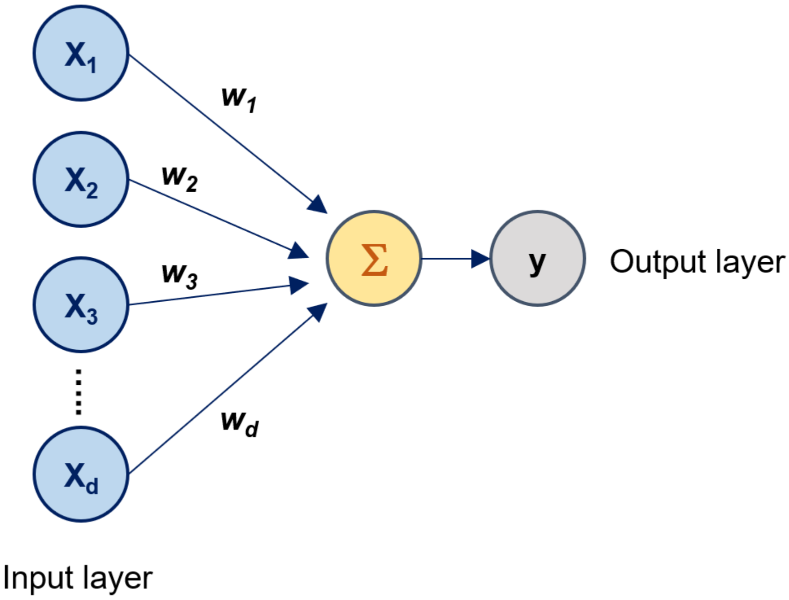


Figure 2.6: Single layer regression with deep neural network

Chapter 3

System Model

In this chapter, the suggested system model was presented with problem formulation, which tries to forecast mobile network traffic demand and optimise resource allocation. The system combines deep learning algorithms, Fusion Model framework, and data analytics. Finally, we go through the techniques used to analyse data and anticipate traffic and usage. Our technique provides proactive decision-making and effective network design to address the changing demands of cellular networks by utilising cutting-edge technologies.

3.1 Problem Formulation

The complexity of comprehending traffic demands within a cellular network arises from the vast and irregular distribution of mobile users connected to a specific network. Moreover, this undertaking becomes even more challenging due to the multitude of diverse devices and constantly changing user patterns. It is important to note that various applications exhibit varying rates of data traffic consumption [30].

From an academic research perspective, gathering a substantial amount of data to train a model poses a significant challenge. Mobile Network Operators (MNOs) do not provide eNodeB-specific detailed datasets with valuable features. Typically, the available Call Detail Records (CDR) present the traffic data in an aggregated format, lacking segregation based on technology, user count per technology, or per-protocol category [6]. As a result, the CDR dataset is not particularly helpful for this research endeavor. Various efforts and initiatives have been made to mine data from Operation Support Systems (OSS) and the radio and core network endpoints to collect a suitable dataset for analysis.

Network traffic forecasting is a crucial task in network dimensioning due to its direct impact on eNodeB-level utilization. Essentially, the traffic and utilization at the eNodeB level exhibit a direct correlation, influencing the overall network performance and user experience. If the utilization increases in an uncontrolled manner, leading to additional users sharing the same physical resource block (PRB), it will adversely affect the user experience since the initial design did not account for such high sharing. To prevent this, network expansion needs to be triggered before reaching the capacity threshold in such cases. On the other hand, if the traffic fails to increase as predicted, the eNodeBs will be underutilized, resulting in wasted resources.

The main challenge lies in the fact that network engineers typically become aware of these scenarios only after they have occurred in specific eNodeBs within a live network. Subsequently, engineers must take proactive measures to address the issue by implementing capacity solutions to increase or decrease resources, as required. However, during this lead time, customers may suffer from a degradation in the quality of service (QoS). It would be highly advantageous for network planners and engineers to identify eNodeB-level traffic patterns, PRB utilization, and estimate radio parameters before these issues manifest in the network. This would enable network engineers to initiate timely actions, mitigating the impact on customers and minimizing their suffering.

Consequently, we have devised a Non-deterministic Polynomial *(NP)* hard problem based above-sated situation to address the solution. *NP*-hard problems are commonly used in formalized research problems [25], [17]. This research question can be classified as an optimization problem as our objective is to find out the maximum user throughput (T_h)in a particular time for each eNodeB and which is inversely proportional with PRB Utilization ($PRBU_t$), and other network contains.

$$\text{Objective function, Max } T_h = \frac{1}{PRBU_t} + C_1 \tag{3.1}$$

Here in the objective function (3.1), the value of constant C_1 will change according to the configured radio bandwidth of each eNodeB. In other words, a user throughput of a particular eNodeB can vary based on configured bandwidth even in the same PRB utilization.

Similarly, future PRB utilization can be computed based on predicted traffic volume on that node and other factors. Suppose we want to calculate a cluster of eNobeB (number of eNodeBs in same geographic are creates cluster) future performance or PRB Utilization ($PRBU$). In that case, this can be possible with predicted traffic volume (Vol), and other factors i.e., Average user equipment $\left(\overline{UE}\right)$, Maximum user equipment $Max\,(UE)$, Downtime (D_T), and other unknown factors C_2. Thus, we can write the PRB Utilization equation as below for a cluster of eNodeBs:

$$median\,\{PRBU_T\} \times BW \lim_{T\to+\infty} \frac{1}{T} \sum_{t+1}^{t+60} \sum_{e\in E} \; (Vol_{T,e} + UE_{T,e} - D_{T,e}) + C_2 \tag{3.2}$$

In the above equation (3.2), we have considered only $\overline{UE}$ (User Equipment), because $Max\,(UE)$ varies in a certain geographic area or cluster only because of special circumstances and social events. Prediction of max UE for a particular eNodeB could be another research question we will address in our future work. However, we have considered average $\overline{UE}$ in the computation process, representing the number of connected user equipment in a particular eNodeB for a specific time frame. eNodeB-wise count of $\overline{UE}$ for a particular hour depends on cellular network operators' market share and population of that eNodeBs coverage area. So, in most cases, the $\overline{UE}$ will not change drastically for the yearly business plan (BP). In standard network conditions, there is minimum eNodeB downtime (D_T), where D_T negatively impacts traffic volume and PRB Utilization. So, it's easily understandable that future traffic is the most vital thing for predicting utilization as well as user throughput. If we can rightly predict the traffic or user throughput, then it's possible to take action to maintain the quality of service. In equation (3.2), $\sum_{e \in E} Vol_{T,e}$ is the summation of all eNodeB (E) traffic in a cluster.

By taking into consideration all of these actual network factors, we can simplify the PRB Utilization equation for one single eNodeB -

$$PRBU_{t \in T} \times BW = \lim_{T \to +\infty} \frac{1}{T} \left(Vol_{T,e} + \overline{UE}_{T,e} - D_{T,e} \right) + C_2 \tag{3.3}$$

As eNodeB-wise bandwidth (BW) and $\overline{UE}$ is almost constant for a particular network planning year, so it can assume that PRB utilization is directly proportional to traffic growth and bandwidth of a particular spectrum band. In equation (3.2) and (3.3), time $T = \{t+1, t+2, \cdots, t+60\}$, that means maximum 60 days hourly future PRB Utilization is denoted as $PRBU_{t+60}$, where Vol_{t+60} indicates predicted traffic volume (unit bits) in the same time frame.

The researchers have recognized the significant potential of deep learning algorithms in predicting time series data. Considering this, they have developed prediction models using a unique fusion strategy of deep learning algorithms. These models aim to forecast future traffic volume and physical resource block (PRB) utilization.

However, their research goes beyond solely predicting traffic and PRB utilization. They also assess the predicted PRB utilization based on the traffic data. Using this assessment, they have developed an algorithm for estimating radio network parameters. The purpose of this algorithm is to trigger appropriate actions to maintain the network's quality of service (QoS) benchmark. In other words, they aim to dynamically adjust the network parameters based on the predicted PRB utilization to ensure optimal QoS for users.

3.2 System Model

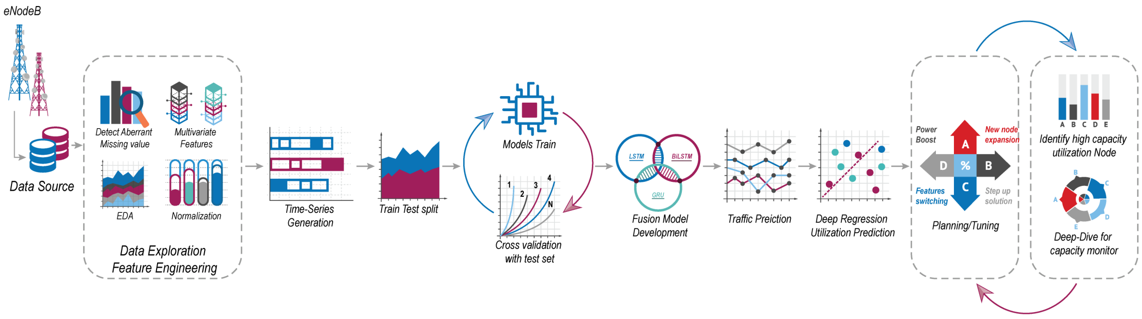


Figure 3.1: Proposed System Model of Cellular Network Traffic Prediction, PRB Utilization and eNodeB Capacity Based Smart Network management.

In this system model section, the cellular network traffic prediction system model is presented. In Fig. 3.1, we have proposed a cellular network traffic prediction model using deep learning algorithms.

Through a thorough data mining method, eNodeB-wise traffic and other associate parameters were gathered in the first step. Due to the complexity of LTE data, which includes a number of underlying attributes and information, it is kept in the local database after collection. The model forecasts traffic, which is the main goal of this study, thus after data storage, we completed the exploratory data analysis (EDA) portion. Exploratory Data Analysis (EDA), as we all know, is the process of locating significant characteristics and patterns in datasets. We have discovered any missing data in the datasets from EDA, which is explained in the section *Exploring Patterns and Correlation in Datasets.*

In second stage, basic model build-up done and based on train model Fusion strategy used in RNN model, detail architecture discussed in section *Architecture and Strategy of Fusion Model with RNN*, before the model build-up multivariate features included as train data to introduce events occurrence and forecast simile to realistic time-series traffic.

