ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOG

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

TELECOMMUNICATION NETWORK ENGINEERING PROGRAM

Mobile Data Traffic Prediction Using Multivariate Time Series Data: The Case of LTE Network in Addis Ababa

By

Endale Mare

Advisor

Dr. –Ing. Dereje Hailemariam

A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Telecommunication Engineering.

September 2021

Addis Ababa, Ethiopia

ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOGY

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

TELECOMMUNICATION NETWORK ENGINEERING PROGRAM

Mobile Data Traffic Prediction Using Multivariate Time Series Data: The Case of LTE Network in Addis Ababa

By:

Endale Mare

Approval by Board of Examiners

_____	_____	____/___/_____
Chairman, School Graduate Committee	Signature	Date

Committee

Dr. –Ing. Dereje Hailemariam	_____	_____/_____/_____
Advisor Name	Signature	Date
_____	_____	_____/_____/_____
Internal examiner	Signature	Date
_____	_____	_____/_____/_____
External examiner	Signature	Date

Declaration

I, the undersigned, declare that this thesis is my work and does not incorporate any material previously submitted for a degree or diploma in any other university or institute of higher learning without acknowledgment. To the best of my knowledge, the work does not contain any material previously published or written by another person without recognition.

Endale Mare _____

Name Signature

Addis Ababa _____/_____/_____

Place Date of Submission

The above candidate has carried out research for the master's thesis under my supervision.

Dr. –Ing. Dereje Hailemariam _____

Name Signature

Abstract

Due to various reasons including the advancement of mobile devices and the proliferation of data-intensive applications, the demand for mobile data traffic is increasing rapidly. Mobile network providers are facing a challenge in improving the Quality of Service (QoS) and user experience due to ever growing data demand. Network optimization and expansion are continuous activities that enhance network quality as well as alleviates network capacity crunch. Nowadays, accurate prediction models are becoming increasingly important for predicting future data traffic demand. Anticipating data traffic demand enables operators to use it for optimization and upgrade, resulting in efficient resource utilization.

In this research, a deep learning-based prediction model is proposed to predict future cellular data traffic demand using multivariate input features. The model is built with a hybrid Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) networks, called the CNN-LSTM model. Initially, the base stations are clustered using K-Means clustering based on temporal traffic patterns, and then a prediction model is developed per cluster level. The model is implemented with a deep learning library, Keras. The effectiveness of the CNN-LSTM model is evaluated using a dataset collected from ethio telecom LTE network and various metrics namely RMSE, MAPE and R^2 are used for performance evaluation. The research compared the model performance for univariate and multivariate input features cases. The results confirm that the CNN-LSTM multivariate features improved the RMSE and MAPE of the model by 58% and 50% respectively. The proposed model is also compared with CNN and SARIMA models and the proposed model outperforms both models in all evaluation metrics criteria.

Keywords: Multivariate features, LTE Technology, Deep Learning, CNN, LSTM, 1D CNN-LSTM, Mobile Data Traffic

Acknowledgment

First and foremost, I thank the almighty God for giving me all the help, courage, and strength to accomplish the thesis.

Second, I would like to deeply thank Dr. -Ing. Dereje Hailemariam for his guidance, encouragement, and constructive comments throughout my thesis. It would not have been possible to complete and achieve the ultimate target of this research without his invaluable support, enthusiasm, and commitment.

Likewise, I would like to express my sincere gratitude to Dr. Ephrem Teshale for his constructive comments and helpful suggestions during thesis progress report presentations.

I would also like to thank Ms. Bethelhem Seifu and Mr. Habtamu Abayneh for their invaluable support in this research.

Finally, I'd like to express my heartfelt gratitude to my beloved wife, Yemesirach Dessie, and my children, Yadon and Elnathan. I am able to complete this research because of your unwavering support, unconditional love, and your precious time.

Table of Contents

List of Figures

List of Table

List of Acronyms

1D CNN	One Dimensional Convolutional Neural Network
2D CNN	Two Dimensional Convolutional Neural Network
2G	Second Generation
3G	Third Generation
4G	Fourth Generation
AA	Addis Ababa
Adam	Adaptive Moment
ANN	Artificial Neural Network
CDMA	Code Division Multiple Access
CNN	Convolutional Neural Network
ELM	Extreme Learning Machine
eNodeB	Evolved NodeB
EPC	Evolved Packet Core
EPS	Evolved Packet System
E-UTRNAN	Evolved Universal Terrestrial Radio Access Network
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
GB	Giga Byte
GPRS	General Packet Radio Service
GTP	GPRS Tunneling Protocol
HSDPA	High Speed Downlink Packet Access
HSS	Home Subscriber Server
LSTM	Long Short Term Memory
LTE	Long Term Evolution

LTE-A	LTE- Advanced
M2M	Machine to Machine
MAPE	Mean Absolute Percentage Error
ML	Machine Learning
MME	Mobility Management Entity
OFDMA	Orthogonal Frequency Division Multiple Access
PCRF	Packet Control and Charging Rules Function
P-GW	Packet Gateway
QoS	Quality of Service
RAB	Radio Access Bearer
RAN	Radio Access Network
RMSE	Root Mean Square Error
RMSProp	Root Mean Square Propagation
RNC	Radio Network Controller
RNN	Recurrent Neural Network
SAE	System Architecture System
SARIMA	Seasonal Auto Regressive Integrated Moving Average
SGD	Stochastic Gradient Descent
S-GW	Serving Gateway
TDD	Time Division Duplex
UE	User Equipment
UMTS	Universal Mobile Telecommunication System
UTRAN	Universal Terrestrial Radio Access Network
WiMAX	Worldwide interoperability for Microwave Access

1. Introduction

1.1 Background

Mobile data traffic demand is increasing globally for several reasons, including continuous evolution of smarter mobile phones, the emerging of machine-to-machine (M2M) connections, and the availability of attractive and data-intensive applications [1]. Different radio access network technologies, such as General Packet Radio Service (GPRS), Universal Mobile Telecommunication System (UMTS), and Long Term Evolution (LTE) are used to provide cellular mobile data services. Some Mobile Network Operators (MNOs) have even started deploying Fifth Generation (5G) networks for various use cases that have diverse requirements [2]. The choice of radio access technology depends on the operators' interest to satisfy their customers' needs by providing a better quality service while reducing associated costs towards implementation and operational costs.

Nowadays, the availability and quality of mobile network services have a major influence on the day-to-day activities of customers. MNOs strive to maintain quality of service in which the service provided to the customers should meet the specified key performance indicates (KPI) of the network. As the demand for data traffic increases continuously, the deployed network needs constant optimization and capacity enhancement to alleviate the capacity crunch. Different techniques are applied to overcome the intermittent and poor quality of service that rises due to capacity crunch. Those approaches include network densification, traffic offloading, spectral efficiency improvement, and using more radio spectrum [3]. MNOs select the appropriate method based on their customer demand and financial capability. Time-series prediction methods play a vital role to forecast future demands for several real-world applications including mobile data traffic

demand [4]. Cellular network prediction models can be used as an input for optimization, network upgrade, network expansion, reducing power consumption, and/or backhaul transmission dimensioning that enables efficient utilization of resources. Those mobile data prediction models are broadly grouped as *conventional* and *computational intelligence* models [5]. Some of the conventional methods include Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA). On the other hand, the computational intelligence techniques include machine learning and deep learning-based models such as Convolutional Neural Network (CNN), Long Term Short Memory (LSTM) networks. These models can be applied for temporal dimension or spatiotemporal dimensions.

Similar to the global data traffic, the data traffic demand in ethio telecom, the major telecom operator in Ethiopia, is increasing, and more data is generated from Addis Ababa city. According to the report in [6] the amount of mobile data traffic generated in Addis Ababa consists of 59.2% of the total data traffic generated across the country. This is in sharp contrast to the fact that the sites in Addis Ababa account for only 10.4% of the entire sites deployed throughout the country.

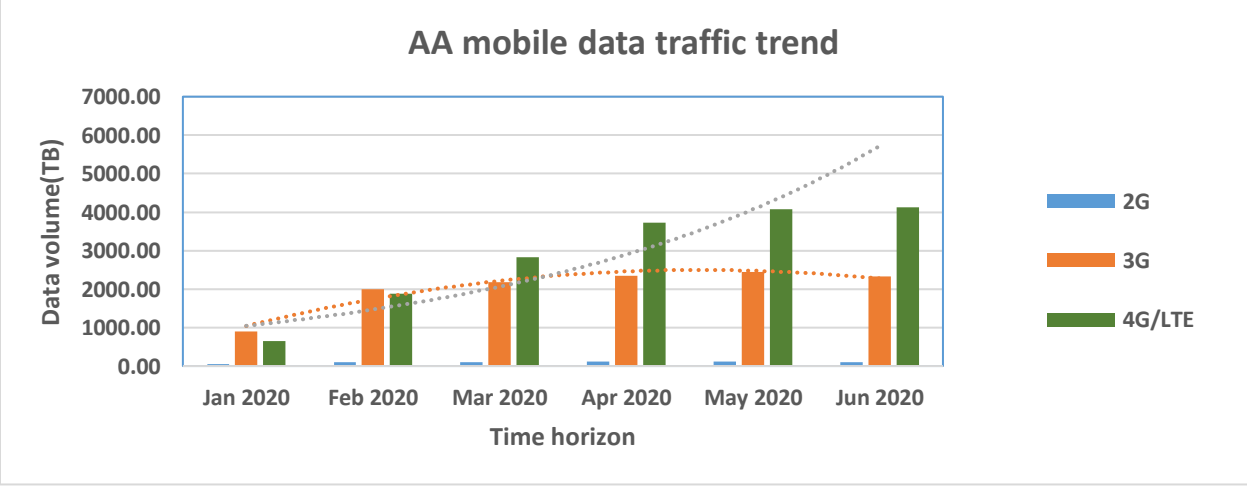


Figure 1.1 2G, 3G and LTE/LTE-A mobile data traffic trend in AA [7].

Ethio telecom deployed Second Generation (2G), Third Generation (3G), as well as Fourth Generation (4G) LTE and LTE-Advanced technologies to provide mobile service in Addis Ababa. Figure 1.1 above illustrates the mobile data traffic trend in Addis Ababa for a duration of six months in the year 2020 for each cellular network technology. LTE network data traffic is increasing significantly, while the 3G network generates a considerable amount of data traffic with a stagnant growth rate. When we consider the amount of data traffic generated from the 2G network, it is insignificant compared to that of the 3G and LTE networks.

The growth in LTE/LTE-Advanced data traffic in Addis Ababa is coupled with an increase in the number of LTE subscribers. As illustrated in Table 1.1 below, the number of customers increased significantly in 2020 (2020* only two-quarters data)

Table 1.1 Number of 4G subscribes in ethio telecom(AA) [8].

Year	2015	2016	2017	2018	2019	2020*
No. of 4G Subscribers	68996	127090	244084	379254	494149	1025095

As the number of LTE subscribers increases, the data usage pattern changes in time and space accordingly. Hence, continuous network traffic analysis is needed to improve the quality of service (QoS); data traffic prediction is one key approach for traffic analysis as it helps to determine the traffic demand proactively. Several works are done to model the dynamics of data traffic that are influenced by factors such as time, spaces, business, and holidays [9]. Most of the earlier works are based on ARIMA and SARIMA which are suitable for linear and stationary datasets. With the development of data analytics towards machine learning, some algorithms like Support Vector Machine (SVM) [10] Linear Regression (LR) [11] have been used to predict cellular traffic demand. Recently, due to the success of neural networks for a nonlinear, complex, and multivariate dataset,

many researchers used deep learning models exhaustively to develop cellular data traffic prediction models. Those prediction models can be developed by using univariate or multivariate features. The univariate prediction model contains data traffic as the only feature, whereas the multivariate model's features differ from one literature to the others depending on the data collection approach and the purpose of the multivariate features. In this study, multivariate features are used to incorporate network as well as customers' data usage behavior. Those features include downlink data traffic volume, uplink data traffic volume, downlink average throughput, cell average user, number of RAB attempts, and cell maximum user. In addition to improving model performance [12], multivariate features greatly aid in handling missing values in the dataset. Motivated by the capability of the one-dimensional CNN to capture salient features from multivariate features and LSTM to alleviate long-term dependency problems, the CNN-LSTM model is proposed in this research for mobile data traffic demand prediction. The proposed model is evaluated with a real-world dataset from ethio telecom which exhibits remarkable growth in data demand and number of customers.

Mobile data traffic prediction can be done based on temporal or spatiotemporal levels. But it is very difficult to develop a spatiotemporal prediction model for a mobile network considering a fine granularity grid and large coverage area [9]. Developing a spatiotemporal prediction model with fine granularity grids requires an additional device to collect the user location and data traffic consumption and it may not be viable for a large city. Aside from grid-based spatiotemporal modeling, a cluster-based approach can be used to study all of the base stations deployed in a city. In the clustering method, the base stations are grouped into a certain number of clusters according to some specific properties. In this work, the eNodeBs or base stations are grouped into clusters based on the data traffic pattern and then the data traffic prediction model is proposed per cluster level.

1.2 Statement of the problem

When the number of customers and the data traffic demand increase, network capacity crunch is the main problem in mobile data service, which requires more physical resources. One way to solve this problem is via network capacity enhancement among several methods mentioned earlier. To decide the appropriate approach that suits an operator, predicting the future data demand becomes crucial. Researchers develop different types of models that are capable of capturing the dynamic characteristics of cellular data traffic. Some of the statistical models are suitable for linear, stationary datasets; however, they impose high bias [9] and are unable to model mobile data traffic with a complex pattern. Recently, many researchers leverage the ability of deep learning methods in predicting future traffic demand. Among several deep learning models, LSTM and CNN have become popular to develop mobile service demand in both temporal and spatiotemporal domains. Some researchers proposed an LSTM network to develop a model on a temporal basis while a combination of LSTM and two dimensional CNN for spatiotemporal data traffic prediction. Even if the LSTM network can handle long-term dependency problems, it can't capture important information among multivariate features. Furthermore, 2D CNN is mostly applicable for grid-based spatiotemporal data traffic prediction. Currently, 1D CNN becomes a viable option for predicting time series data due to reduced computational complexity and low hardware requirement [13]. The hybrid CNN-LSTM prediction model can be used for the time series data as well as it can solve the LSTM's problem mentioned above. In addition, some works justified that multivariate features improve model performance [14], [15], [16] because the model learns from different features which are salient for model accuracy. This research proposed a prediction model using multivariate features to evaluate the effectiveness of the 1D CNN-LSTM model for mobile data traffic. The 1D CNN is used to

select important information from multivariate features while the LSTM network which is connected in tandem with the 1D CNN network, is used to handle long-term dependency.

1.3 Objective

1.3.1 General objective

The general objective of this research is to develop cellular network data prediction model with a CNN-LSTM deep learning network using multivariate input features that capture data traffic variation.

1.3.2 Specific objectives

The specific objectives of the research are:

- To review related works from the literature.
- Understand the basics of LTE technology.
- To study LTE resource access and utilization approaches.
- Study different time series prediction methods.
- Study the characteristics of mobile data traffic.
- To preprocess the raw dataset.
- To cluster the base stations according to temporal characteristics.
- Develop CNN-LSTM prediction model for mobile data traffic.
- Compare the model performance with different prediction models.

1.4 Related works

In this section, related works have been reviewed and these papers are organized as follows considering different perspectives of prediction models.

Univariate Feature Prediction

The paper in [17] proposed a temporal prediction model to forecast ethio telecom's UMTS data traffic future demand considering univariate features using a hybrid SARIMA and Extreme Learning Machine (ELM) models. The SARIMA model is used to handle the linear part of the data traffic while the latter one is used to model the nonlinear part of the data traffic. The results show that the hybrid model improved the performance model outperforming the SARIMA model.

The authors in [5] evaluate the effectiveness of LSTM and GRU models to predict mobile data traffic demand on a temporal basis. The data is collected from Viettel telecom for the UMTS network of fifty-seven sites at an hourly level for one month consisting of calls, SMS, and internet activities. The average hourly data traffic is estimated from Internet activity that used as a comparison to prediction outputs. The models predict traffic demand for a single step of one hour and the result outperforms other models such as MLP, and Adaptive Neuro-Fuzzy Inference System (ANFIS).

The paper in [18] proposed an LSTM-based model for predicting data traffic in a mobile network, and it used 4G data collected from 573 eNodeBs for four months duration. The authors chose a sample of data with a low relative standard deviation that corresponds to a busy time in a congested cell to improve the model's performance. The proposed model is capable of capturing traffic variation in a cellular network, and the multi-step prediction output outperforms statistical methods such as ARIMA and SARIMA.