Based on predicted traffic, with the help of Deep Regression we forecast the PRB utilization and build an algorithm which help to Estimation QoS Parameter and decision direction to *Long-term Solutions involving new node growth or Step-up Solutions involving PRB load assessment* [41].
Finally, in operation phase a Deep-Dive approach taken to identify the abnormally behaved node for further network management which discussed in chapter 5.

3.3 Datasets Collection

We used telecommunications dataset that typically contains a collection of information related to various aspects of a telecommunications LTE network Operations Support System (OSS) of one of the MNO. This data can include a wide range of metrics and measurements that help analyze, optimize, and manage the network's performance and resources. The dataset spans 290 days of hourly data from 890 eNodeBs, yielding a total of 6.2M data points per metric. The dataset is divided into training and test sets, with a ratio of 79:21. Additionally, a 61-day validation dataset is isolated for final model evaluation. The dataset features six key metrics - traffic, user throughput, cell throughput, average user counts, max user counts, and PRB utilization.

This dataset contains all encrypted eNodeB-wise parameter information located in a densely populated city in South Asia. Let's assume the whole dataset as a $E_t = \{E_{c1t}, E_{c2t}, ...E_{cit}\}$, Where E_t is the sets of all eNodeB and E_{cit} is the all features of each individual eNodeB regardless of time (t). So, the aggregated eNodeB-wise traffic (Tr) in a time frame T is,

$$A(T) = \Sigma_{r(t) \in R(T)} E(t) \Sigma_{t \in T}\ a(t) \tag{3.4}$$

The Cellular Network dataset contains the following information and features used in this work:

- eNodeB: eNodeB is the Radio network element of the LTE network, which is also known as Evolved Node B.

- Traffic: Traffic means a combination of Uplink (UL) and Downlink (DL) internet Traffic from the Radio network end. The counter formula of traffic is as below:
$$\sum \text{downlink traffic volume for } PDCP \\ + \sum \text{uplink traffic volume for } PDCP$$
The unit of traffic is Gigabits here.

- Utilization: Utilization indicates the usage of Physical Resource Block (PRB) in LTE system. The higher number of utilization indicates more usage of LTE resources. Utilization can be formulated in counter level by the below formula:
$$\frac{AvgnumberofusedPRBs}{NumberofavailablePRBs}$$

- Max_UE: Maximum number of Users connected at an instance in a particular node considered as Max_UE.

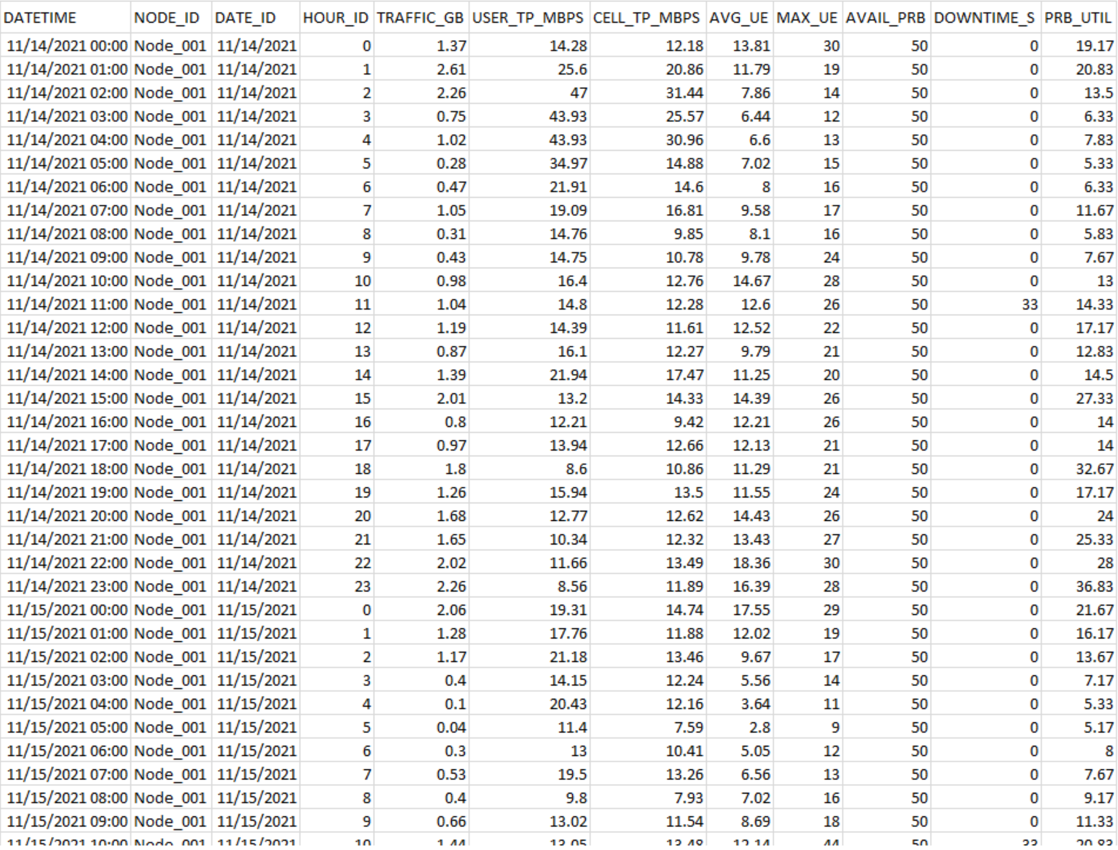
- Avg_UE: Avg_UE is the average number of connected Users per hour in a particular node

- Cell_TP: Cell_TP means Cell Throughput, which is the sum of all users' throughput in a particular eNodeB or any node for a unit time frame. The counter level formula can be represented as below:
$$\frac{\sum \text{downlink traffic volume for PDCP}}{\sum \text{duration of downlink data transmission in a Node}}$$

- User_TP: A particular user receives an amount of data on average, known as User Throughput or User_TP. In other words, the average number of packets received by the User in a unit time frame. The counter level formula for User_TP as below –
$$\frac{(\sum \text{DL traffic} - \text{DL traffic volume sent in last TTI})}{\text{Data transmit duration except last TTI}}$$

During data modeling of traffic forecasting for utilization prediction, all eNodeB has been classified in different classes according to their time-series behavior. One sample Node Data structure shown in fig 3.2 where 7 major *KPIs* are illustrated with hourly values.

With these 7 *KPIs* like Traffic_GB, PRB Utilization, User_TP, Cell_TP, Max_UE, Avg_UE and Downtime a NodeB can be summarized with respect to usages, utilization, QoS and service. There are some other *KQIs* (key quality indicator) in LTE like CQI Index but it only define the data quality index, but selected 7 *KPIs* indicates over-all performance and have correlation with PRB/Capacity Utilization which shown in fig 3.7.



DATETIME	NODE_ID	DATE_ID	HOUR_ID	TRAFFIC_GB	USER_TP_MBPS	CELL_TP_MBPS	AVG_UE	MAX_UE	AVAIL_PRB	DOWNTIME_S	PRB_UTIL
11/14/2021 00:00	Node_001	11/14/2021	0	1.37	14.28	12.18	13.81	30	50	0	19.17
11/14/2021 01:00	Node_001	11/14/2021	1	2.61	25.6	20.86	11.79	19	50	0	20.83
11/14/2021 02:00	Node_001	11/14/2021	2	2.26	47	31.44	7.86	14	50	0	13.5
11/14/2021 03:00	Node_001	11/14/2021	3	0.75	43.93	25.57	6.44	12	50	0	6.33
11/14/2021 04:00	Node_001	11/14/2021	4	1.02	43.93	30.96	6.6	13	50	0	7.83
11/14/2021 05:00	Node_001	11/14/2021	5	0.28	34.97	14.88	7.02	15	50	0	5.33
11/14/2021 06:00	Node_001	11/14/2021	6	0.47	21.91	14.6	8	16	50	0	6.33
11/14/2021 07:00	Node_001	11/14/2021	7	1.05	19.09	16.81	9.58	17	50	0	11.67
11/14/2021 08:00	Node_001	11/14/2021	8	0.31	14.76	9.85	8.1	16	50	0	5.83
11/14/2021 09:00	Node_001	11/14/2021	9	0.43	14.75	10.78	9.78	24	50	0	7.67
11/14/2021 10:00	Node_001	11/14/2021	10	0.98	16.4	12.76	14.67	28	50	0	13
11/14/2021 11:00	Node_001	11/14/2021	11	1.04	14.8	12.28	12.6	26	50	33	14.33
11/14/2021 12:00	Node_001	11/14/2021	12	1.19	14.39	11.61	12.52	22	50	0	17.17
11/14/2021 13:00	Node_001	11/14/2021	13	0.87	16.1	12.27	9.79	21	50	0	12.83
11/14/2021 14:00	Node_001	11/14/2021	14	1.39	21.94	17.47	11.25	20	50	0	14.5
11/14/2021 15:00	Node_001	11/14/2021	15	2.01	13.2	14.33	14.39	26	50	0	27.33
11/14/2021 16:00	Node_001	11/14/2021	16	0.8	12.21	9.42	12.21	26	50	0	14
11/14/2021 17:00	Node_001	11/14/2021	17	0.97	13.94	12.66	12.13	21	50	0	14
11/14/2021 18:00	Node_001	11/14/2021	18	1.8	8.6	10.86	11.29	21	50	0	32.67
11/14/2021 19:00	Node_001	11/14/2021	19	1.26	15.94	13.5	11.55	24	50	0	17.17
11/14/2021 20:00	Node_001	11/14/2021	20	1.68	12.77	12.62	14.43	26	50	0	24
11/14/2021 21:00	Node_001	11/14/2021	21	1.65	10.34	12.32	13.43	27	50	0	25.33
11/14/2021 22:00	Node_001	11/14/2021	22	2.02	11.66	13.49	18.36	30	50	0	28
11/14/2021 23:00	Node_001	11/14/2021	23	2.26	8.56	11.89	16.39	28	50	0	36.83
11/15/2021 00:00	Node_001	11/15/2021	0	2.06	19.31	14.74	17.55	29	50	0	21.67
11/15/2021 01:00	Node_001	11/15/2021	1	1.28	17.76	11.88	12.02	19	50	0	16.17
11/15/2021 02:00	Node_001	11/15/2021	2	1.17	21.18	13.46	9.67	17	50	0	13.67
11/15/2021 03:00	Node_001	11/15/2021	3	0.4	14.15	12.24	5.56	14	50	0	7.17
11/15/2021 04:00	Node_001	11/15/2021	4	0.1	20.43	12.16	3.64	11	50	0	5.33
11/15/2021 05:00	Node_001	11/15/2021	5	0.04	11.4	7.59	2.8	9	50	0	5.17
11/15/2021 06:00	Node_001	11/15/2021	6	0.3	13	10.41	5.05	12	50	0	8
11/15/2021 07:00	Node_001	11/15/2021	7	0.53	19.5	13.26	6.56	13	50	0	7.67
11/15/2021 08:00	Node_001	11/15/2021	8	0.4	9.8	7.93	7.02	16	50	0	9.17
11/15/2021 09:00	Node_001	11/15/2021	9	0.66	13.02	11.54	8.69	18	50	0	11.33
11/15/2021 10:00	Node_001	11/15/2021	10	1.44	13.05	13.48	12.14	44	50	33	20.83