Multivariate Feature Prediction

The authors in [19] develop a cellular network prediction model using raw data gathered from the Physical Downlink Control Channel (PDCCH). A one-month data is collected for two base stations that consist of multivariate features like the number of the resource block, the number of transport blocks, and modulation and coding schemes associated

with a user. The LSTM model was used in the paper due to its ability to handle complex data and exploit the long-term dependency exhibited in the dataset. The authors compared the proposed LSTM model with the Feed Forward Neural Network (FFNN) and the ARIMA models. The result shows that the LSTM model grasp the traffic pattern very well and also it has better performance than the FFNN and ARIMA models.

The authors in [14] develop a CNN-LSTM neural network model that combines CNN and LSTM to predict the residential energy consumption based on multivariate time series data. The dataset contains multivariate features such as global active power, global reactive power, global intensity, voltage, sub-metering 1, sub-metering 2, and sub-metering 3 in addition to temporal information. The three sub-metering features correspond to different equipment in which their energy consumption varies largely. In the proposed model, the CNN model is used to extract dominant features and to remove noises from the multivariate features, and the output of the CNN model is provided to the LSTM model. This LSTM model is used for its capability of short and long-term memory. The performance of the CNN-LSTM prediction model is compared with different types of neural networks such as LSTM, GRU, Bi_LSTM using MSE, RMSE, and MAPE evaluation metrics. The result shows that the CNN-LSTM model outperforms all the above models with all evaluation metrics.

The paper in [16] also proposed a hybrid CNN-LSTM model to predict particulate matter 2.5 (PM2.5) concentration which is the most pollutant related to air quality. The model predicts the amount of PM2.5 concentrations in the city of Beijing from multivariate features such as PM2.5 concentrations, dew point, temperature, wind speed, and atmospheric pressure collected in hourly granularity. The CNN-LSTM model is selected to capture the complex relationship of those features. The data is normalized using a min-max scaler in the data preprocessing stage to improve model accuracy, and 80 percent of

the data is used for training while the remaining 20 percent is used to test the model. Because of the cyclical nature of air quality data, one-week feature values are used as input to predict the PM2.5 concentrations for the following day. The authors compare the result with the LSTM network and also perform a comparison for multivariate features and univariate feature inputs. The result demonstrates that the CNN-LSTM model with multivariate features has better performance for regression metrics of MAPE and RMSE.

Cluster-based Prediction

In their paper [20], the authors used a hybrid model consisting of statistical and deep learning models, namely double seasonality (D-SARIMA) and LSTM. The K-Means clustering is used to group base stations based on their temporal traffic patterns and investigated spatial dependency among different clusters. The prediction of mobile data traffic is performed at both the base station and cluster levels, and the hybrid model results are compared to those of the D-SARIMA and LSTM models. The dataset was collected from ethio telecom UMTS network on an hourly basis for four months' duration. The results show that the hybrid model performs well at the cluster level than at the base station level. The hybrid model also outperforms both the D-SARIMA and LSTM models.

The work in [21] assesses the effectiveness of various time series forecasting models in predicting mobile data traffic demand. The models have been examined in a dataset collected from 13,296 base stations over one week. The base stations are grouped based on their traffic load using K-Means clustering algorithms, and a prediction model is proposed for each cluster. The authors examined the ability of various prediction models, including ARIMA and SARIMA from statistical models, SVM and Decision Trees from machine learning models, and MLP, LSTM, and GRU from neural network models. The

result shows that deep learning models, particularly LSTM and GRU, outperform statistical and machine learning models in terms of performance.

To summarize the above studies, some of the papers [17] and [20] forecast mobile data traffic with a univariate time series forecasting approach using a hybrid model (statistical, machine learning, or deep learning). Those papers handle the linear and nonlinear part of the data traffic separately which makes the model complex and also the linear dataset should be stationary for better accuracy. The paper in [19] predicts mobile data traffic using multivariate features using the LSTM model, but unable to capture important information among features. The works in [14] and [16] use the CNN-LSTM model for multivariate time series data for other domains with decent results. In this work, the CNN-LSTM model is selected to develop a mobile data traffic model due to its ability to handle multivariate time series data and its suitability for multi-step prediction.

1.5 Methodology

In this thesis, the following methodology is used to develop a data traffic prediction model for the LTE network using a multivariate time series method. Data is collected from 690 eNodeBs for four months on an hourly granularity, and related works of literature are reviewed to select the appropriate type model. The dataset contains features like downlink data traffic volume, uplink data traffic volume, downlink average throughput, cell average user, number of RAB attempts, and cell maximum user in cell level with hourly temporal resolution. Identification of missing values, outliers, and noisy values as well as missing value imputation is performed in the data preprocessing part. To select the relevant features from multivariate time series data, correlation analysis is used. Furthermore, the feature scaling technique is employed to scale the features to a specific range. K-Means Clustering is used to group the base stations into a certain number of clusters based on the temporal traffic pattern of downlink data traffic.

The time series problem is organized into a supervised learning problem that enables to train and test deep learning models. In this study, a deep learning network, specifically the CNN-LSTM model, is used to develop a mobile data traffic prediction model. This method is selected due to the nonlinear nature of the dataset, suitable for a multivariate dataset, and the ability of the CNN-LSTM model to extract important features as well as long short-term memory. More importantly, deep learning networks are adequate for multi-step time series forecasting problems. The proposed model consists of two types of deep learning models 1D CNN and LSTM. Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) regression metrics are used to assess the CNN-LSTM model performance. The model is implemented with python in Jupyter Notebook using the K-Means algorithm from sklearn while CNN and LSTM algorithms from Keras backend with Tensorflow. Figure 1.2 below shows the approach that followed in the research.

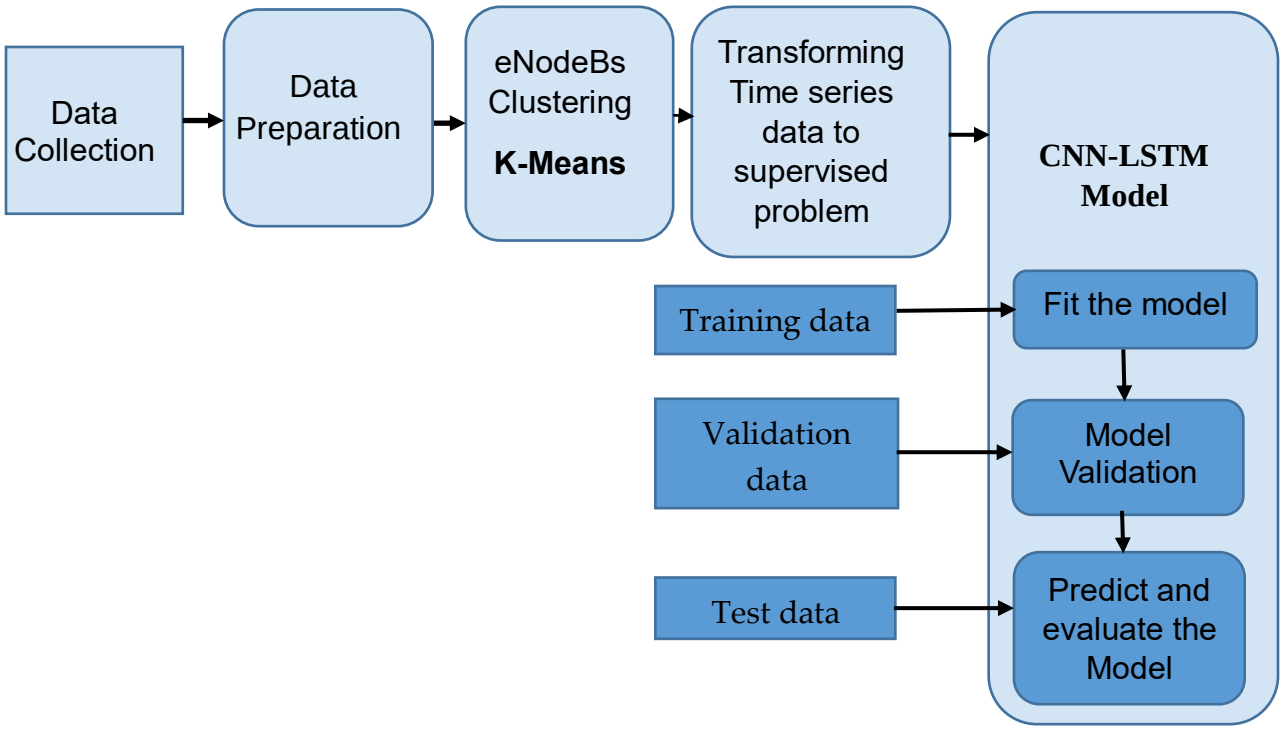


Figure 1.2 Research Methodology

1.6 Scope and Limitation

1.6.1 Scope of the thesis

The scope of the research is to develop a model that predicts the mobile data traffic considering 4G/LTE technology, and it did not include data traffic generated from UMTS, Code Division Multiple Access (CDMA), and fixed broadband networks.

1.6.2 Limitation of the thesis

The downlink (DL) and uplink (UL) data traffic volume of the LTE network are analyzed for a period of one month. The result shows that the ratio of DL to UL is 12:1 which constitutes only about 7%. For this reason, this work is limited to DL data traffic volume.

1.7 Contribution of the research

This research proposed a deep learning-based multi-step prediction model that can predict the mobile data demand considering multivariate features that influence future data traffic demand. It can be used as an input for cell-level or site-level optimization, as well as to select optimal timing to upgrade the network.

1.8 Outline of Thesis

The rest of the paper is organized as follows. Chapter Two describes the basics of LTE/4G technology and concepts associated with LTE resources. In Chapter Three, the characteristics of mobile data traffic, the basics of deep neural networks and some deep learning models used for data traffic prediction are discussed. In Chapter Four, data preprocessing tasks including the K-Means clustering result are presented. The model and tools used for the experiment as well as the metrics used to evaluate the model performance are well defined. Chapter Five presents the result and discussion part for mobile data traffic prediction models. Finally, the conclusion of the study and future work is presented in Chapter Six.

2. LTE Network Overview

A cellular network is evolving continuously and each evolution brought new services. Starting from digital cellular networks, 2G technology is used to provide voice calls and SMS service while it becomes possible to use broadband Internet with a phone in 3G [22]. LTE, a pre-4G technology, is becoming suitable for services requiring high data rates and low latency. It evolved from 3G technology, with changes such as all IP packet networks and FDMA radio technology replacing 3G's CDMA radio technology. International Telecommunication Union Radio communication (ITU-R) set requirements called IMT-Advanced in which a mobile phone and Internet access service should fulfill to be considered as 4G. Those performances related to IMT-Advanced requirements are listed in Table 2.1 below.

Table 2.1 IMT-Advanced Requirements Metrics Values [23].

Performance Metrics	**Downlink**	**Uplink**
Peak data rate	1 Gbps	1 Gbps
Spectral efficiency	15 bps/Hz	6.75 bps/Hz
Bandwidth	Scalable, a minimum of 40 MHz	Scalable, a minimum of 40 MHz
Latency	10 ms for control plane 100 ms for user plane	

The LTE network does not fulfill such requirements and it is marked as 3.9G but LTE Advanced satisfies these IMT advanced requirements and is considered as 4G technology. Some of the key enabling technologies that enable to meet the desired performance are the use of Orthogonal Frequency Multiple Access (OFDMA) and Multi-Input Multi-Output (MIMO).

2.1 LTE Network Architecture

The Evolved Packet System (EPS) is an all-IP packet-based LTE network architecture that consists of three main components: the UE, the E-UTRAN, and the EPC, as illustrated in figure 2.1 below. These three entities are linked by various interfaces and communicate using different protocols. EPC includes a variety of network elements that are used to route signaling and user data traffic via the control plane and user plane, respectively. E-UTRAN constitutes several enhanced NodeBs (eNodeBs) that manage radio resources and user mobility [15], whereas the UE is a device that enables a user to access LTE network services.

2.1.1 Evolved Packet Core (EPC)

EPC incorporates core network elements that include Mobility Management (MME), Home Subscription Server (HSS), Policy Control and Charging Rules Function (PCRF), Serving Gateway (S-GW), and Packet Gateway (P-GW) entities and those components are split into either control plane or user plane entity.

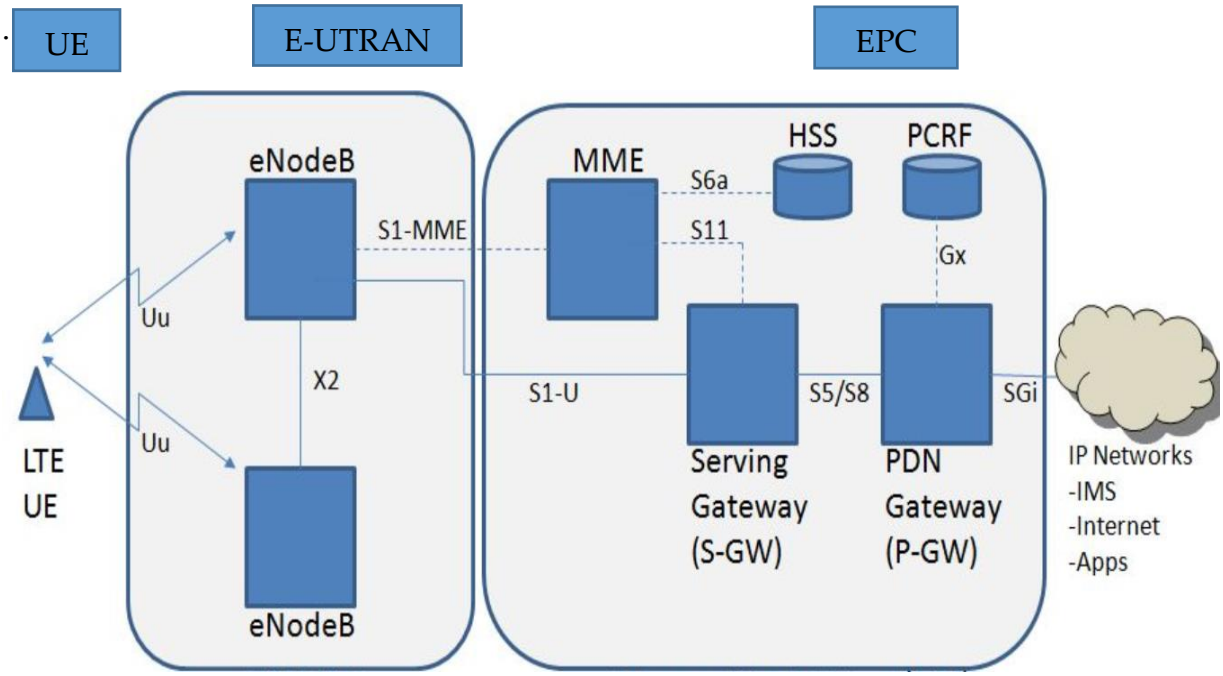


Figure 2.1 3GPP LTE network Architecture [24].

The user plane and control plane, which are responsible for carrying user data traffic and signaling respectively, are separated in EPC, allowing the component to scale independently. Because of this functional separation, operators can smoothly dimension and adapt their network elements. The function of each element in EPC is described below [25]:

MME: MME performs functions that include signaling and security control, tracking area management, inter-core network signaling for mobility between 3GPP access networks, EPS bearer management, and roaming and authentication.

P-GW: The P-GW handles tasks like IP address allocation, packet filtering, and policy enforcement, as well as user-plane anchoring for mobility between 3GPP access networks, allowing the EPC to connect to external IP networks.

S-GW: The S-GW routes and forwards user data packets, as well as maps transport-level service quality. It serves as an interface module for signaling between PGW and MME, and SGW functions as an IP router with GTP support and charging capabilities.

HSS: HSS contains user profile information and authenticates users inside the EPS network.

PCRF: PCRF is the control plane node that manages the QoS inside the EPS network and performs pricing actions [26]**.**

2.1.2 Evolved Universal Terrestrial Radio Access Network (E-UTRAN)

The E-UTRAN comprises evolved NodeBs (eNodeB) that make it different from its predecessor (UMTS, HSPDA and HSDPA+) which contains NodeBs and RNC. The RNC network element is no longer used in LTE, and its functionality has been transferred to eNodeB. The eNodeB accomplishes the functions of NodeB (3G base station) along with protocols applied in the Radio Network Controller (RNC). The eNodeB performs

functions like ciphering, reliable delivery of packets, and header compression. The eNodeB also carries out functions such as radio resource management and routing user plane data towards System Architecture Evolution (SAE) Gateway in the control plane side. Some of the benefits of reducing the access network node to one in LTE network architecture include minimized network latency and processing load distribution among many eNodeBs [27].

2.1.3 User Equipment

The UE is an end-user communication device that contains two distinct elements; the USIM (Universal Subscriber Identity Module) and the ME (Mobile Equipment) [28]. The USIM is used to identify and authenticate users, as well as to acquire security keys for safeguarding the radio interface, whereas the ME is responsible for radio transmission and application storage. The mobile equipment (ME) is further sub-divided into Mobile Termination (MT) and Terminal Equipment (TE) entities. The mobile termination entity is in charge of radio transmission and related tasks, whereas the terminal equipment entity encompasses end-to-end applications like a laptop connected to a mobile phone [29].

2.1.4 LTE Network Interfaces and Protocols

LTE is a flat architecture and contains fewer network nodes, as a result, the complexity of the network is reduced. Each of the radio and core network elements found in LTE network architecture is inter-connected by standardized interfaces that enable interoperability between different vendor products. This compatibility benefits telecom operators to choose network elements from separate vendors. The different types of interfaces in EPS are described below [27].

i. Uu Interface: It is a radio link that connects the eNodeB to the UE and handles all communication between the UE and the LTE network.
ii. S1 Interface: It is the link between E-UTRAN and the EPC, carrying both signaling and user data.
iii. X2 Interface: It connects multiple eNodeBs that enable them to coordinate with each other and enables resource sharing.
iv. S6a Interface: This interface connects MME and HSS that enables the MME to access subscriber-related information and authentication of the user data.
v. S11 Interface: It is an interface between the MME and S-GW based on the Gn interface (an interface in UMTS) with some additional functions for paging coordination and mobility.
vi. S5/S8 interface: The S5/S8 interface links the network elements S-GW and P-GW. When the UE is registered for roaming between different telecom operators, the S8 interface is used, whereas the S5 interface is used if the UE is not subscribed to the roaming service.
vii. SGi: It is the reference point between the PDN GW and the public or private packet data network.

2.2 LTE Access Technology and Resource Allocation

Telecom operators deploy cellular networks capable of providing different services such as voice and data services, which necessitate the use of dedicated resources. Especially, sufficient radio resources are essential to provide data service with high data rates while retaining the required QoS. Different technologies employ various types of access as well as resource allocation mechanisms for efficient utilization of resources. LTE access technology, resource blocks, and traffic classes distinguished during the RAB setup procedure are described below.

2.2.1 LTE Network Access Technology

LTE employs flexible FDD and TDD methods that allow utilization of various channel bandwidths. Possible bandwidths that can be used in LTE are 1.4,3,5,10,15, and 20 MHz [30]. LTE network uses OFDMA in downlink and SC-FDMA in uplink. In the downlink, OFDM divides the bandwidth into many small subcarriers spaced at 15 kHz and assigns each user the bandwidth needed for their transmission. Unutilized subcarriers will be turned off, allowing for power consumption reduction as well as interference mitigation. LTE uses SC-FDMA access technology in the uplink, which has a lower PAPR (Peak-to-Average Power Ratio) than OFDM. This low PAPR consumes less battery power, necessitates a simpler amplifier design, and improves uplink coverage and cell-edge performance. In contrast to OFDMA, where each subcarrier transports unique data, data in SC-FDMA is spread across multiple subcarriers.

2.2.2 LTE Resource block

LTE uses frame and subframe methods like previous cellular technologies to transmits data. Frame and subframe approaches are used for synchronization and efficient data delivery. In LTE, a 10 ms frame is made up of 10 subframes, each of which is divided into two slots [16]. The smallest modulation unit in LTE is the resource element that comprises one 15 kHz subcarrier by one symbol. Aggregated resource elements form resource blocks. A resource block has dimensions of subcarriers by symbols. The physical resource block in the LTE network is illustrated in Figure 2.2 in which the resource block is subdivided into the time and frequency domain. One resource block constitutes seven symbols in the time domain and 12 subcarriers in the frequency domain. A minimum of one resource block is required for each user for data transmission. The downlink and uplink resource allocation differ depending on the use of FDD or TDD spectrum usage techniques. In FDD, the uplink and downlink resource allocations are separated by

frequency; however, in TDD, the resource blocks are transmitted at the same frequency but at different time intervals.

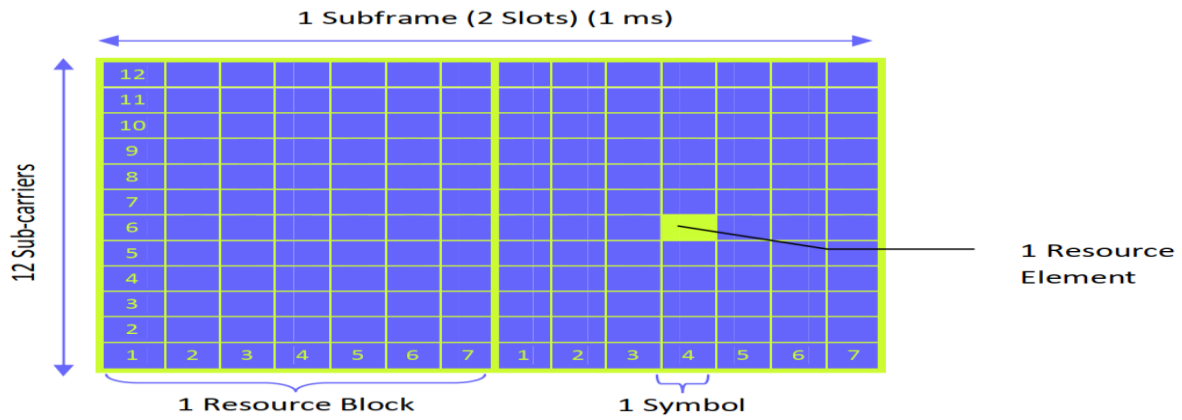


Figure 2.2 LTE resource block and resource element [31]

2.2.3 Radio Access Bearer (RAB)

Radio access bearer is an E-UTRAN resource dedicated to transporting user data for specific connections across the E-UTRAN network, and it can be thought of as a service provided by the E-UTRAN to the core network and UE. The RAB service is defined by a set of attributes such as traffic class, maximum bit rate, guaranteed bit rate, and maximum source data unit (SDU) size, which define the traffic or QoS profile of a particular application or service [32]. Those different attributes of RAB are described below.

Traffic Classes: The various traffic classes incorporate conversational, streaming, interactive, and background and they are distinguished primarily by delay sensitivity.

Maximum Bitrate: It is the maximum data rate that a user or application can transmit.

Guaranteed Bitrate: It is defined as the number of guaranteed bits given by a service access point divided by the period's duration.

Maximum SDU Size: It specifies the maximum permissible SDU size and it is used for optimizing transport data as well as admission control and policing.

3. Mobile Data Traffic Prediction and Deep Learning Basics

In this chapter, an overview of mobile data traffic considering temporal and spatial characteristics is discussed. And also a brief description of deep neural networks, that are used to build a predictive model for mobile data traffic, is presented.

3.1 Mobile Data Traffic Characteristics and Prediction

The characteristic of mobile data traffic is dynamic due to users' usage behavior, the prevalence of smartphones, availability of different applications that require various data rates, and the emergency of the Internet of thing (IoT). Mobile data traffic predictive models become imperative for Mobile Network Operators (MNOs) to capture the dynamic property of the network. Using mobile data traffic predictive models helps MNOs for planning and efficient resource utilization such as spectrum and power. Especially in LTE technology, the use of the OFDMA method enables to switch off the idle resource blocks, and this approach can be enhanced using accurate mobile data traffic prediction models.

3.1.1 Mobile Data Traffic Characteristics

Mobile data traffic exhibits different properties in both time and spatial domains. Trend and seasonality are used to demonstrate the temporal properties of time-series data. In time-series data, a trend is a long-term increase or decrease, whereas seasonality is a repeating pattern with a fixed period such as daily, weekly and yearly. Figure 3.1 below illustrates the data traffic pattern for a sample eNodeBs for a duration of nine days. Even if the average daily traffic differs for different days, there is a rising trend between 4:00 AM and 8:00 PM. The data traffic expectedly declines after 8:00 PM to 4:00 AM with a downward trend. Due to peak hours, more data traffic is generated around 10:00 AM in the daytime and 8:00 PM during the night. The figure also shows the daily seasonality

that exists in data traffic in which the data traffic load becomes low around 4:00 AM and becomes significantly high around 8:00 PM every day. It also exhibits weekly seasonality as the data traffic load repeats after a week (for days 13 and 20).

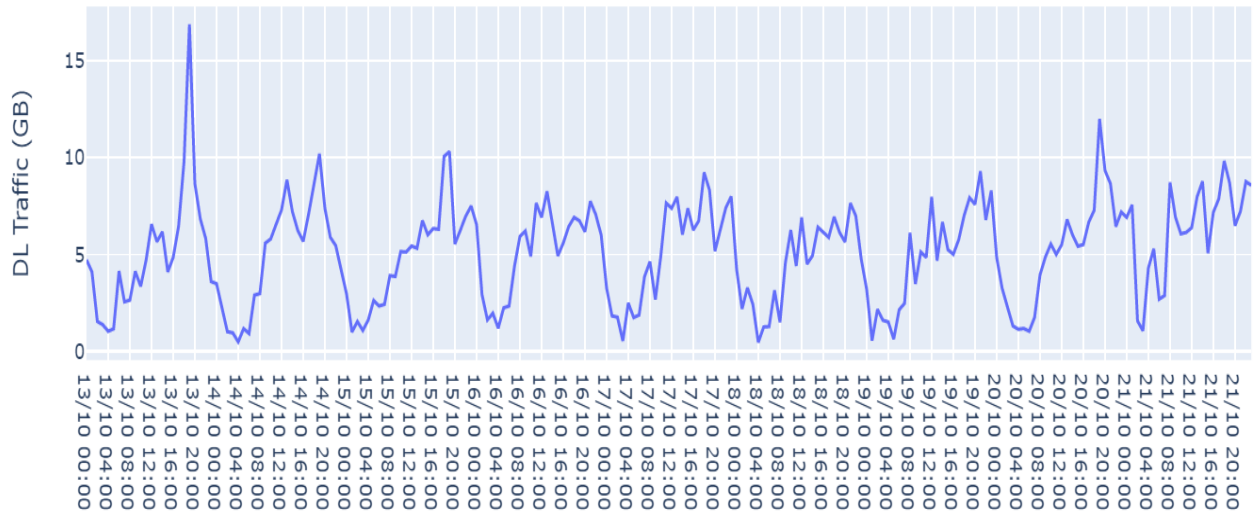


Figure 3.1 One-week data traffic for sample LTE site

Apart from temporal characteristics, investigating spatial property is important to analyze the data traffic of base stations located in different areas. Even though the data traffic patterns differ for areas such as residential, business, and entertainment areas, some base stations located remotely exhibit the same data traffic pattern. Figures 3.2 and 3.3 below demonstrate the data traffic load for eNodeBs located at different locations. The temporary properties of the sites related to average daily traffic, trend, and seasonality are almost the same. Spatial characteristics of mobile data traffic can be analyzed by selecting a certain number of base stations by forming a grid or using the clustering method. Identifying those LTE data traffic characteristics assists MNOs in optimizing the network by applying pool resource allocation systems.

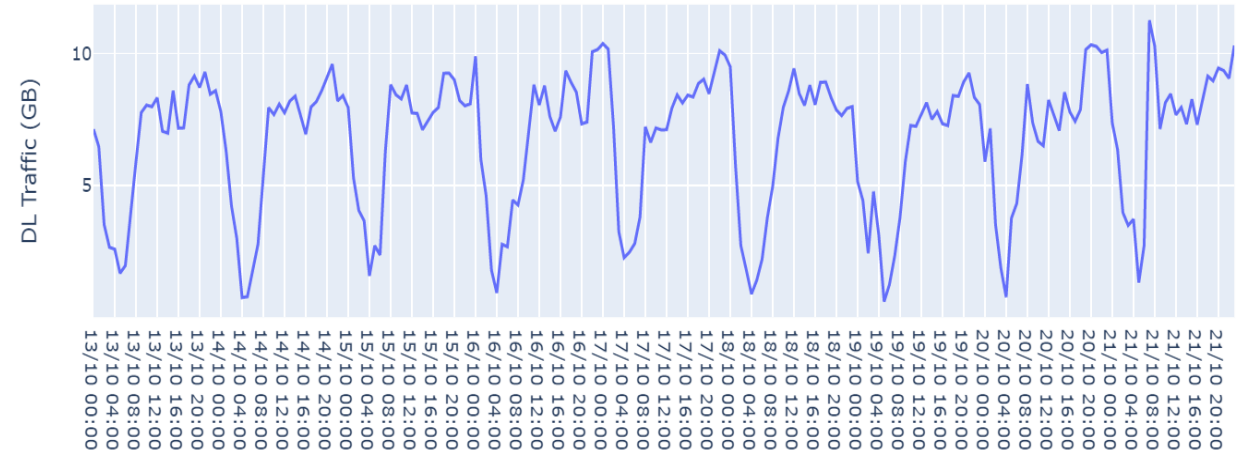


Figure 3.2 Data traffic pattern for sample LTE site at location A

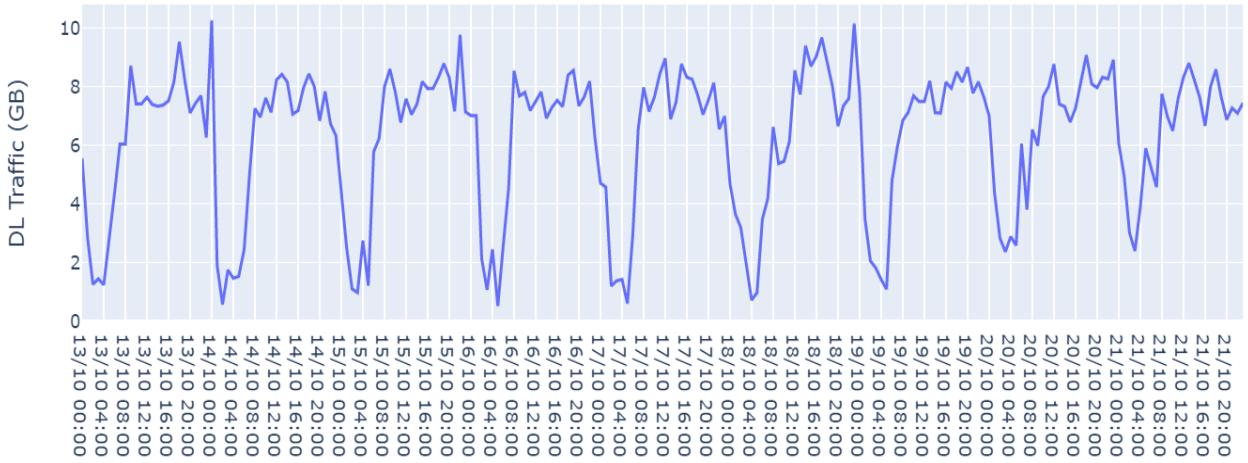


Figure 3.3 Data traffic pattern for sample LTE site at location B

3.1.2 Mobile Data Traffic Prediction

Mobile data traffic prediction models could be developed in temporal and Spatio-temporal domains. Those approaches can be realized with statistical, machine learning, or state-of-the-art deep learning-based methods. The following section describes data traffic prediction considering temporal and Spatio-temporal domains as well as cluster-based prediction methods.

3.1.2.1 Temporal Data Traffic Prediction Model

The temporal data traffic prediction model forecasts future data traffic demand based on the previous values and the current value of data traffic considering only the time domain. It can be based on univariate or multivariate features.

In univariate prediction, only a single variable is measured over time neglecting the effects of other variables. The model can be a single-step or multi-step model, and the equations below describe the formulation for such models respectively.

$$\hat{y}_{t+1} = f_1(y_t, y_{t-1}, \dots, y_{t-M+1}) \quad (1)$$

$$\hat{y}_{t+k} = f_k(y_t, y_{t-1}, \dots, y_{t-M+1}) \quad (2)$$

where $\hat{y}_{t+1}$ - predicted value at time t+1;

$\hat{y}_{t+k}$ - predicted value at t+k;

k - number of future time steps;

M+1 - the number of lag observations.