Figure 3.2: Data structure of a sample Node

3.4 Exploration of Patterns and Correlation in Datasets

We first extracted critical features of the dataset through feature engineering. Essential feature means which information has highly correlated with traffic data. As we considered multivariate inputs for the traffic prediction model, those inputs have different units. For this reason, data normalization is necessary to avoid systematic bias. We have used the min-max method in this work to transform all multivariate inputs from zero to one. Scaling input data helps reduce biasness as well as increase the accuracy of the traffic forecasting model. Equation (3.5) is used for data transformation:

$$z_n = \frac{x - x_{\min}}{x_{\max - x_{\min}}} \left(New_{\max_x} - New_{\min_x}\right) + New_{\min_x} \tag{3.5}$$

Maximum data denoted as $x_{\max}$, and $x_{\min}$ is the minimum of the data. $New_{\min}$ and $New_{\max}$ is the zero and one respectively [29]. After Normalization and transformation, we divided data into two parts: test and train. In this research, we have split the training and test data ratio as 79:21.

We try to comprehend the dataset before trying to identify patterns because it provides some insight into how traffic fluctuates over time and the crucial hours that contribute to overall data traffic [41].

$$E(t) = \sum_{i=1}^{351} (\text{Traffic in hours})\,/\text{Number of Days} \tag{3.6}$$

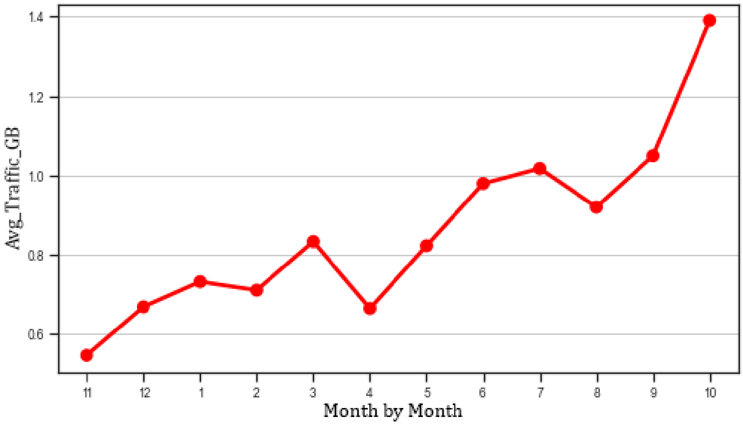


Figure 3.3: Month by Month Average traffic (GB) per eNodeB

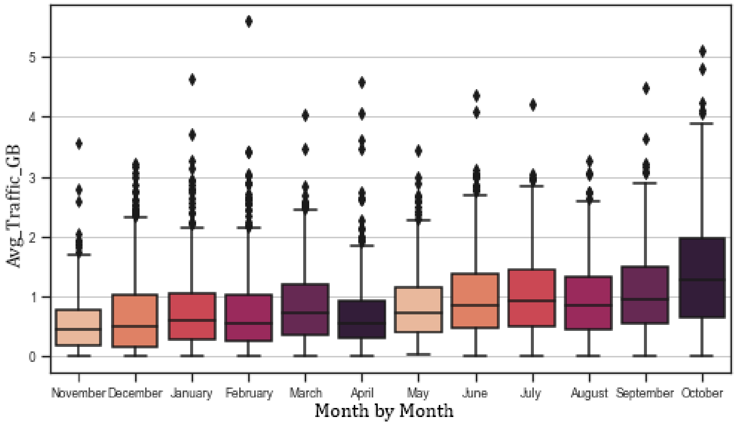


Figure 3.4: Boxplot of Month-by-Month Average traffic (GB)

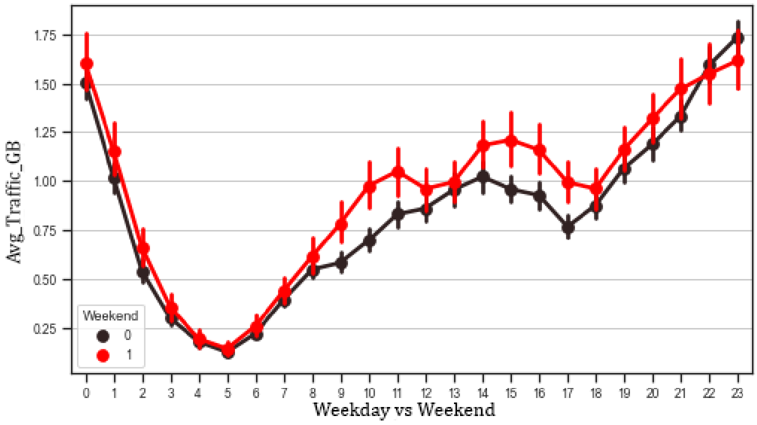


Figure 3.5: Weekday vs. Weekend hourly traffic Pattern

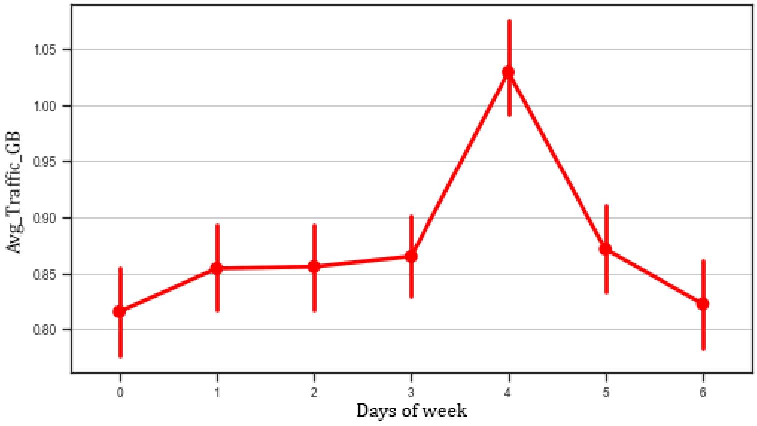


Figure 3.6: Daily Average traffic (GB)

From the above four figures, we can quickly identify the pattern of the hourly traffic dataset of 351 days. Below equation (3.6) used to identify data patterns per eNodeB. From Fig. 3.3, it can be easily understood that traffic is increasing over the period. We have also noticed the hourly traffic difference between weekdays and weekends in Fig. 3.5. The Monthly average traffic box plot is showing the median (Q2) traffic is increasing in every month Fig. 3.4. In Fig. 3.6, it represents the one special day in a week when traffic is almost double the rest of the days. Five features that are crucial components for forecasting cellular network traffic and comprehending the whole LTE network for sizing have been gathered in this study to anticipate future eNodeB-wise traffic. For the Fig. 3.7 correlation (r) plot, we used the below equation (3.7) for each pair –

$$r = \frac{n\left(\sum xy\right) - \left(\sum x\right)\left(\sum y\right)}{\sqrt{\left[n\sum x^2 - \left(\sum x\right)^2\right]\left[n\sum y^2 - \left(\sum y\right)^2\right]}} \tag{3.7}$$

The correlation diagram in graphic form According to Fig. 3.7, utilization and traffic are directly associated, but User_TP (User Throughput) and both traffic and utilization are adversely correlated. This suggests that increased traffic will result in increased usage, decreased user throughput, or a reduction in the quality of services (QoS). From a network design standpoint, we want to maintain usage at its highest

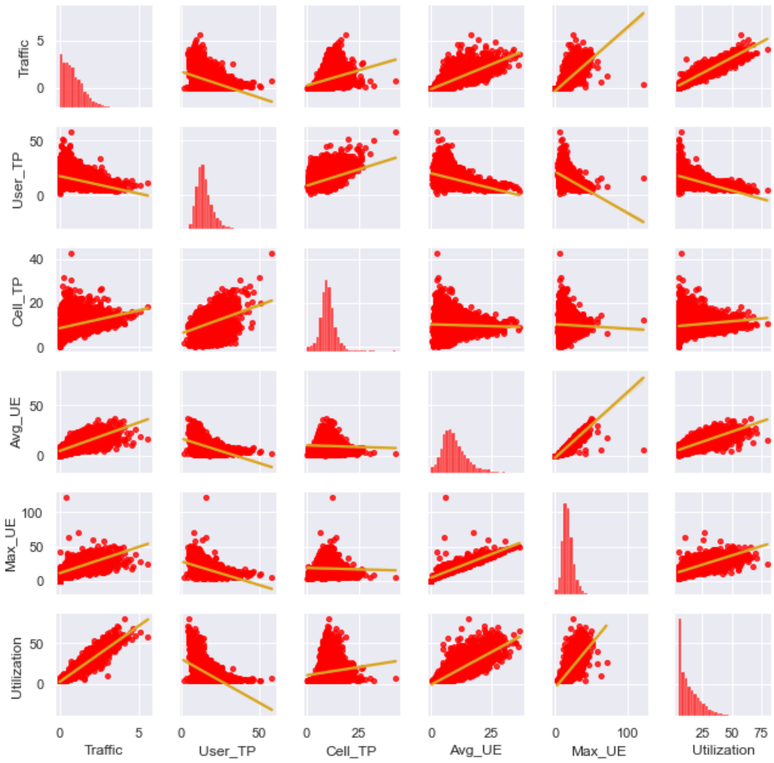


Figure 3.7: Correlation Plot from different features

level. If use rises, the eNodeB will experience unacceptable traffic, necessitating the need for network expansion. MNOs can maintain standard usage and QoS in this way [41].