Only the next step value is forecasted in single-step prediction, whereas the multi-step prediction model provides an output for more than one time step, and they are referred to as short-term and long-term prediction models respectively.

In contrast to univariate, multivariate times series forecasting use more than one variable to forecast the desired parameter (data traffic volume). This method considers the effects of other variables while developing a forecasting model. Like the univariate time series forecasting method, the multivariate time series prediction models can be a single step or multi-step as expressed in the equation below.

$$\hat{y}_{t+1} = f_1(X_t, X_{t-1}, \dots \dots X_{t-M+1}) \quad (3)$$

$$\hat{y}_{t+k} = f_k(X_t, X_{t-1}, \dots \dots X_{t-M+1}) \quad (4)$$

where

k – number of the future prediction values;

M+1 - the number of lag observations;

y_t & $\hat{y}_{t+1}$ - actual & predicted values;

X= $\{x_1, x_2, \ldots . x_n\}$, n is the number of features.

3.1.2.2 Spatio-Temporal Data Traffic Prediction Model

The spatiotemporal data traffic prediction approach considers spatial correlation in addition to temporal dependencies noticed in mobile data traffic to predict the traffic demand for a certain geographical area in a given time. Since mobile users constantly move within a given cellular network, the traffic pattern across neighboring base stations are correlated or complemented, such that developing in both the spatial and temporal dimensions would provide better information for telecom operators [33]. Spatiotemporal data traffic prediction incorporates different user behavior such as mobility and network behavior like the number of handovers in the network [34].

3.1.2.3 Cluster-based Data Traffic Prediction

In mobile data traffic prediction, cluster-based prediction is to group the eNodBs with similar traffic load together and those base stations within the same cluster have similar characteristics. The base stations can be clustered based on either geographical location or temporal behavior [21]. Clustering base stations based on their geographical area is referred to as spatial clustering. The assumption in spatial clustering is that neighboring base stations exhibit similar temporal properties. In temporal-based clustering, the clustering is done based on temporal behavior irrespective of geographical location. Considering more than one base station in time series clustering incorporates the spatial

information of the data traffic. After clustering the base stations, the data traffic prediction model is developed per cluster level.

3.2 Deep Learning Basics

Artificial Intelligence (AI) is a field of study that aims to make machines intelligent by programming them to learn, think, and imitate the actions of people and other animals [35]. On the other hand, Machine Learning (ML) is a subset of artificial intelligence in which computers use algorithms to analyze, comprehend, and identify patterns in data. Machine learning is capable of making decisions with little or without human intervention. Machine learning algorithms are classified into three types: supervised, unsupervised, and reinforcement. In supervised learning, the model is trained on a labeled dataset that incorporates both input data and its outcome, whereas unsupervised learning algorithms use unlabeled data and search for a concealed structure and pattern to predict the data point to a specific group. The agent can assess and interact with its environment in the reinforcement technique, which allows it to take actions by trial and error based on feedback from its experiences [36].

An artificial neural network (ANN) is a computational system modeled after biological neural networks in terms of structure, processing method, and learning ability. Similar to the human brain that has neurons interconnected to one another, artificial neural networks also have neurons that are interconnected to one another in several layers of the networks called nodes. ANN has three layers, input layer, hidden layer, and output layer as illustrates in Figure 3.4 below. The input layer in an artificial neural network accepts features and transfers them to a hidden layer without conducting any computations. Hidden layers, on the other hand, perform computations on the features conveyed from the input layer and send the results to the output layer. The output layer

takes input from the neighboring hidden layer and uses an activation function to compute and deliver a result.

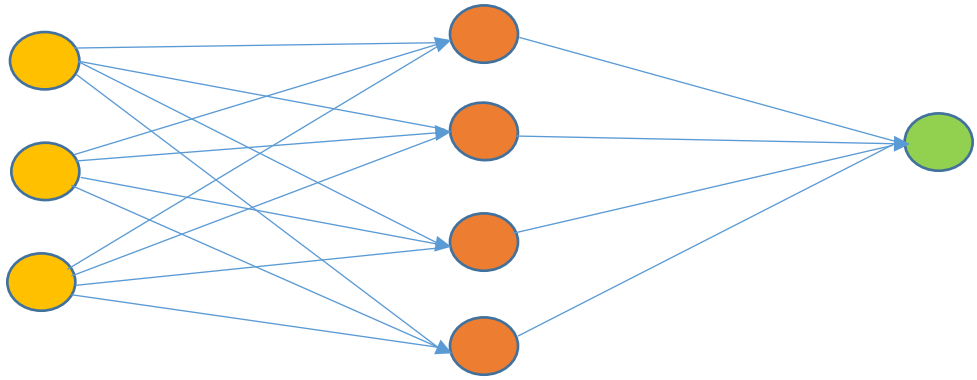


Figure 3.4 A schematic diagram of neural network architecture [37]

A neural network with three or more layers is called a deep neural network. Deep learning is capable of learning from unlabeled data, identifying significant features, and solving nonlinear and complex modeling problems.

The following terminology and concept are necessary to understand deep learning-based prediction models.

a) Weights - weights are numeric values that are multiplied with inputs and used to control the strength of the connection between two neurons.
b) Bias - Bias are constants terms that are added to the weighted input before the activation function is applied in each neuron.
c) Activation function **-** Activation function is an internal state of neurons used to convert the weighted sum of input features on neural network nodes to an output value. The commonly used activation function includes sigmoid, tanh, and ReLU.
d) Loss function - While applying an optimization algorithm, the function used to evaluate the candidate solution is referred to as the objective function. To find the

best solution, we can either maximize or minimize this objective function. It is referred to as a loss or cost function when the objective function is minimized through optimization.

e) Optimization algorithm - Optimization algorithms update weights and learning rates in a neural network to minimize error or losses. Some of the optimization algorithms are SGD, RMSProp, and Adam.

f) Forward propagation - In a neural network, the output of a given hidden layer is computed using inputs and parameters, and this output is propagated to the next hidden layer as an input. Applying this method enables to compute the error at the output layer called forward propagation.

g) Backward propagation - Forward propagation method is used to obtain the error using a loss function that is the difference between the final output and target value. To minimize the losses, the errors are back propagated to the earlier layers to compute the gradient, and this process is called backward propagation.

h) Vanishing and exploding gradient - During the backward propagation process, when we move backward starting from the output layer, the gradient value may be too small or too large, resulting in vanishing and exploding gradient. Exploding gradient causes instability during model training. In vanishing gradient, the earlier layers will not get a new update, and the model experiences difficulties in optimizing those layers.

3.3 Deep Learning for Mobile Data Traffic Forecasting

Deep learning models such as Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), and Convolutional Neural Network(CNN) are becoming popular in dealing with time-series data such as text, speech, and image recognition [38]. The basics of LSTM and

CNN networks that are used to develop the proposed model are discussed in the following subsection.

3.3.1 Recurrent Neural Network Model

Recurrent neural networks (RNN) are neural networks modeled to deal with sequential data [39] in which the preceding layer's output is fed as an input to the next layer, allowing the network to capture the dependency of sequential data. LSTM is a type of RNN network built as a solution for short-term dependency problems as well as to resolve exploding and vanishing gradient problems. LSTM network has three gates that decide which information is to add or remove from the cell state, and the cell state memory stores the desired information. The function of the gates and the cell state memory in the LSTM is described below.

Forget gate: Before forwarding to the cell state, the forget gate selects whether information should be maintained or throw away from the preceding hidden layer and the current input layer.

Input gate: This gate uses the sigmoid function to process inputs from the previous hidden state and current input layer, and outputs values between 0 and 1.

Cell state: The cell state stores the desired information and forwards it to the next hidden layer.

Output gate: Based on the information from the current input and the preceding hidden states, the output gate determines the value of the next hidden layer. Cell state memory is used to retrieve information about previous hidden states, and the structure of the LSTM network is depicted in figure 3.5 below.

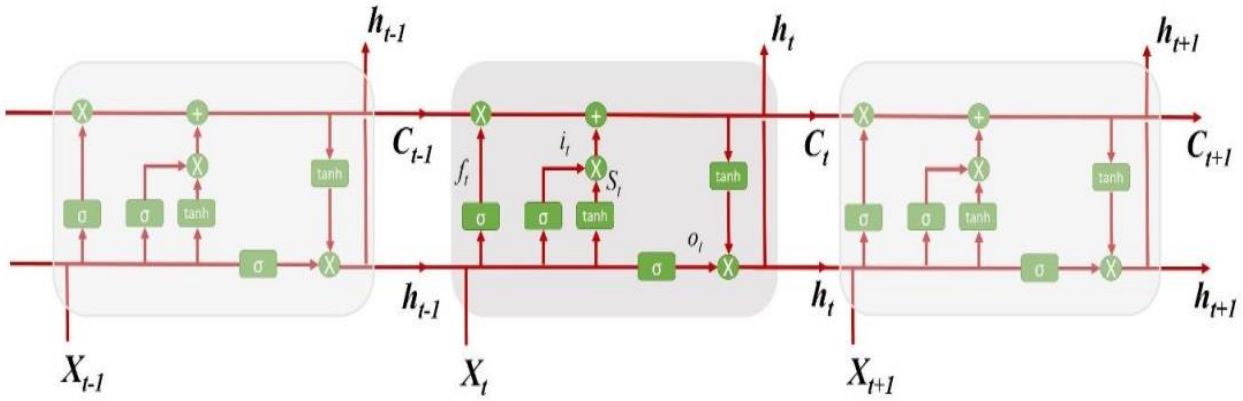


Figure 3.5 The structure of the LSTM network [38].

The mathematical expression for the LSTM network at time t is described as follows

$$f_t = \sigma(W_f \,.X_t + U_f \,.h_{t-1} + b_f) \quad (3.1)$$

$$i_t = \sigma(W_i.X_t + U_i.h_{t-1} + b_i) \quad (3.2)$$

$$S_t = \tanh(W_c \,.X_t + U_c \,.h_{t-1} + b_c\,) \quad (3.3)$$

$$C_t = \; i_t * S_t + f_t * \; S_{t-1}) \quad (3.4)$$

$$o_t = \sigma(W_o \,.X_t + U_o \,.h_{t-1} + V_o \,.C_t + b_o\,) \quad (3.5)$$

$$h_t = \; o_t * \tanh(C_t) \quad (3.6)$$

Where

- tanh and σ are activation functions
- i_t, f_t, and o_t are values of the input gate, the forget gate, and the output gate at time t
- b_i, b_f, b_c and b_o are bias vectors for input gate, forget gate, cell state and output gate respective
- X_t is the input vector to the memory cell at time t
- $W_f\,, W_i\,, W_c\,, W_o\,, U_f\,, U_i\,, U_c\,, U_o$ and V_o are weight matrices for gates and cell state

3.3.2 Convolutional Neural Network (CNN) Model

CNN models are adequate for processing data that are in the form of multi-dimensional arrays. Many data modalities are represented as multiple arrays, with 1D being used for signals, sequences, and language processing, 2D for images or audio spectrograms, and 3D suited for video or volumetric images [40]. Recently, the CNN model becomes prominent for handling time series data (1D) apart from 2D and 3D data.

3.3.2.1 2D Convolution Neural Network

The CNN network architecture comprises of convolutional layer, pooling layer, and fully connected (FC) as shown in figure 3.6 below.

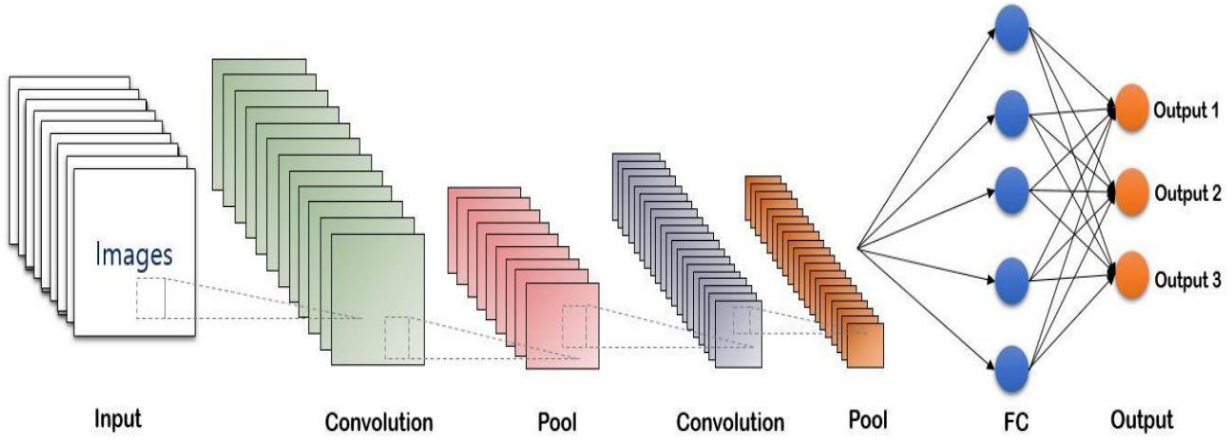


Figure 3.6 Convolutional Neural Network structure [38]

The CNN model's convolutional layer captures essential features from the input data, while the pooling layer minimizes computing complexity by down sampling the feature. Flattering is performed by the fully connected layer, which turns the data into a single vector that could be utilized as the input data for the next layer. The number of filters, kernel size, stride size, and padding are all important hyper parameters to consider when building a CNN model. The filter extracts salient features from the data, and a Kernel is a filter that specifies the width and the height of the filter. Stride specifies the amount of movement after each convolutional operation while the padding is a parameter that

resolves border problems. Without padding, the convolution operation is performed only once for the features found in the border.

3.3.2.2 1D Convolutional Neural Network

Although CNN models are typically used to analyze spatial or multi-dimensional data, the 1D CNN model is also used for more general data types, including texts and other time-series data [41]. The 1D CNN can extract salient and representative features of 1D time-series sequence data through performing 1D convolution operations using multiple filters [42].

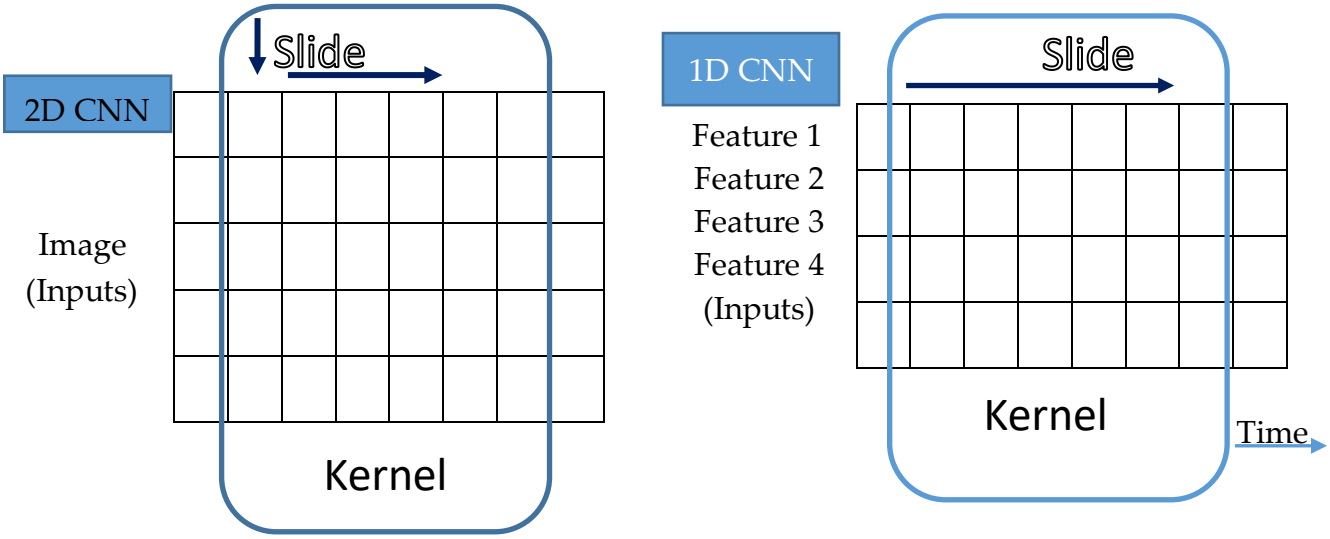


Figure 3.7 2D and 1D CNN model input type and kernel slide direction [42]

Figure 3.7 above shows the comparison of 2D and 1D CNN input types and how the kernel is slide during convolutional operation. In 2D CNN, the input is two dimensional that should be in the form of an image, and the kernel moves in two directions during performing the convolution with each feature while in the case of 1D CNN, the inputs are the number of features along with temporal property and the kernel moves only in one direction. The mathematical expression for the 1D CNN model is described below.

$x_i^0 = \{x_1, x_2, x_3, \ldots, x_n\}$ denotes the number of time steps in the per window size

The output in the convolutional layer for each feature map is expressed as

$$y_{ij}^{l} = \sigma(b_j^l + \sum_{m=1}^{M} w_{m,j}^{l}) \quad (3.7)$$

where

σ - activation function

b_j^l – biases at l^{th} convolutional layer for j^{th} feature

y_{ij}^{l} _ output at l^{th} convolutional layer for row i and feature j

m – the index value of a filter

w - weights

The feature map from pooling layer using applying max-pooling layer also expressed as

$$p_{ij}^{l} = \max_{r \in R} y_{i*T+r,j}^{l-1} \quad (3.8)$$

where

p_{ij}^{l} – is the output of the pooling at l^{th} layer

R – the size of the pooling

T – the stride value

3.3.3 CNN-LSTM Model

The capability of the CNN model to automatically learn and extract features from raw sequence data was discussed in the previous section. It is possible to combine this capability of the CNN model with the LSTM model. The LSTM network captures long-term and short-term dependency of temporal features more efficiently. The CNN model accepts input data sequences and extracts salient feature information, while the LSTM model connected in tandem interprets and provides an output. This combination of CNN and LSTM models is called a CNN-LSTM model [43].

4. CNN-LSTM Model and Data Preparation

In this Chapter, the model and tools used in the experiment are well defined and described, and also tasks associated with data preparation are briefly discussed. Furthermore, the metrics used to evaluate the model performance are explained, and the results from K-Means clustering are also discussed as a part of the data preprocessing.

4.1 Proposed CNN–LSTM Model

The proposed model consists of 1D CNN and LSTM networks that are connected in tandem as shown in figure 4.1 below.

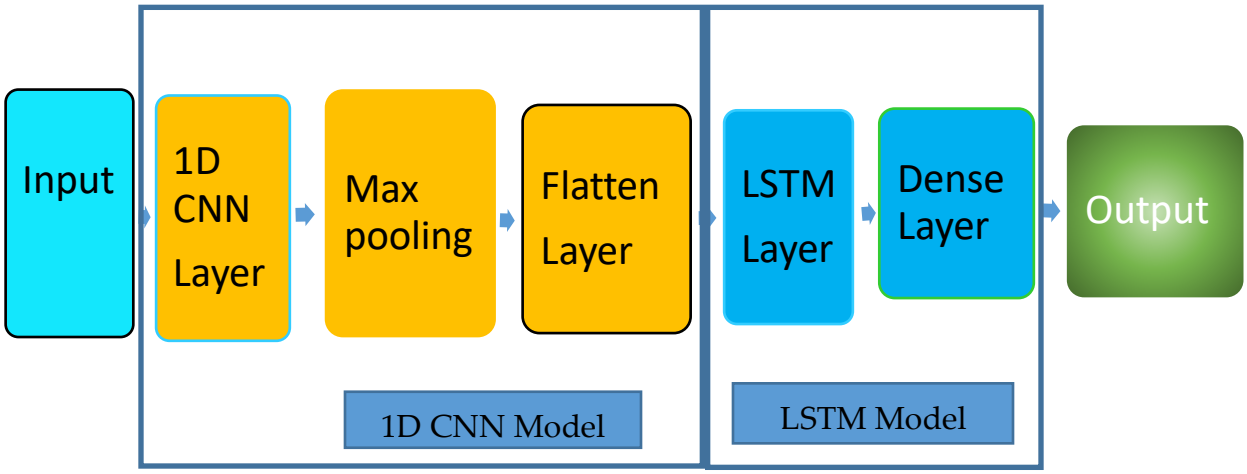


Figure 4.1 Proposed System Model

Figure 4.1 above illustrates the high-level architecture of the CNN-LSTM model that incorporates 1D CNN and LSTM models, and different parameters are set out for each model, including input and output layers. The parameters that need to be determined in the 1D CNN model include the number of filters, kernel size, the value of stride, and max-pooling layer size. Hyperparameters that refer to the LSTM model are the number of nodes and dense layers.

4.2 Dataset Preparation

Data preparation in developing a model includes data selection, data preprocessing, and data transformation [44]. Among these data preparation tasks, data preprocessing is crucial as most of the real-world data are incomplete and not consistent, contain missing values, errors, and outliers. Data preprocessing techniques improve the quality of the data, thus help to enhance the accuracy and efficiency of the resulting data model [45]. The data preparation approach used in the research is explained here.

4.2.1 Data Selection

For this study, the dataset is collected from ethio telecom's performance report system (PRS) for a duration of four months in hourly granularity for 690 LTE eNodeBs. The dataset contains multivariate features such as downlink data traffic volume, uplink data traffic volume, cell average users, cell average throughput, cell maximum user, and the number of radio access bearer related information. In addition, the latitude and longitude of the sites are collected as a separate dataset. Among several features four features namely DL traffic, UL traffic, Cell average user, and RAB success rate are chosen.

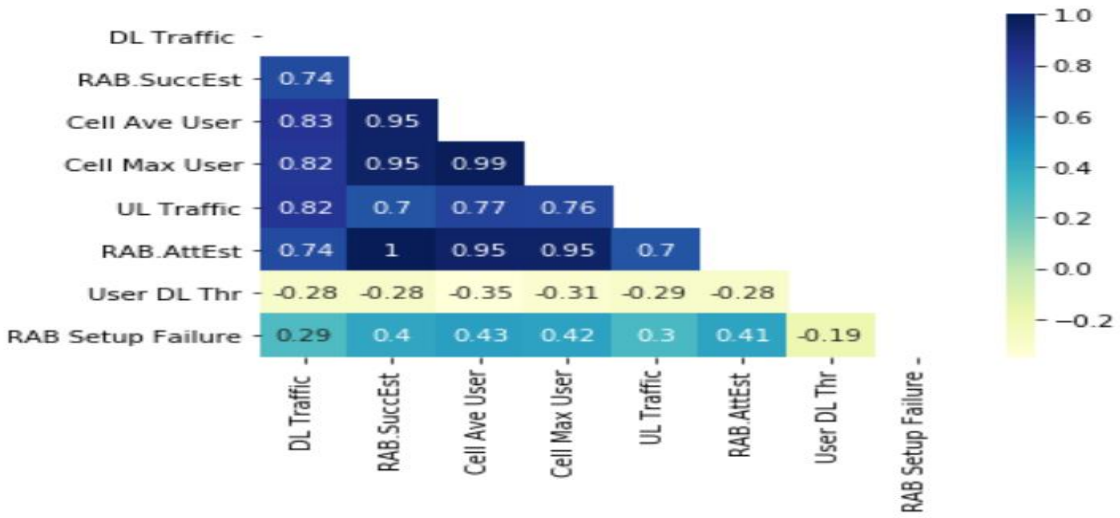


Figure 4.2 Feature correlation results

These features are selected based on the correlation analysis result as illustrated in Figure 4.2 above and they are highly correlated to downlink data traffic as compared to others.

4.2.2 Data Preprocessing

1) Data cleaning

Data cleaning during data preprocessing entails tasks such as imputing missing values as well as handling noisy data and outliers in the dataset. When working with operational data, there are two common approaches to dealing with missing numbers. The first method is to remove data samples with missing values, which is appropriate if the proportion of missing values is low. The other option is to use missing value imputation to fill missing values. For time-series data, discarding the missing data is not applicable since the data should preserve the temporal. The following steps are followed to impute the missing value of the data set.

a) Identifying types of missing value

When we analyze the dataset, there are three types of missing values in the dataset as illustrated in Table 4.1 below. The first type of missing value is a value with N/A, which is easy to regard as a missing value; the second type is a value with zero value records, which corresponds to either transmission failure or power outage; and the last type is a missing value due to missing dates or time in the dataset, that must be filled to maintain temporary sequence. In addition to identifying the missing values for each cell, the percentage of the missing values for 690 LTE eNodeBs is investigated, and the result is depicted in Table 4.2 below. From the total base stations, 443 sites had no missing values, while 15 sites have a missing value of more than 45 percent.

Table 4.1 Types of missing value in multivariate dataset

Date Time	DL(MB)	UL(MB)	User Thr	CAU	CMU	RAB A	RAB S	RAB F
10/10/2020 00:00	68.784	7.752	18.559	0.9356	3	73	73	0
10/10/2020 01:00	46.492	4.332	19.8345	0.6728	2	52	52	0
10/10/2020 04:00	3.182	0.881	8.6606	0.2625	2	29	29	0
10/10/2020 05:00	**N/A**	**N/A**	**N/A**	**N/A**	**N/A**	**N/A**	**N/A**	**N/A**
10/10/2020 06:00	256.3	14.305	21.741	1.0894	5	68	68	0
10/10/2020 07:00	197.887	20.266	16.1825	1.7028	6	68	68	0
10/10/2020 11:00	1.403	0.535	7.2378	0.5469	4	66	66	0
10/10/2020 12:00	36.955	5.757	12.9871	1.6512	6	97	97	0
10/10/2020 13:00	663.996	38.385	53.5658	1.1222	5	83	83	0
10/10/2020 14:00	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
10/10/2020 15:00	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

These 15 sites are excluded from this study because it is extremely difficult to fill the missing value while maintaining daily and weekly seasonality.

Table 4.2 Percentage of missing value for each eNodeB

	Percentage of missing value in the dataset					Total sites
	0	1-4	5-8	9-14	Above 45	
Number of Sites	**443**	**75**	**149**	**8**	**15**	**690**

b) Missing data imputation

There are different imputation techniques for imputing missing values and they can be generally categorized as statistical and model-based methods [46]. Statistical methods include replacing the missing value with mean, mode, or median but such method imposes high bias because the newly imputed data are the same as the mean of the observed data. Some of the model-based methods include autoregressive, support machine vectors, K-nearest neighbors, linear regression, and interpolation. Among

available alternatives, interpolation and autoregressive methods are selected for the main reasons, as the interpolation method is adequate if the missing values are small, and the adjacent time values don't change suddenly [47].

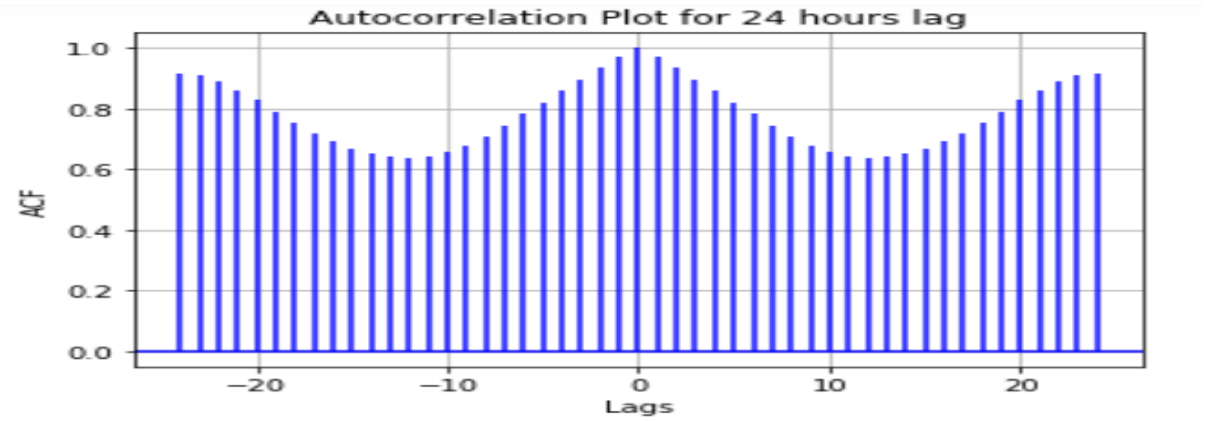


Figure 4.3 Autocorrelation for 24 hours lag values sample LTE eNodeB

An autoregressive method can be applied to impute missing values if the dataset exhibits seasonality. Figure 4.3 above demonstrates the autocorrelation function plot of downlink data traffic for 24 hours lags, and the result shows the presence of daily correlation as the Auto Correlation Function (ACF) value becomes peak after 24 hours' lag. In this case, the autoregressive method is appropriate to impute values if the missing values occur for consecutive hours (days).

The missing value for each cell is imputed using the aforementioned techniques, and then the data traffic is aggregated at the site level. Some sites that have a missing value for two consecutive weeks are removed from the study because it is difficult to impute the missing value preserving the seasonality of the data traffic. Further checking the data traffic volume of the sites, additional 17 sites are not considered in this research due to low data traffic (less than 100 GB per month) that has an average value of about 1200 GB.

c) Outliers detection

Outliers are data points that differ significantly from the rest of the data points in the dataset. Those outliers found in the data cause inconsistencies that can lower the model's performance. As a result, identifying and eliminating outliers is crucial before proceeding to the next task. A scatter plot is a visualization method that is used to identify the presence of outliers.

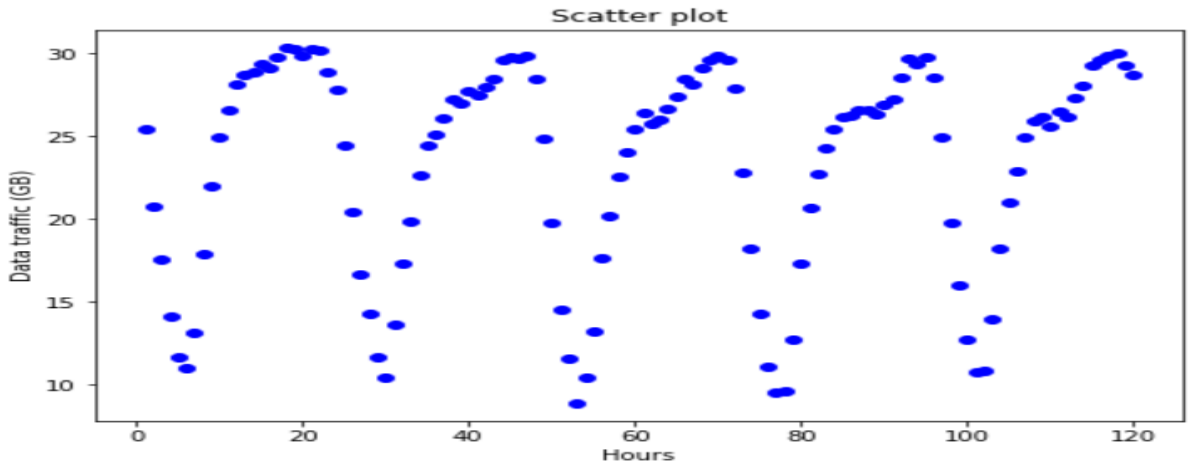


Figure 4.4 Scatter plot for downlink data traffic

Figure 4.4 above shows a sample scatter plot for aggregate data traffic of eNodeBs for a duration of five days that implicitly illustrates the daily seasonality pattern. The data points follow the seasonality of data traffic, and there is no any outliers present in the dataset.

2) Data Integration

In addition to LTE data traffic dataset, the engineering parameters dataset is collected that contain information such as the location of eNodeBs and the number of cell in each site, and those datasets are combined as a single dataset.

4.2.3 Data Transformation

Feature scaling is used to transform the value of features into some specific range. Without feature scaling, some machine learning algorithms that use distance metrics are affected by the span of the value found in the dataset. The feature values in the research dataset have different ranges, and the standard scaler method is used for scaling DL data traffic, UL data traffic, cell maximum user and RAB features.

4.3 Clustering

Clustering is the process of grouping data points depending on certain characteristics, and it is used to group base stations based on traffic load for network traffic analysis [18]. Among different clustering methods, K-Means clustering is selected for two reasons. First, for large datasets, other clustering methods such as hierarchical clustering took much processing time because it evaluates more metrics to decide the data point into one cluster. Secondly, K-Means clustering is used for clustering mobile data traffic in [20] and can provide decent output even if the mobile data traffic volume is dynamic in time.