3.5 Multivariate Deep Learning Algorithms

With the aid of three cutting-edge deep learning algorithms (LSTM, BiLSTM, and GRU), we proposed a state-of-the-art fusion model for predicting network traffic. We have taken into account multivariate inputs for modeling and forecasting Mobile network data traffic, in contrast to the majority of previous studies. Based on anticipated traffic, we estimated eNodeB-level utilization (or cell load). One of the important results of this research is the deep learning model-based traffic forecasting technique's prediction of eNodeB-level utilization, which will assist MNOs in deciding whether to expand their networks in order to maintain benchmark QoS [41].

3.5.1 Architecture for Multivariate Time Series Prediction

Actual cellular network traffic has other relationships besides those with the prior data trend. The amount of data traffic on a specific eNodeB might vary depending on a number of variables. For instance, if an eNodeB is unavailable for a longer period of time than usual, traffic may drop significantly. The reason for increased traffic is also any social or religious event that draws more people to congregate beneath one or more eNodeBs in a specific area. Therefore, we explored Multivariate input-based time-series traffic prediction in light of those actual network dimensioning issues [41].

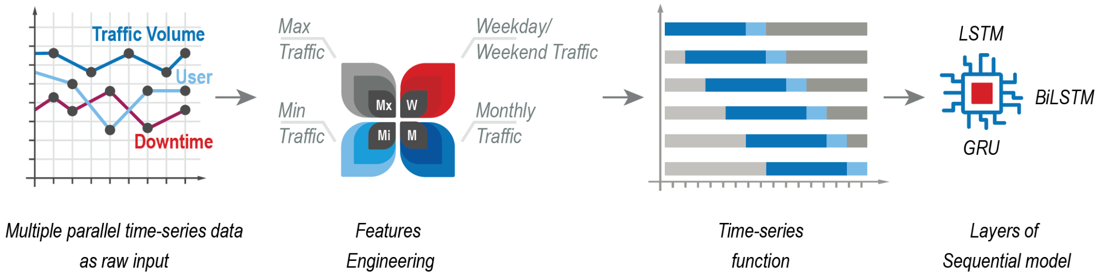


Figure 3.8: Architecture of the proposed multivariate Deep Neural Network for multiple parallel time-series prediction

If we consider both the factors as affecting elements for traffic forecasting. In other words, denoting the related variables by $\mathrm{x}_{1,t}, \mathrm{x}_{2,t}, \cdots \mathrm{x}_{k,t}$ and at the end of t time traffic $\mathrm{T}_{1,t}$ can be represent as equation (3.8)

$$\mathrm{T}_{1,t} = f1\left(\mathrm{x}_{1,t}, \mathrm{x}_{2,t}, \cdots, \mathrm{x}_{k,t}, \mathrm{x}_{1,t-1}, \mathrm{x}_{2,t-1}, \cdots, \mathrm{x}_{k,t-1} \cdots\right) \tag{3.8}$$

After forecasting traffic $\mathrm{T}_{1,t}$, *next t+1* time traffic will be dependent on all previous stage variables. Considering this logic equation (3.8) can be written as below for t+k time predicted traffic $\mathrm{T}_{k,t+k}$:

$$\mathrm{T}_{k,t+k} = \mathrm{f}_k\left(\mathrm{x}_{1,t}, \mathrm{x}_{2,t}, \cdots, \mathrm{x}_{k,t}, \mathrm{x}_{1,t-1}, \mathrm{x}_{2,t-1}, \cdots, \mathrm{x}_{k,t-1} \cdots\right) \tag{3.9}$$

As per the working principle of multivariate time series analysis, where different variables are dependent on their previous value as well as other feature or variables [35]. Like univariate time series, the major objective of multivariate time series prediction is to get the data forecast. But multivariate function enables more accurate results with the help of other associate parameters, which we include in this research work. As represented in the deep neural network model architecture in Fig.3.8, after collecting raw time-series input data, key features are extracted from dataset. Later on, we generated time series from this feature extracted data by using the sliding window technique algorithm. The sliding window technique works on N-1 historical time series data [14]. The working principle is after feature extraction, *ts function* generate the time series data (as shown in Fig.3.8) and explained by algorithm 1.

Algorithm 1: Time-Series generation with sliding window technique

```
Data:
A: array of traffic and feature[1]
p: number of days in past as sliding window
f: number of total features
Result: return array of X and target Y
initialization;
$x, y \leftarrow 0$;
for $i \leftarrow p$ to $length(A)$ do
    $append(A[i - p : i, 0 : f])toX$
    $append(A[i : i + 1, 0])toY$
end
return $X, Y$
```

3.6 Introduction of Fusion Model

In many fields, including finance, weather forecasting, and sales forecasting, time series forecasting is an essential responsibility. In time series data, complex patterns and connections are frequently difficult to detect using conventional methods. Due to their capacity to recognize long-term dependencies, deep learning architectures like LSTM, BiLSTM, and GRU have demonstrated promising outcomes. Each architecture, however, has advantages and disadvantages. The objective of this study is to develop an ensemble fusion model by utilizing the complementary nature of these architectures. And in this paper, we propose a fusion model for time series forecasting by leveraging the strengths of Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), and Gated Recurrent Unit (GRU) architectures. The fusion approach aims to improve the accuracy and robustness of predictions by combining the unique features of each individual model. We present the design, training process, and evaluation results of the fusion model compared to standalone LSTM, BiLSTM, and GRU models.

3.7 Architecture and Strategy of Fusion Model with RNN

Three primary parts make up the fusion model: LSTM, BiLSTM, and GRU. Separate temporal patterns and characteristics are extracted by each component after individually processing the input time series data to predict for future traffic volume and PRB utilization. The final forecast is then produced by combining the results of these components using an ensemble technique, such as *Blending.* A validation dataset is kept separate and used to make predictions rather than the entire dataset being used to train the basic models. Models are trained individually on each dataset, and their predictive performance is evaluated using various accuracy metrics. Based on the evaluation results, each dataset is assigned the model that demonstrated the best accuracy. A systematic comparison of the models' performance showcases their relative strengths.The analysis reveals that the performance

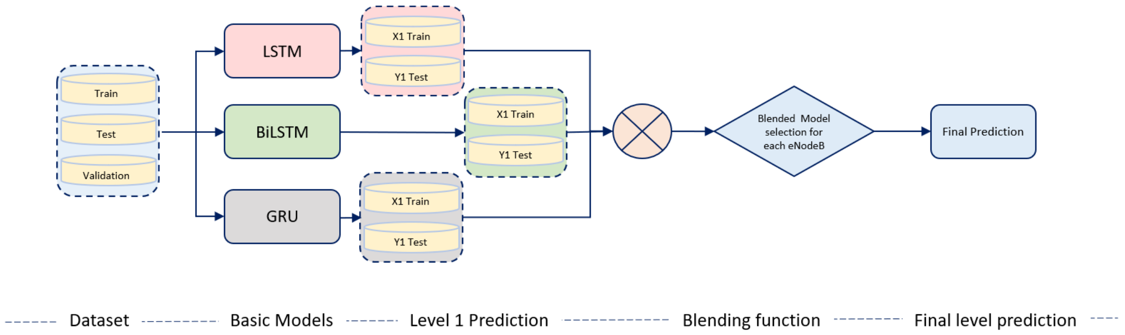


Figure 3.9: Fusion Model with blended approach

of LSTM, BiLSTM, and GRU models varies across datasets. Some datasets exhibit strong correlations with LSTM's ability to capture long-range dependencies, while others benefit from the bidirectional nature of BiLSTM or the efficiency of GRU.

3.8 Comprehensive RNN and Deep Regression

The conventional recurrent neural network (RNN) algorithm processes inputs individually, but it suffers from degraded performance over long time series or sequences due to its lack of memory in the architecture [27].

> To address this memory issue, we have chosen to utilize Long Short-Term Memory (LSTM) instead of the traditional RNN model. LSTM is an advanced version of the recurrent neural network model that excels at capturing chronological sequences and their long-range dependencies more effectively than conventional RNNs. It was specifically designed to tackle the long-term dependency problem faced by standard RNNs. In the LSTM architecture for computing forecasting traffic or any type of time series data, the process begins with calculating the output value from the previous time data and presenting the input series data, which is then used as input for the forget gate [24], [28].
>
> GRU is a relatively newer RNN model introduced by Kyunghyun Cho et al. in 2014, and it shares a similar architecture with LSTM. However, GRU models offer greater convenience and simplicity in both training and implementation. The neural network architecture of GRU reduces computational complexity due to the presence of update and reset gates, which allows it to effectively retain long-term states of the cell [23].
>
> Alternative of ML Regression algorithm, Deep architectures, or neural networks with several layers, are frequently used in deep regression. An input layer, one or more hidden layers, and an output layer make up these networks. The nodes (neurons) that make up each layer use weights and activation functions to alter the input data.By introducing non-linearities, activation functions allow the model to reflect intricate interactions between inputs and outputs. Deep regression frequently employs the ReLU (Rectified Linear Unit), sigmoid, and tanh activation functions.

Chapter 4

Experimental Results

This chapter will give an overview over the achieved results, the environment setup, used data and the experiment process to solve the given research questions. The following questions will be answered in detail, with section focusing on the setup of the data set and the experiments and section 4.4 presenting the achieved results.

We'll thoroughly review our models' performance in this chapter. To gauge how well our predictions correspond with the actual data, we'll make use of numbers, graphs, and comparisons. We'll also take a closer look at a unique method in which we combine various models to function as a cohesive one. This collaborative strategy, known as fusion, gives our analysis a fresh perspective.