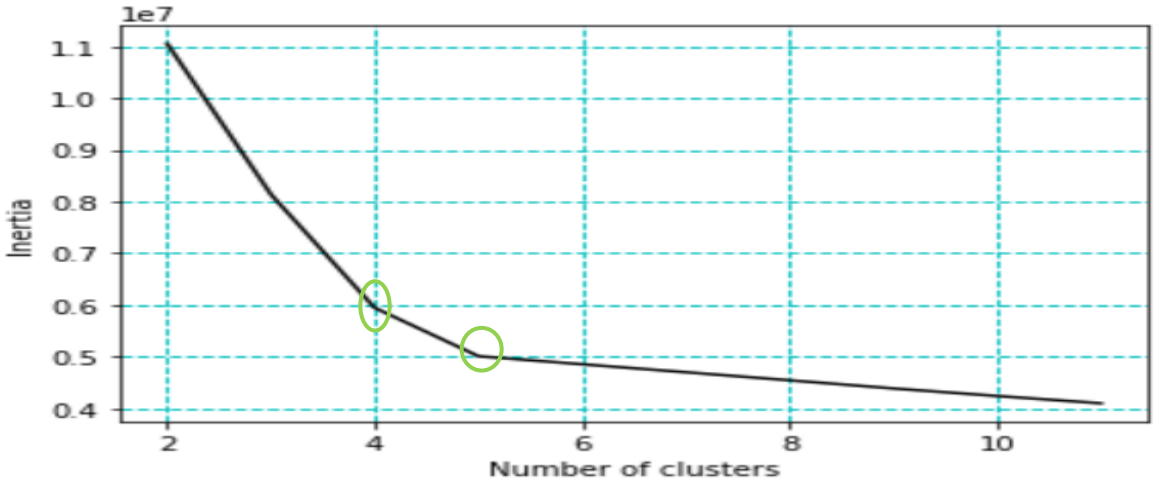


Figure 4.5 Elbow method to determine the optimal value of the number of clusters

Figure 4.5 above shows the output of the elbow method used to determine the optimal number of clusters in K-Means clustering using the sklearn library. The number of

clusters is varied from 2 to 12 while running the K-Means algorithm, and the sum of the squared distance (inertia) is evaluated. Figure 4.5 below demonstrated the output of K-Means clustering, and the elbow occurs at numbers 4 and 5 indicating the optimal cluster for the dataset is either 4 or 5.

A silhouette score is a metric used to evaluate the goodness of a clustering technique in which the optimal number of clusters have a higher score. Table 4.3 below shows the value of the silhouette score for K-Means clustering algorithm output.

Table 4.3 Silhouette score values for some number clusters

Number of clusters	**3**	**4**	**5**	**6**
Silhouette score	0.59	**0.65**	0.54	0.51

The optimal number of clusters is selected to be four because the silhouette score value is higher as compared to other clusters scores. After that, the sites are grouped into four clusters and the number of sites in each cluster is illustrated in Table 4.4 below.

Table 4.4 Number of LTE eNodeBs in each cluster

Cluster-ID	**Cluster Number**	**Number of eNodBs**
0	1	284
1	2	102
2	3	217
3	4	55

Figure 4.6 below depicts the geographical distribution of sites in each cluster, and it illustrates how sites from various geographical areas are grouped into the same cluster having similar traffic patterns. It also demonstrates that some base stations found in the same locations are grouped into different clusters, particularly sites in clusters one and

three. Furthermore, it shows that the data traffic patterns of some adjacent base stations are not the same and for such sites, time-series-based clustering is more important than geographical-based clustering for optimization purposes.

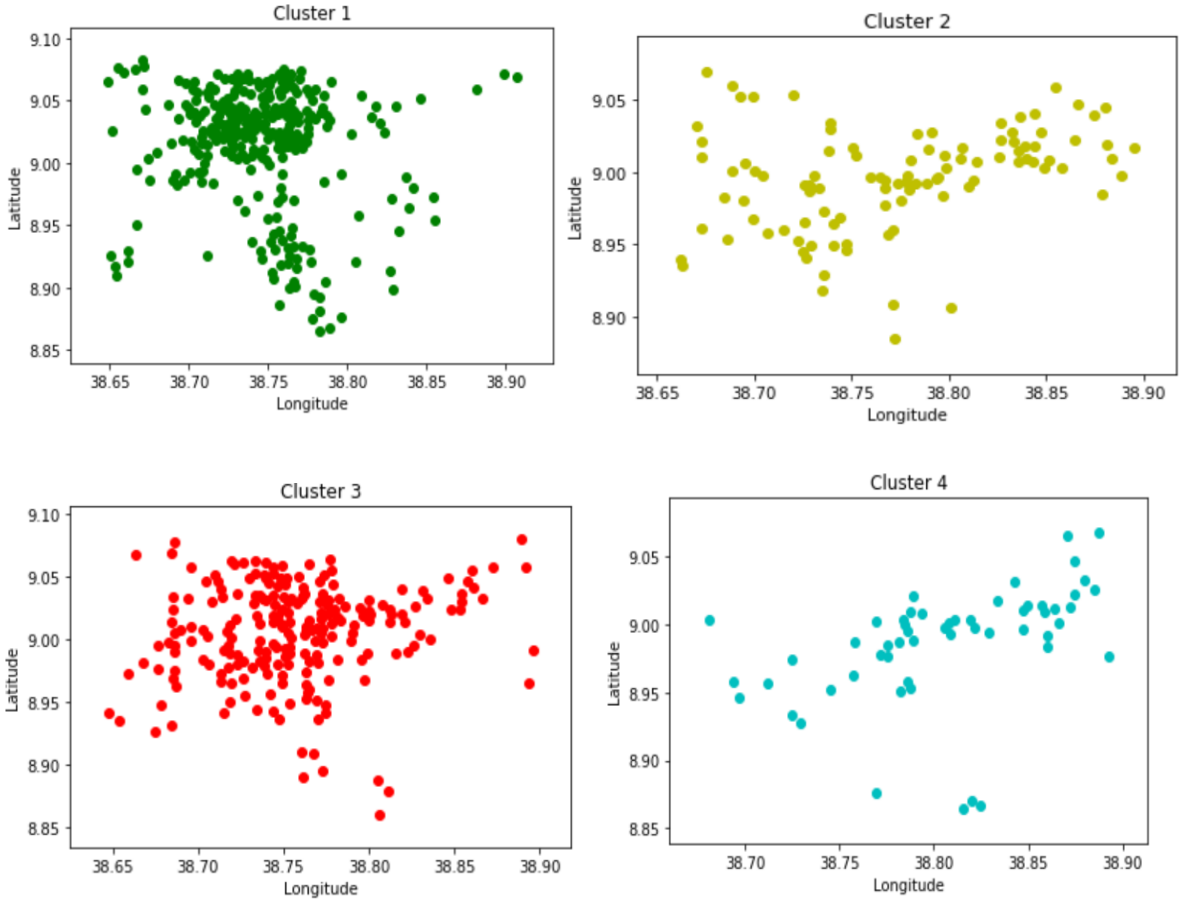


Figure 4.6 4G eNodeBs geographical distribution in each cluster

4.4 Dataset Split and Model Evaluation

The majority of real-world datasets are classified as either cross-sectional data or time-series data. Cross-sectional data are observations gathered from different individuals or groups, whereas time-series data is a collection of information gathered from a single entity at evenly spaced time intervals.

To achieve an unbiased estimate of model performance, cross-sectional data is split into train, validation, and test data using a variety of cross-validation techniques. However, in the case of time-series data, the dataset should be split while maintaining chronological order. In the case of time-series data, the walk-forward method is used to split the dataset rather than the cross-validation method, in which the validation sets are some steps forward in time from the training sets [48]. In this study, the dataset is split into train, validation, and test set with a percentage of 70%,15%, and 15%, respectively, and a walk-forward validation technique is used to evaluate the model. The optimal hyperparameters values are obtained during model training using the training data set [49].

After building the model, it has been evaluated using validation data set to check whether it is overfitting, underfitting, or a good fit. Overfitting occurs when a model performs well in the training data but not in the test data set. It is due to that the model learns every aspect and unable to generalize for the unseen dataset. But in the case of underfitting, the model will have high training error and test error because it can't capture the relationship between input and output values. The model performs well for both training and test data sets for the case of good fit.

The most common performance evaluation metrics for regression models are Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). In this work, RMSE and MAPE were used as evaluation metrics, and the formula for those metrics are:

$$RMSE = \sqrt{\frac{1}{n}} \sum_{i=1}^{n} (y_i - \hat{y}_i)2 \quad (4.1)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (4.2)$$

Where n is the number of data points

4.5 Model Configuration

In this research, the CNN-LSTM model is proposed to solve the regression problem for mobile data traffic. While developing a model, the values of hyperparameters should be initialized before training the model, and the grid search technique is applied to determine the optimal configuration of the model. A grid search method does an exhaustive search in a subset of predefined values to find the best parameters inside sample space. [50]. Some hyper parameters such as type of optimizer, loss function, and activation function are not included in the grid search to minimize the searching time because the grid search method is computational expensive even for a small number of parameters in the sample space. Those hyper parameters are selected as follows.

Loss function: The most common loss function for regression problems is MSE, which minimizes the difference between actual and predicted values, and MSE was used as the loss function in this study.

Optimizer: An optimizer updates the model in response to the loss function. Adam optimizer is used due to its effectiveness for stochastic gradient optimization because it is computationally efficient, requires small memory, and uses adaptive learning rates, which helps for fast convergent [51].

Activation function: The type of activation function is selected based on regression or classification problems and the types of deep learning networks such as RNN and CNN. ReLU activation function is recommended for convolutional neural networks, while sigmoid activation function is encouraged for LSTM networks [52]. In this research, ReLU is used for the CNN model, and sigmoid is for the LSTM model.

In addition, some parameters are intentionally chosen to capture the temporal characteristics of the data traffic, and those parameters are also not included in the grid search. Those parameters are kernel size, the value of stride, and the size of the Max pooling layer, and the values of those parameters are found in Table 4.5 below.

Table 4.5 Table CNN hyper parameters values

Hyper parameter	Value
Kernel size	4*1
Stride	1
Max pooling layer size	2*1

Table 4.6 below shows the remaining hyper parameters, grid search sample space, and optimal values from grid search output. The values for the sample space are selected based on the recommended values, for example, the number of filters in convolutional layers should be expressed as a power of two for reducing computational time in the convolutional operation.

Table 4.6 Grid search result for model hyper parameters

Parameter	Grid search sample space	Selected value
Number of filters in convolutional layer	[32,64]	64
Number of nodes for LSTM	[50,100]	50
Batch size	[2,8,16]	8
Number of epochs	[100,200,500]	500
Learning rate	[0.01,0.001,0.0001]	0.001

The model is trained with those optimal parameters using a training dataset, and the performance of the model is evaluated with a validation dataset. Parameter tuning is

performed for some hyper parameters to obtain a better result. The model was checked whether it is overfitting, under fitting, or a good fit before the final evaluation of the model with the test set.

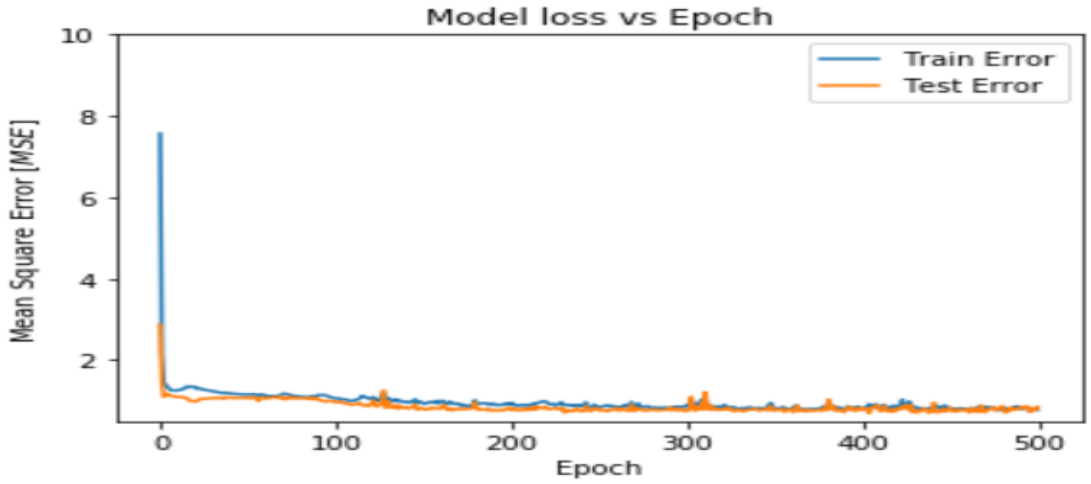


Figure 4. 7 Model evaluation output

Figure 4.7 above illustrates the comparison of training and validation error against the number of epochs, and about 300 epochs, the MSE becomes the same and constant for both training and validation datasets. It indicates that the model is well fitted and ready to evaluate with the test dataset.

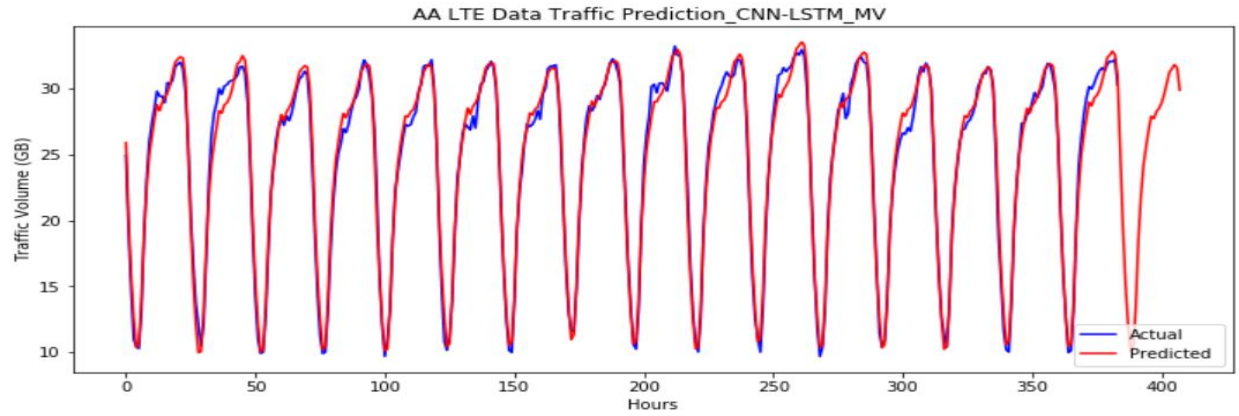
5. Result and Discussions

In this chapter, the proposed CNN-LSTM prediction model results are discussed, and the performance of the model is evaluated using the test dataset. The performance of the proposed model is compared with different models such as another deep learning model, CNN, and a baseline statistical model, SARIMA, to predict mobile data traffic per cluster level. The effect of using multivariate input features are compared with that of univariate input feature. Furthermore, the impact of imputing missing values, the effect of using different input time steps, and K Fold cross-validation for time series cellular data are investigated.

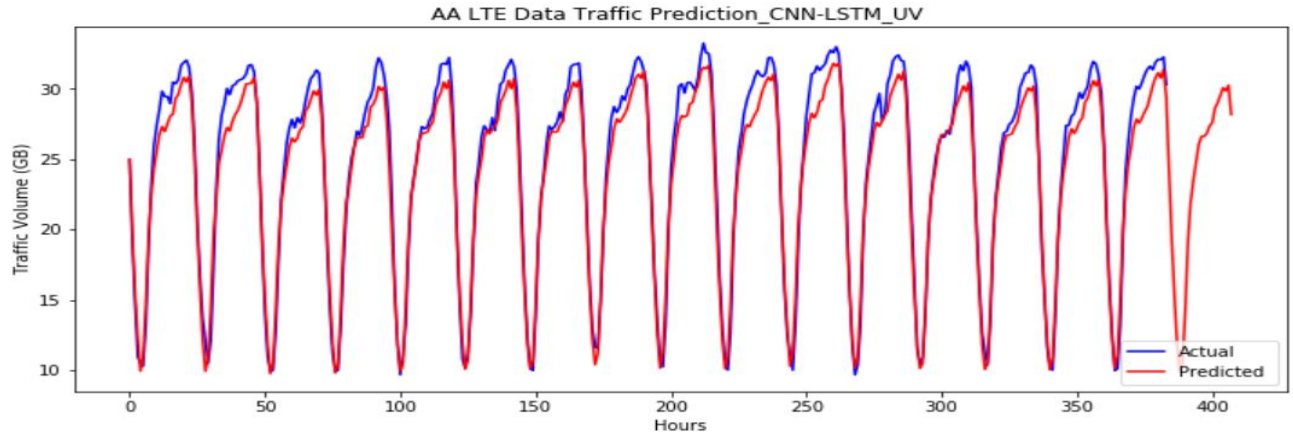
The model is developed to predict the data traffic demand for the next 24 hours in an hourly granularity providing 24 hours input data.