We'll get into the specifics of our analysis as we go on. We'll discuss the performance of our various models and how they interacted. We'll compare the actual figures to the forecasts to determine how well they match. We'll learn from this process how effective our strategies are in practical settings. So let's set out on this voyage and investigate the revelations that lie in store for us in the sections that follow.

4.1 Environment setup and Sequential model design

TensorFlow, scikit-learn, and some common python libraries like pandas, seaborn, etc. are used to set up the virtual environment. The system requirements were created using Windows 10, a Ryzen 5 3600 processor, 32GB of RAM, and an RTX 3070 graphics card. The parameter configuration for creating the model and taking Multivariate input below structure shows the best evaluation score.

For each model *epochs size*: 100, *batch size*: 128, Adam optimizers with learning rate: 0.001 is considered. For optimized and efficient training, some callbacks are used like *EarlyStopping*: To stop training when a monitored metric has stopped improving, *ModelCheckpoint*: To save the Keras model or model weights at some frequency and *ReduceLROnPlateau*: To reduce the learning rate when a metric has stopped improving.

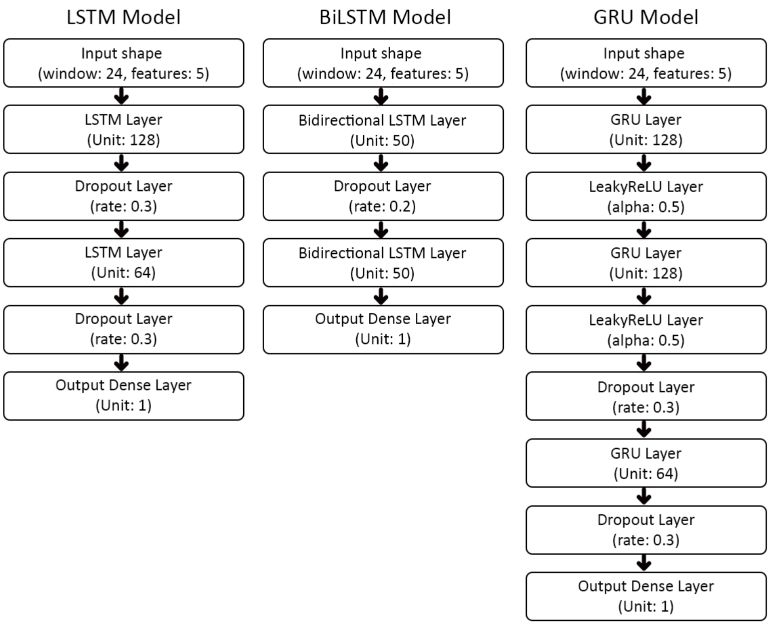

LSTM Model
Input shape (window: 24, features: 5)
LSTM Layer (Unit: 128)
Dropout Layer (rate: 0.3)
LSTM Layer (Unit: 64)
Dropout Layer (rate: 0.3)
Output Dense Layer (Unit: 1)
BiLSTM Model
Input shape (window: 24, features: 5)
Bidirectional LSTM Layer (Unit: 50)
Dropout Layer (rate: 0.2)
Bidirectional LSTM Layer (Unit: 50)
Output Dense Layer (Unit: 1)
GRU Model
Input shape (window: 24, features: 5)
GRU Layer (Unit: 128)
LeakyReLU Layer (alpha: 0.5)
GRU Layer (Unit: 128)
LeakyReLU Layer (alpha: 0.5)
Dropout Layer (rate: 0.3)
GRU Layer (Unit: 64)
Dropout Layer (rate: 0.3)
Output Dense Layer (Unit: 1)

Figure 4.1: Sequential model for LSTM, BiLSTM and GRU

Figure 4.1 illustrates the Sequential model of the multivariate LSTM, BiLSTM, and GRU models designed for predicting multiple time-series instances. These models are configured with five features and 24 time steps for the prediction process. The entire ensemble of 890 nodes across various sites undergoes training through this tailored model and proceeds to generate predictions for the subsequent 62-day period.

In Figure 4.2 shows that training and validation loss accuracy was closer within 17 *epochs* for a sample eNodeB

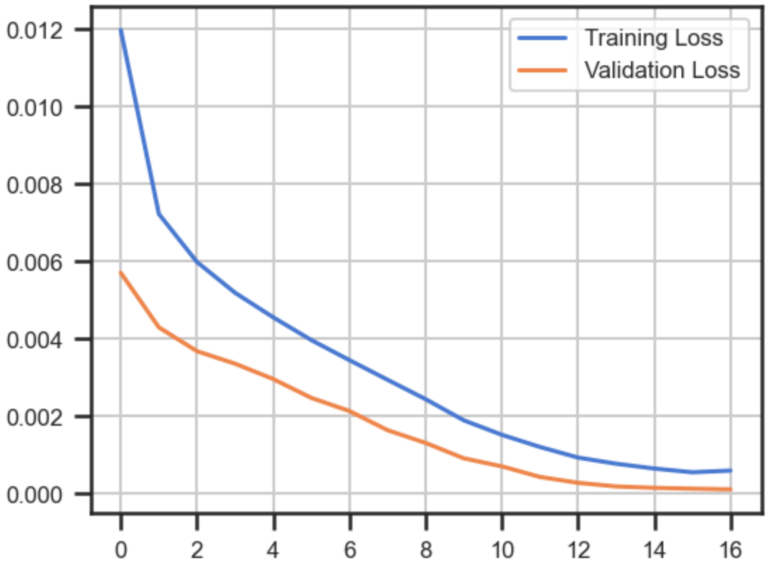


Figure 4.2: Training vs Validation loss of a sample eNodeB

4.2 Evaluation and Performance Metrics

Performance metrics are an integral component of every machine learning pipeline. They provide valuable insights into the progress and quantify the model's effectiveness. Regardless of whether the machine learning model is as simple as linear regression or as advanced as state-of-the-art techniques like BERT, a metric is essential to assess model performance. These metrics play a vital role in both regression and classification tasks, reflecting the nature of the performance evaluation. Metrics are utilized to monitor and evaluate the model's performance during both training and testing phases, and they do not necessarily need to be differentiable.

4.2.1 Regression Metrics

Regression metrics are evaluation metrics used to assess the performance of machine learning models in regression tasks. In regression, the goal is to predict continuous numerical values rather than discrete classes.

Mean Squared Error (MSE) is a widely used performance metric in regression tasks within the field of machine learning and statistics. It measures the average squared difference between the predicted values and the actual values of a continuous target variable.

$$MSE = \frac{1}{N}\sum_{k=1}^{n}(y_t - x_t)^2 \tag{4.1}$$

Root Mean Squared Error (RMSE) is the square root of MSE and is preferred when the metric needs to be in the same unit as the target variable. It represents the average magnitude of errors in the original scale of the data.

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{n}(y_t - x_t)^2} \tag{4.2}$$

Mean Absolute Error (MAE) measures the average absolute difference between the predicted values and the actual values. It is less sensitive to outliers compared to MSE. MAE is robust to outliers because it treats all errors equally. It is particularly useful when the dataset contains extreme values or when large errors are less critical for the application. For example, in some scenarios, being off by a certain amount may have similar consequences regardless of the actual value, and in such cases, MAE provides a more relevant evaluation metric

$$MAE = \frac{1}{N}\sum_{k=1}^{n}|y_t - x_t| \tag{4.3}$$

R-squared (R^2) also known as the coefficient of determination, evaluates the proportion of variance in the target variable that can be explained by the model. It provides a measure of how well the model fits the data.The coefficient of determination can also be interpreted as a percentage by multiplying it by 100.

$$R^2 = 1 - \frac{\sum_{k=1}^{n}(y_t - x_t)^2}{\sum_{k=1}^{n}(\bar{y} - x_t)^2} \tag{4.4}$$

4.3 Experimental Model Outcome

We have forecasted the 62-day traffic as well as the usage of that specific eNodeB for each of the 36 clusters out of the total 890 eNodeB using the system design and trained model. In the first stage, we use all three deep learning algorithms—LSTM, BiLSTM, and GRU—to predict traffic for each enodeB. Then, components using an ensemble technique, such as Blending, a fusion of deep learning algorithms create to mix all of the deep learning algorithms to forecast best for our datasets after determining which deep learning models match well for which specific enodeB.

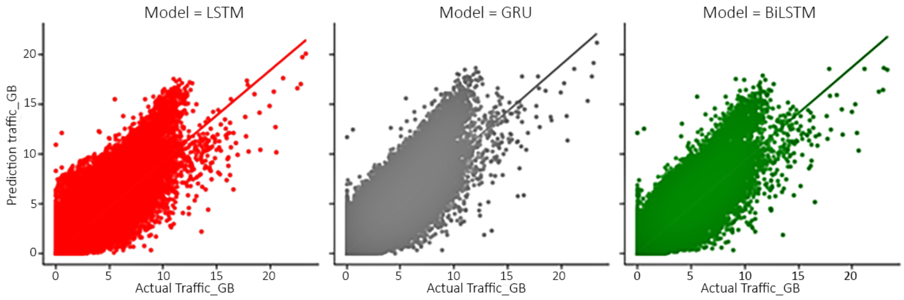


Figure 4.3: Regression plots of the models at the training phase

Model	MSE	MAE	RMSE	R^2
LSTM	0.4478	0.4355	0.6692	0.7635
GRU	0.4461	0.4346	0.6679	0.7644
BiLSTM	0.3922	0.4158	0.6262	0.7929

Table 4.1: Performance of the model in the testing phase

Evaluating the model's overall accuracy Evaluation Criteria are combined and shown in Fig. 4.3. This plot is used to find the predicted and actual values relationship. Also, Table 4.1 presents the testing results of the proposed model.

4.4 Performance of the Fusion Model

According to each node's observation, BiLSTM achieves a high score, 24% in LSTM and 20% in GRU, out of 890 nodes in 56% of nodes. The model's forecasting accuracy varied because different nodes' traffic patterns differ. A fusion model was used in this situation, and the best model was chosen based on training accuracy and minimal loss. The overall system's prediction R^2 score from the fusion model was 0.8034. The experimental outcomes of the suggested model during the testing phase were ideal, as shown by Fig.4.4 and Table 4.2.

Figure 4.5 also displays an example of eNodeB's traffic patterns with various model and R2 values. The finding supports the hypothesis that the proposed multivariate fusion model is capable of time series traffic forecasting and demonstrates the identical fundamental character of the actual and forecasted data sets.

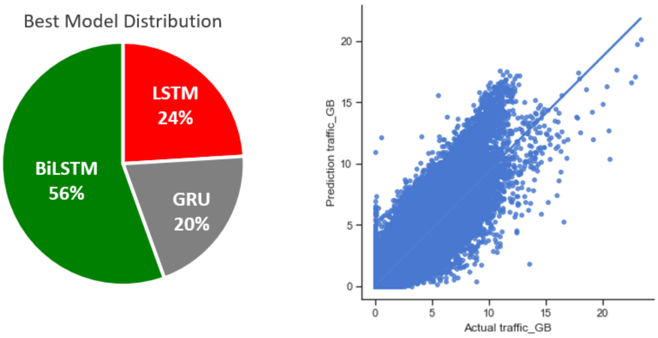


Figure 4.4: Regression plots of the Fusion Model

Fusion Model	
MSE	0.3723
MAE	0.4025
RMSE	0.6101
R-Square	0.8034

Table 4.2: Performance of the Fusion Model

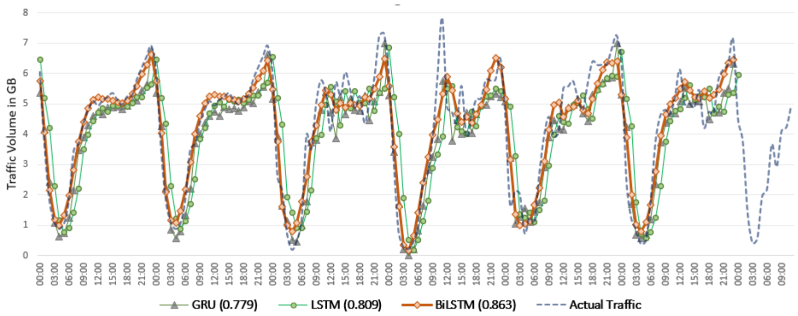


Figure 4.5: Actual vs. Predicted Traffic based on different Model

4.5 Discussion of Performance Evaluation

This chapter has given a general overview of the methodology we used to choose the best deep learning models that were customized for each unique eNodeB and allowed for precise traffic forecast. We used various deep learning algorithms after determining the best deep learning model for each eNodeB to create a fusion model for thorough training. Our results show that this fusion model fared better at predicting cellular network traffic than individual models.

Then, using the Deep Regression technique, we were able to use the trained model to estimate traffic volume (Vol) and then predict LTE channel Physical Resource Block (PRB) Utilization ($PRBU$) for various eNodeB clusters. By establishing acceptable throughput benchmarks, PRB utilization is crucial for preserving Quality of Services (QoS). Our fusion model showed that it could count highly utilized samples accurately and with few errors. This proved the fusion model's usefulness.

Chapter 5

Smart Network Management

5.1 LTE QoS plaining and Over utilization solution

After predicting both traffic and utilization, the network planner used network optimization techniques to solve over utilization problems. These methods involved predicting radio characteristics with a goal of reducing Mobile Network Operators' Capital Expenditure (CapEx) spending. Planner decides whether to use *step-up* or *long-term* solutions based on the design threshold. In *step-up* solutions, planner offers soft parameter tuning or enables capacity adjustment of CQI Switch or PRB power boost, on the other hand in *long-term* solutions It is recommended for hard capacity expansion with the implementation of Multibeam Cell Split Solution [7], New Spectrum addition [8], and planning and deployment of a new node [8], [10]. Capital expenditure (CapEx) also increases with new node expansion.

5.2 Capacity Monitoring after Planning and Deployment

After the initial planning or existing eNodeB upgradation and implementation phases, LTE capacity monitoring is a critical component in maintaining the performance and efficiency of a cellular network. LTE is a commonly utilised wireless technology that offers mobile devices high-speed data and multimedia capabilities. Following the completion of network planning and deployment, continuing network capacity monitoring is crucial for a number of reasons, including network *performance optimisation, congestion management, load balancing, resource efficiency, and subscriber experience management.*

In a nutshell LTE capacity monitoring is essential for preserving the overall functionality and health of cellular networks. It gives network operators the knowledge and abilities needed to control network congestion, optimise performance parameters, make informed strategic decisions, and guarantee a top-notch user experience. Operators can dynamically adjust to shifting conditions and demands by keeping a close eye on how network resources are being used. The ability to proactively address new difficulties is fostered by this continual process of network capacity monitoring and analysis, which ultimately improves organisational performance and customer happiness. In a constantly changing technical environment, LTE capacity monitoring essentially acts as the cornerstone for preserving the success, responsiveness, and functionality of cellular networks.

5.3 Problem Statement in Legacy Capacity Management

The increasing demand for data-intensive applications, and the complex interplay of network components. In Legacy LTE capacity management faces significant hurdles in efficiently handling the growing demand for data services, optimizing network resources, and maintaining a consistent quality of service (QoS) for users. As the usage of data-hungry applications continues to rise, legacy LTE networks struggle to accommodate the increased traffic load, resulting in potential congestion, reduced data speeds, and deteriorating user experiences. Moreover, the legacy architecture's static resource allocation approach lacks the flexibility needed to adapt to varying usage patterns and demands. This rigid allocation can lead to under utilization of resources in some areas and over utilization in others, hindering overall network efficiency.

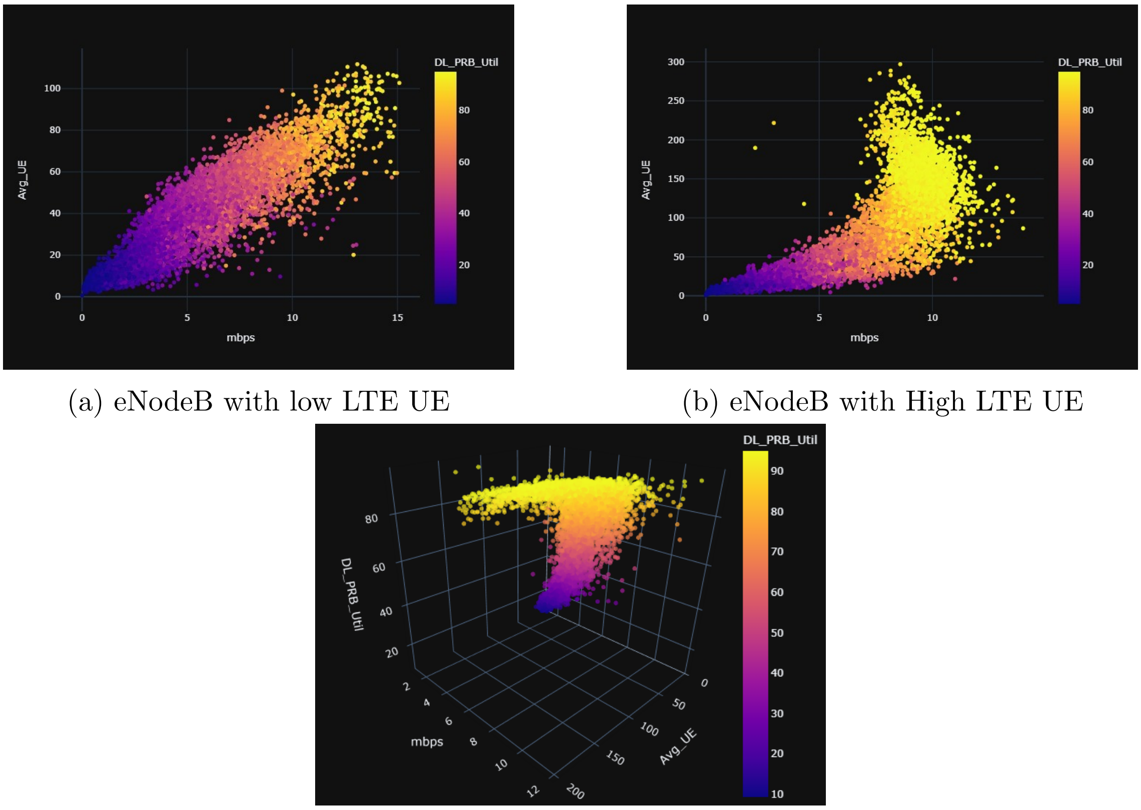
Furthermore, legacy LTE capacity management often relies on manual interventions and lacks real-time insights into network performance. This reactive approach not only delays issue identification but also limits the operators' ability to proactively address capacity constraints and bottlenecks. The lack of comprehensive and automated monitoring tools hampers the operators' capacity to accurately predict network congestion points and plan for necessary upgrades or optimizations.

In addition, the complexities introduced by the coexistence of multiple technologies, such as LTE and earlier generations, exacerbate the challenge of legacy LTE capacity management. Ensuring seamless interoperability and optimizing the utilization of resources across different technologies poses intricate technical hurdles.

Overall, legacy LTE capacity management struggles to strike a balance between catering to ever-increasing data demands, efficiently allocating network resources, and delivering a satisfactory user experience. Transformative strategies and advanced tools are imperative to modernize capacity management practices and alleviate the constraints imposed by the limitations of legacy LTE networks.

5.4 Traditional Network Management

The amount of LTE User Equipment (UE), which includes things like smartphones and tablets, can have a big impact on how much user throughput (measured in megabits per second, Mbps), is actually available. However, the user throughput (Mbps) metric is typically the only consideration when performing an LTE capacity analysis. Additionally, in the process of aggregating data at the network level to assess spectrum efficiency, there is a tendency to overlook or smooth out the distinctive characteristics associated with individual network nodes.