5.1 CNN- LSTM Model

The output of the proposed model for the mobile data traffic prediction in a typical cluster is illustrated in Figure 5.1 below. It also shows the actual data traffic pattern, which aids in comparing and visualizing the prediction result. Figure 5.1 (a), illustrates that the predicted data traffic volume has similar patterns when it is compared with the actual data traffic, and the forecasted data traffic pattern also maintains the daily seasonality. The proposed model is capable of capturing and performing well at both ends of the data traffic pattern, which has some irregularity and sharp edges for the case of multivariate features. On the contrary, the model does not succeed in capturing traffic variation for sharp edges, which has distinctly observed in figure 5.1 (b) for the univariate case. The improved result with multivariate features also demonstrates the ability deep learning model, CNN-LSTM, to extract salient information from complex data required for prediction.



(a)



(b)

Figure 5.1 Mobile data traffic prediction with CNN-LSTM model a) Multivariate features b) Univariate feature

The proposed model's performance is assessed for both univariate and multivariate situations, and the results of the evaluation metrics are provided in Table 5.1 below. The proposed CNN-LSTM model with multivariate features outperforms the univariate input feature with all specified metrics.

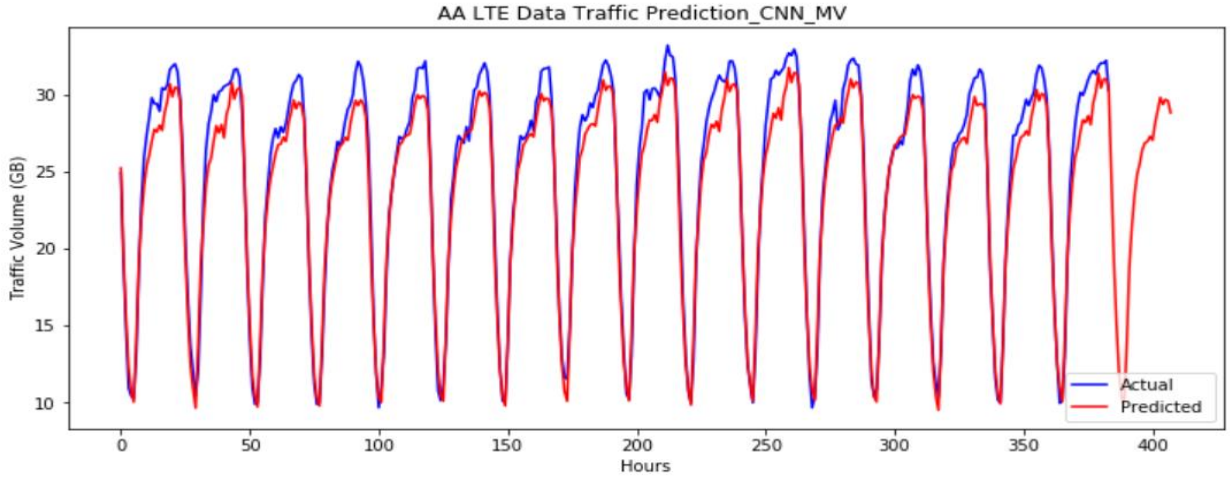
Table 5.1 Performance evaluation of CNN-LSTM model

	RMSE	MAPE	R^2
Multivariate Features	0.81	2.97	0.99
Univariate Feature	1.28	4.48	0.98

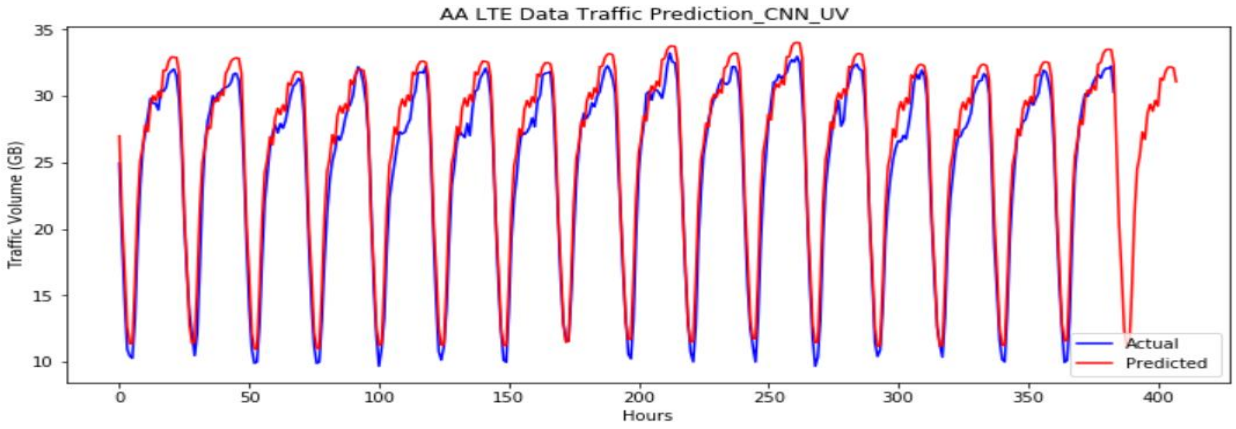
This result confirms that using multivariate features improved the performance of the proposed data traffic prediction model as compared to univariate feature.

5.2 CNN Model

In addition to the proposed approach, another deep learning based model is used to analyze and compare the performance of the proposed method. The CNN model is used to predict data traffic demand for both multivariate and univariate features similar to the proposed model.



(a)



(b)

Figure 5.2 Mobile data traffic prediction using CNN- model a) Multivariate features b) Univariate feature

Figure 5.2 above depicts the CNN model's output for mobile data traffic prediction and actual data traffic pattern. It demonstrates that the CNN model can learn the mobile data traffic pattern while keeping the daily seasonality of data traffic. However, the CNN model cannot capture high data traffic volume variations with multivariate input features due to sharp edges and irregular shapes. The CNN model also fails to learn the traffic variations for high and low data traffic during univariate feature input. Like the CNN-LSTM model, the CNN model with multivariate features has a better performance when we compared it to univariate input features. The result indicates that the 1D CNN model grasps the dynamics of cellular data traffic variation but it fails to match with actual data traffic for high traffic load.

Similarly, the CNN model performance is evaluated for both univariate and multivariate cases, and the results of the evaluation metrics have listed the Table 5.2 below.

Table 5.2 CNN Model evaluation results

	RMSE	MAPE	R^2
Multivariate Features	1.34	4.44	0.98
Univariate Feature	1.53	6.20	0.97

Table 5.2 above illustrates the superiority of multivariate features over univariate input features for the CNN model when we compare the evaluation metrics.

5.3 SARIMA Model

The effectiveness of the deep neural network for predicting mobile data traffic can be verified by comparing it with the statistical model, and the SARIMA model is used as a baseline. Figure 5.3 presents the data traffic prediction using SARIMA *(1,1,1) (1,0,0) [24]* model using univariate input feature. It tries to predict the temporal variation of the data traffic most of the time but unable to sufficiently learn the patterns at low and high traffic

loads. Because of its inability in handling nonlinear and complex data, the SARIMA model fails to capture sharp edges and irregularities.

The performance of the SARIMA model is evaluated considering the univariate input feature, and it has a value of 3.14, 11.34, and 0.94 for evaluation metrics of RMSE, MAPE, and R^2 respectively. These results are significantly lower than those obtained by the proposed CNN-LSTM deep learning model.

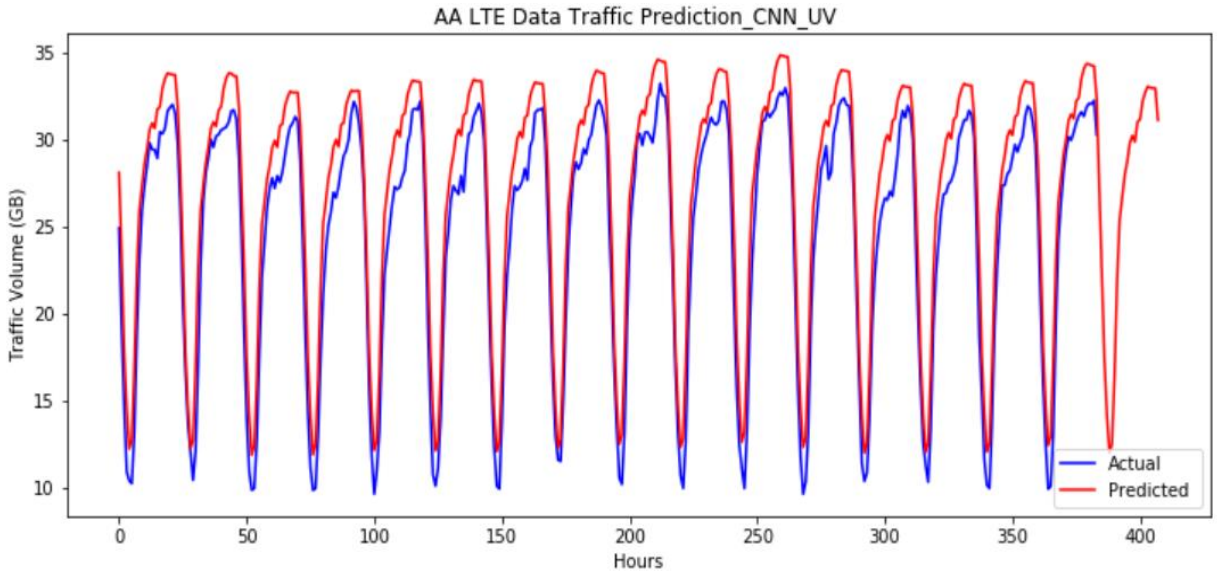


Figure 5.3 Mobile data traffic prediction using SARIMA model

5.4 Missing Values, Input time step and K Fold CV Impact

5.4.1 Impact of Missing Values

The effect of missing values is analyzed by removing the data with missing values. Figure 5.4 below demonstrates the model prediction results with and without imputing missing values from the dataset. The output shows that without filling the missing values, the prediction result is unable to match the actual data traffic pattern, whereas the model performs well in capturing traffic variation for the imputed dataset. When we evaluate

the model without filling the missing value, the model performance is not good. The model's performance becomes poor due to that the model encounters a different set of data during testing owing to missing values in the time series dataset.

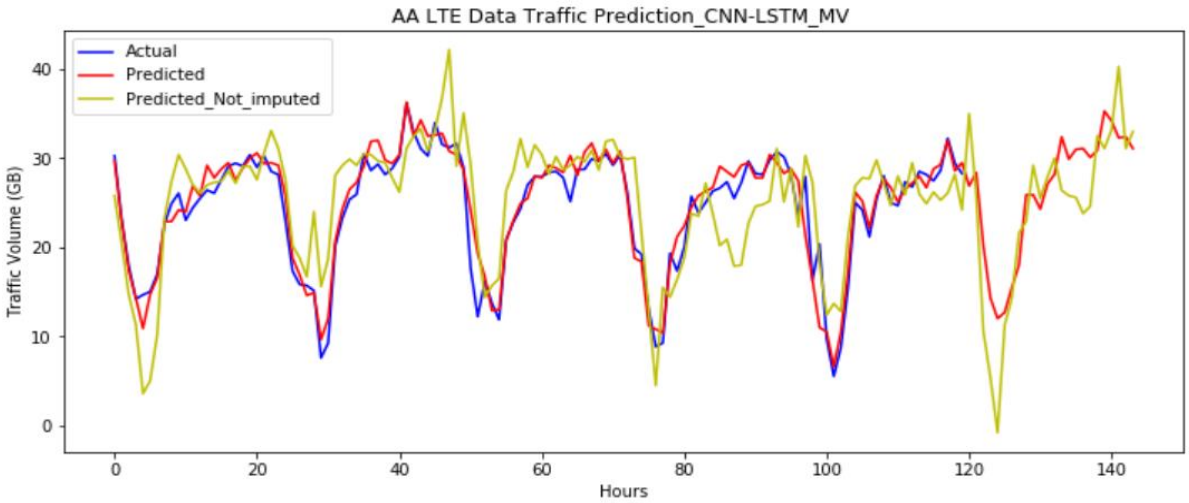


Figure 5.4 Model prediction for with and without imputation of missing value

The performance of the models is compared as illustrated in table 5.3 below, and it indicates that the performance of the model after filling missing has a better result.

Table 5.3 Model performance comparison for the effect of missing value

CNN-LSTM	RMSE	MAPE
With missing values imputation	2.01	6.88
Without missing values imputation	4.56	19.01

5.4.2 Impact of input time step length

Next, the impact of input time steps is investigated for an input time step of 168 hours in addition to 24 hours input time steps. Figure 5.5 below illustrates the data traffic prediction for the CNN-LSTM model using 168 hours input time steps compared to the actual data traffic, and it captures the data traffic variation well, including for irregular shapes and sharp edges at both ends.

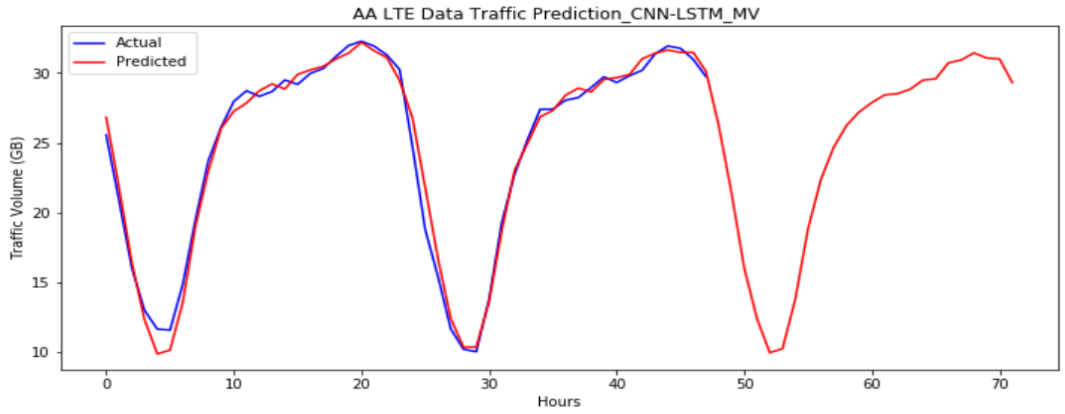


Figure 5.5 CNN-LSTM model prediction output for 168 input time steps

Table 5.4 shows model performance comparison for the two input time steps, and an input time step with 168 hours provides slight improvement.

Table 5.4 Model performance comparison for 24 and 168 input time steps

Input time step	RMSE	MAPE
24 hours	0.81	2.97
168 hours	0.78	2.69

However, this modest performance improvement comes at the expense of computational time. The elapsed time in both cases while training the model is illustrated in figure 5.6 below. The computational time for 168 hours input step is approximately three times that of the 24 hours input time step, and the computation time difference between the two input steps will be more significant as the size of the dataset becomes larger.

```
101 print(time.ctime())

Fri Jun 18 10:25:39 2021
Fri Jun 18 10:26:49 2021
```

(a)

```
101 print(time.ctime())

Fri Jun 18 09:52:43 2021
Fri Jun 18 09:55:59 2021
```

(b)

Figure 5.6 Model computational time a) 24 hours b) 168 hours input time step

5.4.3 Applying K Fold Cross Validation

To investigate the effect of walk-forward validation that preserves the temporal ordering of the time series data, the K Fold cross-validation is applied to compare model performances. The dataset is split using K fold cross-validation, choosing the value of K=5 depending on the size of the dataset. Figure 5.7 shows the comparison of actual data traffic and predicted data traffic for the CNN-LSTM model using 5 Fold cross-validation. The result shows that it is unable to learn the traffic variation for irregular shapes and a sharp edge that occur during high and low traffic loads respectively.