(a) eNodeB with low LTE UE

(b) eNodeB with High LTE UE

(c) eNodeB with High LTE UE 3Dimension relation

Figure 5.1: Characteristics of two LTE 10Mhz BW Nodes

Fig 5.1 its observe that with same configuration and capacity of two different eNodeB have different characteristics, Fig 5.1a shows that eNodeB with lower UE can serve upto 15 Mbps at height PRB utilization, on the other hand in Fig 5.1b with high UE can serve upto 10 Mbps and bootblack to height PRB utilization. As a result, each eNodeB exhibits unique characteristics based on user behavior, and this is dependent not only on the traffic volume they can handle but also on the UE/devices that are simultaneously accessing that specific eNodeB. The Fig 5.2 also shows that utilization is highly correlate with UE and traffic volume.
In traditional capacity monitoring with manual interventions, not all usage patterns are typically recorded and for simplicity only Network Busy Hours (NBH) samples are taken to determine throughput at a desired PRB usage. But when the NBH sample of all eNodeBs is combined for regression, the idiosyncrasies of an individual eNodeB are diluted.

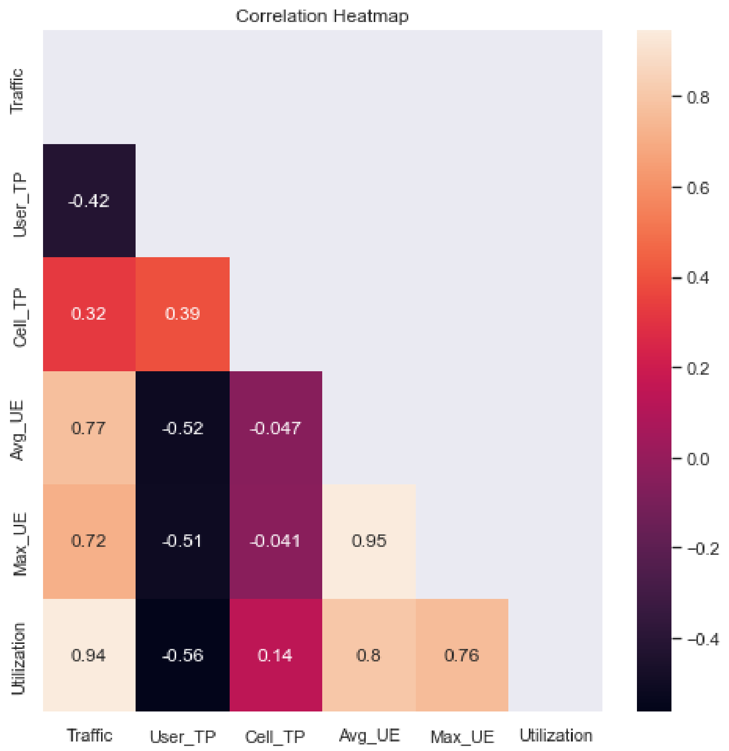
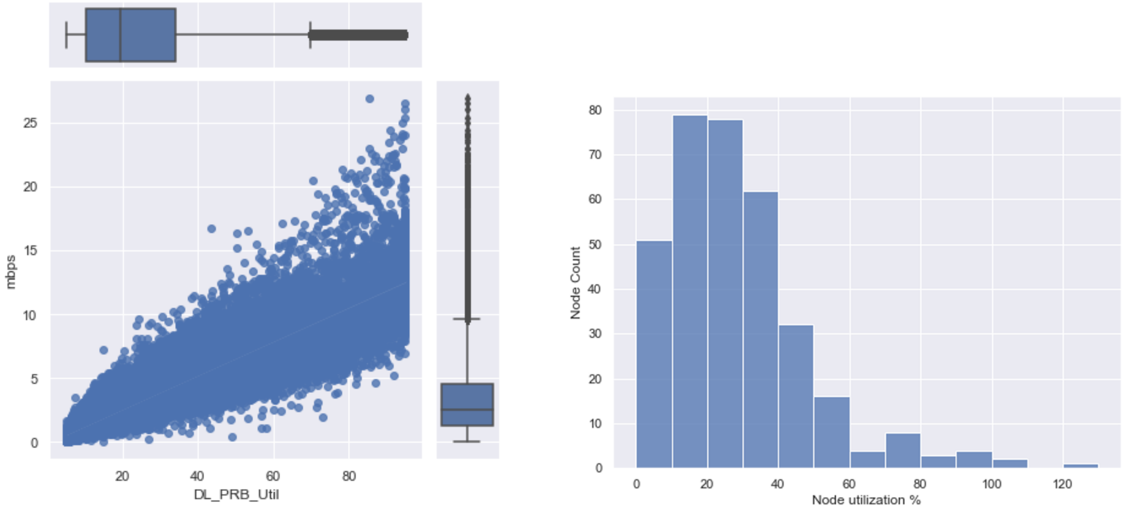


Figure 5.2: Correlation Heatmap of LTE KPI

In Fig 5.3a, Outliers sample not removed in NBH Sample, so overfitting or underfitting will bias the data, Although it shows that the coefficient of determination is $R2$ 0.8562 and the predicted throughput is 10.9511 mbps at 80% utilization, it is not true that all eNodeB will provide 10 mbps at 80% utilization.

From Fig 5.4 shows that high capacity utilized eNodeB ($> 80\%$) have lower PRB utilization ($< 85\%$) *(Green dot and Orange dot)* which determine that utilization



(a) NBH Sampless with outliers

(b) Utilization and Histogram

Figure 5.3: Network Busy Hours Samples eNodesB

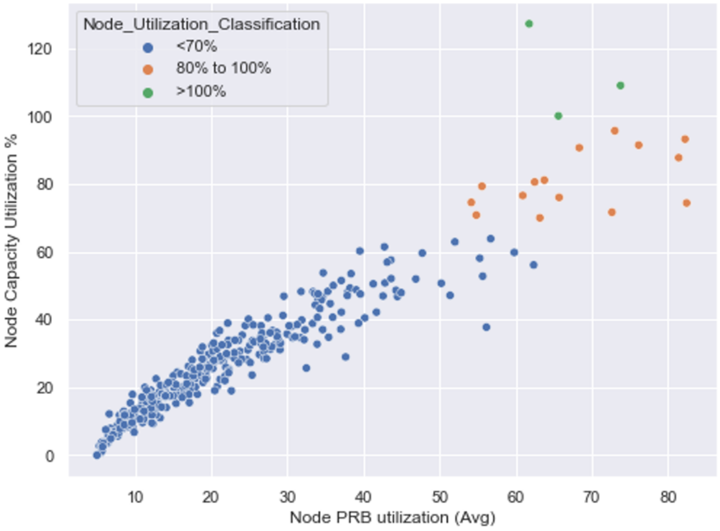


Figure 5.4: Capacity Utilization vs PRB Utilization from NBH Sample

tagging is not identified properly.

5.5 Smart LTE Node monitoring

In this section, we'll talk about using machine learning algorithms like XGBoost regression to monitor **each individual Node's capacity** use in a methodical manner. Before starting the train test split, we first employ interquartile range (IQR) to remove outliers from the sample that could dilute eNodeB characteristics.

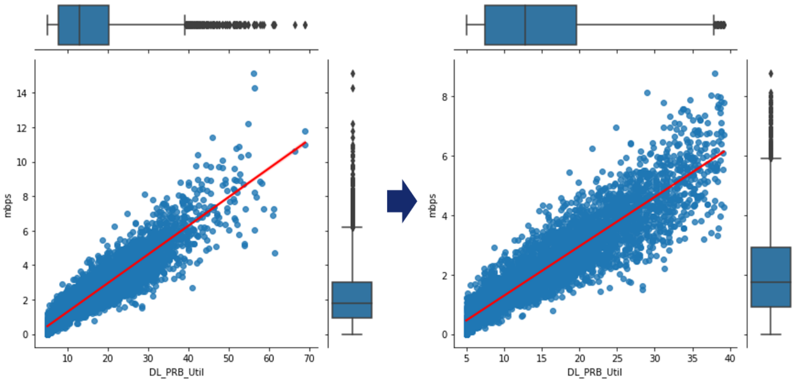


Figure 5.5: Outliers detection and remove for one sample eNodeB

In next stage, eNodeB sample split into train and test in 70:30 ratio and use XGBRegressor to identify UE throughput at desire PRB utilization (80%). To maximized the accuracy we introduce GridSearchCV to find best value during hyperparameters settings *"max_depth": [4, 5, 6], "n_estimators": [500, 600, 700], "learning_rate": [0.01, 0.015]*. As outcome with this training for that specific eNodeB the coefficient of determination is $R2$ 0.9556 and the predicted throughput is 13.2316 mbps at 80% utilization.

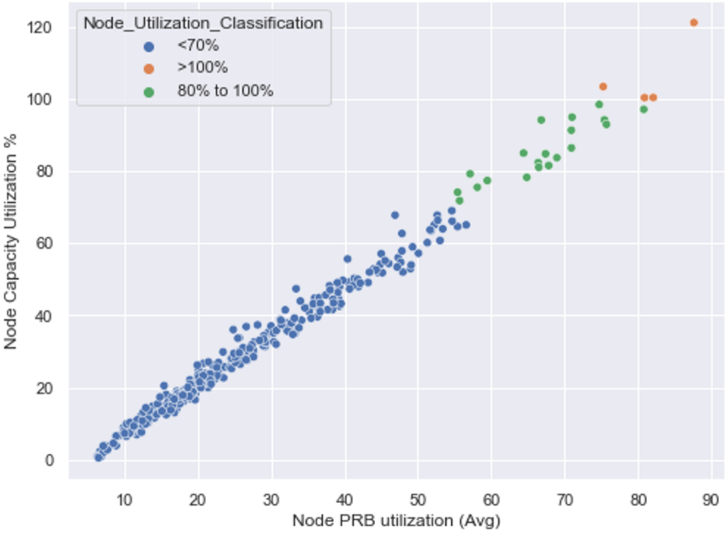


Figure 5.6: Capacity Utilization vs PRB Utilization with Deep-Dive Approach

This process was repeated for all 890 eNodeB and fig 5.6 shows that there is symmetric relation between Capacity Utilization vs PRB Utilization for individual Nodes. Now with this Deep-Dive Approach LTE Nodes are monitored and tagged with proper utilization tag.

Chapter 6

Conclusion

6.1 Conclusion

In order to solve the NP-hard problem of maximizing user throughput, this research ingeniously develops a fusion model in a combination of three deep learning algorithms for the most detailed level of cellular network traffic prediction. A Deep-Dive approach is also suggested in order to correctly identify the high utilized Nodes following the initial planning or current eNodeB upgrade and implementation phases.

Using the Fusion Strategy in RNN, the suggested model's accuracy is raised by $6.6 - 7.0\%$ while maintaining an excellent R^2 score, or 0.8034, which denotes a very accurate forecast of network traffic volume. Another innovation of this research is the Deep-Drive strategy for managing network capacity, which would assist network engineers in managing the capacity after planning and implementing soft parameters before the quality of service (QoS) deteriorates in comparison to the benchmark. Thus, customers will be less sufferer from capacity expansion lead time from the MNO side.

6.2 Future Research

In the future, we will use Restricted Boltzmann Machines (RBM) with Conditional Random Fields (CRFs) to address the prediction of traffic peaks during social events in a specific geographic area or eNodeB serving area. CRFs could help incorporate contextual information and relationships between different factors affecting network traffic, such as event type, location, time of day, and more. on other hand RBM designed to recognize patterns and relationships within data. In this context, RBMs might be used to analyze historical data and identify patterns in network usage during previous social events.

Additionally, depending on anticipated traffic and consumer demand, we will concentrate on smooth dynamic resource allocation in heterogeneous complex networks systems, including GSM, LTE, 5G, and beyond technologies of a specific era.

Bibliography

[1] R. J. Erb, “Introduction to backpropagation neural network computation,” *Pharmaceutical research*, vol. 10, pp. 165–170, 1993.

[2] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[4] I. Toyoda, F. Nuno, Y. Shimizu, and M. Umehira, “Proposal of 5/25-ghz dual band ofdm-based wireless lan for high-capacity broadband communications,” in *2005 IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications*, IEEE, vol. 3, 2005, pp. 2104–2108.

[5] Y. Singh and A. S. Chauhan, “Neural networks in data mining.,” *Journal of Theoretical & Applied Information Technology*, vol. 5, no. 1, 2009.

[6] V. D. Blondel, M. Esch, C. Chan, *et al.*, “Data for development: The d4d challenge on mobile phone data,” *arXiv preprint arXiv:1210.0137*, 2012.

[7] M. Caretti, M. Crozzoli, G. Dell’Aera, and A. Orlando, “Cell splitting based on active antennas: Performance assessment for lte system,” in *WAMICON 2012 IEEE Wireless & Microwave Technology Conference*, IEEE, 2012, pp. 1–5.

[8] J. Xiao, R. Q. Hu, Y. Qian, L. Gong, and B. Wang, “Expanding lte network spectrum with cognitive radios: From concept to implementation,” *IEEE Wireless Communications*, vol. 20, no. 2, pp. 12–19, 2013.

[9] H. Al-Zayadi, O. Lavriv, M. Klymash, and A.-S. Mushtaq, “Increase throughput by expectation channel quality indicator,” in *2014 First International Scientific-Practical Conference Problems of Infocommunications Science and Technology*, IEEE, 2014, pp. 120–121.

[10] N. Morozs, T. Clarke, and D. Grace, “Intelligent secondary lte spectrum sharing in high capacity cognitive cellular systems,” in *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*, IEEE, 2015, pp. 1–2.

[11] M. B. Shahab, M. A. Wahla, and M. T. Mushtaq, “Downlink resource scheduling technique for maximized throughput with improved fairness and reduced bler in lte,” in *2015 38th International Conference on Telecommunications and Signal Processing (TSP)*, IEEE, 2015, pp. 163–167.

[12] M.-C. Nguyen, H. Nguyen, D.-H. Nguyen, E. Georgeaux, P. Mege, and L. Martinod, "Adaptive physical resource block design for enhancing voice capacity over lte network in pmr context," in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, IEEE, 2016, pp. 1–5.

[13] F. Xu, Y. Lin, J. Huang, *et al.*, "Big data driven mobile traffic understanding and forecasting: A time series approach," *IEEE Transactions on Services Computing*, vol. 9, no. 5, pp. 796–805, 2016. DOI: 10.1109/TSC.2016.2599878.

[14] H. Hota, R. Handa, and A. Shrivas, "Time series data prediction using sliding window based rbf neural network," *International Journal of Computational Intelligence Research*, vol. 13, no. 5, pp. 1145–1156, 2017.

[15] M. M. Hasan, S. Kwon, and J.-H. Na, "Adaptive mobility load balancing algorithm for lte small-cell networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2205–2217, 2018. DOI: 10.1109/TWC.2018.2789902.

[16] H. D. Trinh, L. Giupponi, and P. Dini, "Mobile traffic prediction from raw data using lstm networks," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, 2018, pp. 1827–1832.

[17] D. Babicz, A. Tihanyi, M. Koller, C. Rekeczky, and A. Horváth, "Simulation of an analogue circuit solving np-hard optimization problems," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, 2019, pp. 1–5.

[18] D. Chmieliauskas and D. Guršnys, "Lte cell traffic grow and congestion forecasting," in *2019 Open Conference of Electrical, Electronic and Information Sciences (eStream)*, IEEE, 2019, pp. 1–5.

[19] J. Kim and N. Moon, "Bilstm model based on multivariate time series data in multiple field for forecasting trading area," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2019.

[20] S. Lathuilière, P. Mesejo, X. Alameda-Pineda, and R. Horaud, "A comprehensive analysis of deep regression," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 9, pp. 2065–2081, 2019.

[21] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of lstm and bilstm in forecasting time series," in *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 3285–3292.

[22] N. Tavakoli, "Modeling genome data using bidirectional lstm," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, IEEE, vol. 2, 2019, pp. 183–188.

[23] J. Yuan, H. Wang, C. Lin, D. Liu, and D. Yu, "A novel gru-rnn network model for dynamic path planning of mobile robot," *IEEE Access*, vol. 7, pp. 15 140–15 151, 2019.

[24] X. Yuan, L. Li, and Y. Wang, "Nonlinear dynamic soft sensor modeling with supervised long short-term memory network," *IEEE transactions on industrial informatics*, vol. 16, no. 5, pp. 3168–3176, 2019.

[25] L. Feng, W. Li, Y. Lin, L. Zhu, S. Guo, and Z. Zhen, “Joint computation offloading and urllc resource allocation for collaborative mec assisted cellular-v2x networks,” *IEEE Access*, vol. 8, pp. 24 914–24 926, 2020.

[26] A. Kirmaz, D. S. Michalopoulos, I. Balan, and W. Gerstacker, “Mobile network traffic forecasting using artificial neural networks,” in *2020 28th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2020, pp. 1–7. DOI: 10.1109/MASCOTS50786.2020.9285949.

[27] X. Wang and D. Liang, “Lstm-based alarm prediction in the mobile communication network,” in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, IEEE, 2020, pp. 561–567.

[28] A. Yadav, C. Jha, and A. Sharan, “Optimizing lstm for time series prediction in indian stock market,” *Procedia Computer Science*, vol. 167, pp. 2091–2100, 2020.

[29] F. W. Alsaade and M. Hmoud Al-Adhaileh, “Cellular traffic prediction based on an intelligent model,” *Mobile Information Systems*, vol. 2021, 2021.

[30] J. L. Bejarano-Luque, M. Toril, M. Fernandez-Navarro, C. Gijon, and S. Luna-Ramirez, “A deep-learning model for estimating the impact of social events on traffic demand on a cell basis,” *IEEE Access*, vol. 9, pp. 71 673–71 686, 2021.

[31] H. S. Jang, H. Lee, H. Kwon, and S. Park, “Deep learning-based prediction of resource block usage rate for spectrum saturation diagnosis,” *IEEE Access*, vol. 9, pp. 59 703–59 714, 2021. DOI: 10.1109/ACCESS.2021.3073670.

[32] N. Li, L. Hu, Z.-L. Deng, T. Su, and J.-W. Liu, “Research on gru neural network satellite traffic prediction based on transfer learning,” *Wireless Personal Communications*, vol. 118, no. 1, pp. 815–827, 2021.

[33] A. Masood, T.-V. Nguyen, and S. Cho, “Deep regression model for videos popularity prediction in mobile edge caching networks,” in *2021 International Conference on Information Networking (ICOIN)*, IEEE, 2021, pp. 291–294.

[34] D. Rügamer, C. Kolb, C. Fritz, *et al.*, “Deepregression: A flexible neural network framework for semi-structured deep distributional regression,” *arXiv preprint arXiv:2104.02705*, 2021.

[35] H. Widiputra, A. Mailangkay, and E. Gautama, “Multivariate cnn-lstm model for multiple parallel financial time-series prediction,” *Complexity*, vol. 2021, 2021.

[36] “Ericsson mobility report.” (Jan. 2022), [Online]. Available: https://www.ericsson.com/49d3a0/assets/local/reports-papers/mobility-report/documents/2022/ericsson-mobility-report-june-2022.pdf.

[37] Y. Fang, S. Ergüt, and P. Patras, “Sdgnet: A handover-aware spatiotemporal graph neural network for mobile traffic forecasting,” *IEEE Communications Letters*, vol. 26, no. 3, pp. 582–586, 2022. DOI: 10.1109/LCOMM.2022.3141238.

[38] L. Lo Schiavo, M. Fiore, M. Gramaglia, A. Banchs, and X. Costa-Perez, “Forecasting for network management with joint statistical modelling and machine learning,” in *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2022, pp. 60–69. DOI: 10.1109/WoWMoM54355.2022.00028.

[39] F. Sun, P. Wang, J. Zhao, *et al.*, “Mobile data traffic prediction by exploiting time-evolving user mobility patterns,” *IEEE Transactions on Mobile Computing*, vol. 21, no. 12, pp. 4456–4470, 2022. DOI: 10.1109/TMC.2021.3079117.

[40] Q. Yu, H. Wang, T. Li, *et al.*, “Network traffic overload prediction with temporal graph attention convolutional networks,” in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2022, pp. 885–890. DOI: 10.1109/ICCWorkshops53468.2022.9814643.

[41] S. T. Nabi, M. R. Islam, M. G. R. Alam, *et al.*, “Deep learning based fusion model for multivariate lte traffic forecasting and optimized radio parameter estimation,” *IEEE Access*, vol. 11, pp. 14 533–14 549, 2023. DOI: 10.1109/ACCESS.2023.3242861.