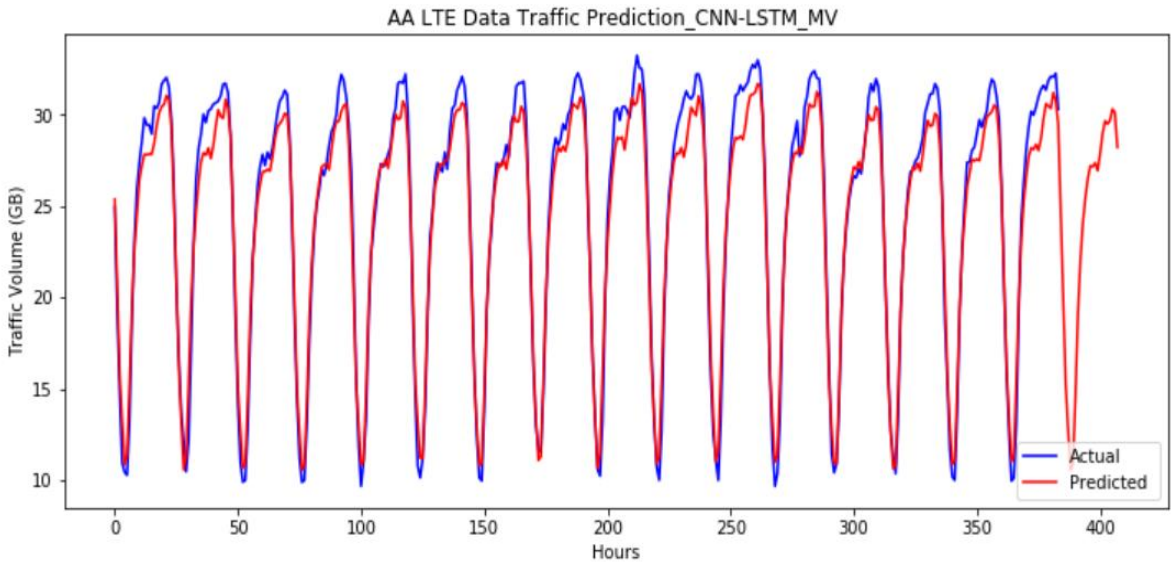


Figure 5.7 Model prediction output 5 Fold Cross validation

The CNN-LSTM model performance with 5 Fold cross-validation techniques is evaluated, and it has a value of 1.20 and 4.42 for evaluation metrics of RMSE and MAPE, respectively. Those results are outperformed by the CNN-LSTM model using walk-forward validation techniques but using the K Fold cross-validation method is simple due to the availability of the algorithm in Sklearn library.

6. Conclusion and Recommendation

6.1 Conclusion

The cellular network capacity constantly changing due to increasing mobile data demand with different user behaviors. Predictive methods become crucial to capture the dynamics of mobile data traffic demand as many real-world forecasting problems. In this work, mobile data traffic prediction model is proposed based on a deep learning model using multivariate input features that help in improving the performance of the model. The prediction model is proposed per cluster level in which eNodeBs with similar data traffic patterns are grouped into the same cluster. Studying the data traffic characteristics per cluster level helps significantly for optimization by using the resource pooling approach. K-Means clustering is used for clustering after arranging the dataset to be appropriate for time series data clustering. The proposed hybrid CNN-LSTM networks exploit the power of the CNN model to extract salient features in the complex and non-linear dataset as well as an LSTM to capture long-short dependency for time series data. The model predicts the next 24 hours of data traffic by observing the previous day's data traffic and has RMSE of 0.81 and 1.27 for multivariate and univariate features respectively. The result shows that using multivariate features improves the model performance by 58% compared to the univariate input feature, and it also indicates that a hybrid 1D CNN and LSTM model can be promising tools for analyzing mobile data traffic data.

In addition, the effect of missing values is considered by eliminating the missing values from the dataset. As demonstrated by the results, imputing the missing values improve model accuracy, whereas removing the missing values from the dataset affects the model's performance. The impact of input time steps is also investigated, and the model is tested for input time steps of 24 hours and 168 hours. The results demonstrate that using a 168-hour input time step increases model performance slightly over a 24-hour

input time step. However, compared to the 24-hour input steps, it took a significant amount of processing time.

Furthermore, the impact of using K-fold cross-validation is examined for the proposed CNN-LSTM model while splitting the dataset into train and test. The model performance is assessed for 5-Fold cross-validation technique compared to walk-forward validation. The result shows that the walk-forward validation technique outperforms the K-fold cross-validation approach.

Finally, the study findings show the prediction capability of the CNN-LSTM model for mobile data traffic demand along with multivariate input features, which significantly improves the model's performance. The research output can be used for:

- Short-term optimization: Cluster-based data traffic prediction considers all the eNodeBs, and having insight about all base stations gives the chance to implement a resource pool approach.
- Long-term capacity enhancement: The model can be used for long-term capacity planning only by changing the input time step granularity.
- The model can be used for the optimization of individual cells, sites, or aggregated sites.

6.2 Recommendation

Investigating the impact of other variates of clustering methods on model performance can be considered as future works. In addition, considering more specific multivariate features such as the amount of spectrum utilized and RAB attributes like maximum source data, traffic type, and maximum bit rate can improve the model performance.

References

[1] W. Shi, "Almost one zettabyte of mobile data traffic in 2022 – Cisco," *Telecoms.com*, Feb. 20, 2019. https://telecoms.com/495666/almost-one-zettabyte-of-mobile-data-traffic-in-2022-cisco/ (accessed Oct. 03, 2020).

[2] S. E. Elayoubi *et al.*, "5G service requirements and operational use cases: Analysis and METIS II vision," in *2016 European Conference on Networks and Communications (EuCNC)*, Jun. 2016, pp. 158–162. doi: 10.1109/EuCNC.2016.7561024.

[3] "GSMA-Data-Demand-Explained-June-2015.pdf." Accessed: Aug. 06, 2021. [Online]. Available: https://www.gsma.com/spectrum/wp-content/uploads/2015/06/GSMA-Data-Demand-Explained-June-2015.pdf

[4] "Introduction to Time Series Analysis: Time-Series Forecasting Machine learning Methods & Models | by Neelam Tyagi | Analytics Steps | Medium." https://medium.com/analytics-steps/introduction-to-time-series-analysis-time-series-forecasting-machine-learning-methods-models-ecaa76a7b0e3 (accessed Nov. 17, 2020).

[5] Q. H. Do, T. T. H. Doan, T. V. A. Nguyen, N. T. Duong, and V. V. Linh, "Prediction of Data Traffic in Telecom Networks based on Deep Neural Networks," *J. Comput. Sci.*, vol. 16, no. 9, pp. 1268–1277, Sep. 2020, doi: 10.3844/jcssp.2020.1268.1277.

[6] Ethio telecom, "Mobile network traffic and KPI analysis," *Addis Ababa*, May 2020.

[7] Ethio telecom, "ethio telecom OSS data." Addis Ababa, Ethiopia, Aug. 20, 2020.

[8] Ethio telecom, "Ethio telecom marketing section." Addis Ababa, Ethiopia, Sep. 2020.

[9] D. Zhang, L. Liu, C. Xie, B. Yang, and Q. Liu, "Citywide Cellular Traffic Prediction Based on a Hybrid Spatiotemporal Network," *Algorithms*, vol. 13, no. 1, Art. no. 1, Jan. 2020, doi: 10.3390/a13010020.

[10] N. I. Sapankevych and R. Sankar, "Time Series Prediction Using Support Vector Machines: A Survey," *IEEE Comput. Intell. Mag.*, vol. 4, no. 2, pp. 24–38, May 2009, doi: 10.1109/MCI.2009.932254.

[11] "Use of Local Linear Regression Model for Short-Term Traffic Forecasting - Hongyu Sun, Henry X. Liu, Heng Xiao, Rachel R. He, Bin Ran, 2003." https://journals.sagepub.com/doi/abs/10.3141/1836-18 (accessed Aug. 10, 2021).

[12] B. Mohammed, N. Krishnaswamy, and M. Kiran, "Multivariate Time-Series Prediction for Traffic in Large WAN Topology," in *2019 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, Sep. 2019, pp. 1–4. doi: 10.1109/ANCS.2019.8901870.

[13] "1D convolutional neural networks and applications: A survey," *Mech. Syst. Signal Process.*, vol. 151, p. 107398, Apr. 2021, doi: 10.1016/j.ymssp.2020.107398.

[14] T.-Y. Kim and S.-B. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks," *Energy*, vol. 182, pp. 72–81, Sep. 2019, doi: 10.1016/j.energy.2019.05.230.

[15] B. Mohammed, N. Krishnaswamy, and M. Kiran, "Multivariate Time-Series Prediction for Traffic in Large WAN Topology," in *2019 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, Sep. 2019, pp. 1–4. doi: 10.1109/ANCS.2019.8901870.

[16] T. Li, M. Hua, and X. Wu, "A Hybrid CNN-LSTM Model for Forecasting Particulate Matter (PM2.5)," *IEEE Access*, vol. 8, pp. 26933–26940, 2020, doi: 10.1109/ACCESS.2020.2971348.

[17] T. Getinet, "Hybrid SARIMA-ELM-based Data Traffic Forecasting: The Case of UMTS Network in Addis Ababa, Ethiopia," Thesis, AAU, 2018. Accessed: Oct. 04, 2020. [Online]. Available: http://etd.aau.edu.et/handle/123456789/15234

[18] A. Dalgkitsis, M. Louta, and G. T. Karetsos, "Traffic forecasting in cellular networks using the LSTM RNN," in *Proceedings of the 22nd Pan-Hellenic Conference on Informatics*, New York, NY, USA, Nov. 2018, pp. 28–33. doi: 10.1145/3291533.3291540.

[19] 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)*, Sep. 2018, pp. 1827–1832. doi: 10.1109/PIMRC.2018.8581000.

[20] B. S. Shawel, T. T. Debella, G. Tesfaye, Y. Y. Tefera, and D. H. Woldegebreal, "Hybrid Prediction Model for Mobile Data Traffic: A Cluster-level Approach," in *2020 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2020, pp. 1–8. doi: 10.1109/IJCNN48605.2020.9207655.

[21] B. Mahdy, H. Abbas, H. Hassanein, A. Noureldin, and H. Abou-zeid, "A Clustering-Driven Approach to Predict the Traffic Load of Mobile Networks for the Analysis of Base Stations Deployment," *J. Sens. Actuator Netw.*, vol. 9, p. 53, Nov. 2020, doi: 10.3390/jsan9040053.

[22] N.Bhandari, S.Devra and K.Singh, "Evolution of Cellular Network: From 1G to 5G," *Int. J. Eng. Tech.*, vol. 3, no. 5, Oct. 2017.

[23] T.-T. Tran, Y. Shin, and O.-S. Shin, "Overview of enabling technologies for 3GPP LTE-advanced," *EURASIP J. Wirel. Commun. Netw.*, vol. 2012, no. 1, p. 54, Feb. 2012, doi: 10.1186/1687-1499-2012-54.

[24] "LTE Definited through LTE Network Diagrams," *RCR Wireless News*, May 09, 2014. https://www.rcrwireless.com/20140509/evolved-packet-core-epc/lte-network-diagram (accessed May 17, 2021).

[25] A.Elnashar and M.A. El-saidny, "Design, Deployment and Performance of 4G-LTE Networks: A Practical Approach," *John Wiley Sons*, 2014.

[26] "Evolved Packet Core - an overview | ScienceDirect Topics." https://www.sciencedirect.com/topics/computer-science/evolved-packet-core (accessed May 17, 2021).
[27] Ghassan A. Abed,Mahamod Ismail,Kasmiran Jumari, "The Evolution to 4G Cellular Systems: Architecture and Key Features of LTEAdvanced Networks," *Int. J. Comput. Netw. Wirel. Commun. IJCNWC*, vol. Vol. 2, Jan. 2012.
[28] Harri Holma and Antti Toskala, "LTE for UMTS Evolution to LTE-Advanced," Second Edition., John Wiley and Sons, 2011, pp. 26–30.
[29] 3GPP TS 23.101 version 8.0.0 Release 8, "Universal Mobile Telecommunications System (UMTS) and LTE General UMTS Architecture," *Tech. Specif.*.
[30] 3GPP A Global Iniviative, "LTE Resource Guide," [Online]. Available: http://www.cs.columbia.edu/~hgs//teaching/ais/hw/anritsu.pdf
[31] Telesystem Innovations, "LTE in a Nutshell: The Physical Layer," WHITE PAPER, 2010.
[32] Xi Li, "Radio Access Network Dimensioning for 3G UMTS," Ph.D. dissertation., Hunan,China, 2009.
[33] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "DNN-based prediction model for spatio-temporal data," Oct. 2016, pp. 1–4. doi: 10.1145/2996913.2997016.
[34] X. Wang, Z. Zhou, Z. Yang, Y. Liu, and C. Peng, "Spatio-temporal analysis and prediction of cellular traffic in metropolis," in *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, Oct. 2017, pp. 1–10. doi: 10.1109/ICNP.2017.8117559.
[35] "Difference Between Deep Learning and Machine Learning Vs AI." https://www.guru99.com/machine-learning-vs-deep-learning.html (accessed May 15, 2021).
[36] S. Arora, "Supervised vs Unsupervised vs Reinforcement," *AITUDE*, Jan. 29, 2020. https://www.aitude.com/supervised-vs-unsupervised-vs-reinforcement/ (accessed Jun. 04, 2021).
[37] K. Shiruru, "An Introduction to Artificial Neural Network," *Int. J. Adv. Res. Innov. Ideas Educ.*, vol. 1, pp. 27–30, Sep. 2016.
[38] R. Rajagukguk, R. A. Ardiansyah Ramadhan, and H.-J. Lee, "A Review on Deep Learning Models for Forecasting Time Series Data of Solar Irradiance and Photovoltaic Power," Dec. 2020. doi: 10.3390/en13246623.
[39] A. Cecaj, M. Lippi, M. Mamei, and F. Zambonelli, "Comparing Deep Learning and Statistical Methods in Forecasting Crowd Distribution from Aggregated Mobile Phone Data," *Appl. Sci.*, vol. Volume 10, Sep. 2020, doi: 10.3390/app10186580.
[40] Y. LeCun, Y.Bengio and G.Hinton, "Deep learning," *Nature*, vol. 52, no. 7553, pp. 436–444, May 2015.

[41] "Convolutional Neural Network (CNN) for Time Series Classification." https://www.macnica.co.jp/business/ai_iot/columns/135112/ (accessed May 31, 2021).

[42] Gaowei Xu, Tianhe Ren, Yu Chen and Wenliang Che, "A One-Dimensional CNN-LSTM Model for Epileptic Seizure Recognition Using EEG Signal Analysis," vol. 14, Dec. 2020.

[43] J. Brownlee, "Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python," Machine Learning Mastery, 2018.

[44] "Data preparation: Selection, Preprocessing, and Transformation Literature: Literature: I.H. Witten and E. Frank, Data Mining, chapter 2 and chapter ppt download." https://slideplayer.com/slide/7751000/ (accessed May 22, 2021).

[45] M. Joshi and T. H. Hadi, "A Review of Network Traffic Analysis and Prediction Techniques," *ArXiv150705722 Cs*, Jul. 2015.

[46] I. Pratama, A. E. Permanasari, I. Ardiyanto, and R. Indrayani, "A review of missing values handling methods on time-series data," in *2016 International Conference on Information Technology Systems and Innovation (ICITSI)*, Oct. 2016, pp. 1–6. doi: 10.1109/ICITSI.2016.7858189.

[47] M.N. Noor , A.S. Yahaya , N.A. Ramli, and A.M. Mustafa, "Filling missing data using interpolation methods: study on the effect of fitting distribution," *Trans Tech*, vol. Vols 594-595, 2014, doi: 10.4028/www.scientific.net/KEM.594-595.889.

[48] Allison Koenecke, "Applying Deep Neural Networks to Financial Time Series Forecasting." Institute for Computational & Mathematical Engineering.

[49] T. Shah, "About Train, Validation and Test Sets in Machine Learning," *Medium*, Jul. 10, 2020. https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7 (accessed Jun. 03, 2021).

[50] I. Syarif, A. Prügel-Bennett, and G. Wills, "SVM Parameter Optimization using Grid Search and Genetic Algorithm to Improve Classification Performance," 2016, doi: 10.12928/TELKOMNIKA.V14I4.3956.

[51] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *ArXiv14126980 Cs*, Jan. 2017, Accessed: Jun. 05, 2021. [Online]. Available: http://arxiv.org/abs/1412.6980

[52] J. Brownlee, "How to Choose an Activation Function for Deep Learning," *Machine Learning Mastery*, Jan. 17, 2021. https://machinelearningmastery.com/choose-an-activation-function-for-deep-learning/ (accessed Jun. 05, 2021).