JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

Cellular Network Traffic Forecast Using RNN LSTM for Proactive Quality of Service Management: The Case of Jimma

By

Aklilu Alemayehu

This thesis is submitted to School of Graduate Studies of Jimma University in partial fulfilment of the requirements for the degree of

Master of Science

in

Communication Engineering

April 2022

Jimma, Ethiopia

JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

FACULTY OF ELECTRICAL AND COMPUTER ENGINEERING

Cellular Network Traffic Forecast Using RNN LSTM for Proactive Quality of Service Management: The Case of Jimma

By

Aklilu Alemayehu

Advisor: Dr. Kinde Anlay

Co-Advisor: Eng. Sherwin N. Catolos

Submission Date: March, 2022

Declaration

I declare this thesis with the title of "Cellular Network Traffic Forecast Using RNN LSTM for Proactive Quality of Service Management: The Case of Jimma" as my own work except where explicitly stated otherwise in the text and I assure it with my signature.

RESEARCH THESIS SUBMITTED BY

Aklilu Alemayehu ______________ 29/03/22

Signature Date

Approved by Advisors:

Advisor: Dr.Kinde Anlay ______________ 29/03/2022

Signature Date

Co-advisor: Eng. Sherwin N. Catolos ______________ 26/03/2022

Signature Date

Approved by faculty of Electrical and computer engineering research

Examination members:

1.Muluneh Mekonnen (PhD) ______________ ______________

Signature Date

2. Getachew Alemu (M.Sc) ______________ 29/03/2022

Signature Date

3. Sofiya Ali ______________ 29/03/2022

Signature Date

Acknowledgements

First and foremost, praise be to the Almighty God for his countless blessings. Then I would like to thank my Advisor Dr. Kinde Anlay and my Co-Advisor Eng. Sherwin N. Catolos for their invaluable support throughout the thesis preparation.I am also grateful to Mr. Getachew Alemu, the former Communication Chair and the incumbent Mrs. Sofia Ali for all the role they played during my study and thesis preparation.

Abstract

Mobile Network Operators(MNOs) are expected to wisely and proactively manage the required QoS of the ever increasing cellular network traffic for the diverse services that are expected to be provided over their networks. New radio resource dimensioning, continuous expansion and optimization, and making business strategic decision requires knowledge of the amount and mix of traffic going to be carried over the mobile network. In Ethiopia, Ethio Telecom have been the only integrated telecom service provider aspiring to meet the tremendously growing cellular traffic demand in the country. The activities performed by the company to meet the demands of its customers is being challenged by unexpected explosive growth in traffic demand and evolution in user behavior. In a multi-operator market the company have to stay alert and monitor the service demand in advance in order to stay competitive both in terms of meeting its customers' Quality of Service(QoS) requirement and the return on investment (ROI).

This thesis proposed a cellular network forecasting model using state of the art deep learning technique, LSTM-RNN to predict the future peak hour traffic volume and mix that will be carried over the mobile network. This is achieved by training and validating the model using both CS (Circuit Switched) and PS (Packet Switched) peak hour traffic data collected from Ethio Telecom mobile network. The proposed traffic forecasting model can be used as a tool to predict future demand which will be an input for cellular radio network design, optimization and to make high level strategic decision on service and technology evolution.

The model is developed to perform prediction for voice, downlink and Uplink data traffic. Model performance is evaluated using 10% of the dataset which is a prediction about three months' time ahead and the mean absolute percentage error (MAPE) is found out to be 0.07%, 0.03% and 0.04% for voice, downlink and uplink data traffic respectively.

Keywords: ***deep learning, cellular traffic forecasting, RNN-LSTM, proactive QoS management***

Contents

List of Figures

List of Tables

Abbreviations

2G	**S**econd **G**eneration
3G	**T**hird **G**eneration
3GPP	**T**hird **G**eneration **P**artnership **P**roject
5G	**F**ifth **G**eneration
ANN	**A**rtificial **N**eural **N**etwork
ARIMA	**A**uto **R**egressive **I**ntegrated **M**oving **A**verage
BRNN	**B**idirectional **R**ecurrent **N**eural **N**etwork
CN	**C**ore **N**etwork
CNN	**C**onvolutional **N**eural **N**etwork
CS	**C**ircuit **S**witched
EDGE	**E**nhanced **D**ata rates for **GSM** **E**volution
ELM	**E**xtreme **L**earning **M**achine
FFNN	**F**eed **F**orward **N**eural **N**etwork
GB	**G**iga **B**yte
GPRS	**G**eneral **P**acket **R**adio **S**ervice
GRNN	**G**eneral **R**egression **N**eural **N**etwork
GRU	**G**ated **R**ecerrent **U**nit
GSM	**G**lobal **S**ystem for **M**oblie communication
HSPA	**H**igh **S**peed **P**acket **A**ccess
HUP	**H**ardware **U**tilization **p**ackage
IP	**I**nternet **P**rotocol
KPI	**K**ey **P**erformance **I**ndicator
LSTM	**L**ong **S**hort-**T**erm **M**emory
LTE	**L**ong **T**erm **E**volution

LTE-A	**LTE A**dvanced
MAE	**M**ean **A**bsolute **E**rror
MAPE	**M**ean **A**bsolute **p**ercentage **E**rror
MIMO	**M**ultiple **I**nput **M**ultiple **O**utput
MLP	**M**ulti **L**ayer **P**ercepron
MNO	**M**obile **N**etwork **O**perator
MSE	**M**ean **S**quared **E**rror
MT	**M**obile **T**erminal
NAR	**N**onlinear **A**uto **R**egressive
NNAR	**N**eural **N**etwork **A**uto **R**egressive
PS	**P**acket **S**witched
QoS	**Q**uality of **S**ervice
RAB	**R**adio **A**ccess **B**earer
RAT	**R**adio **A**ccess **T**echnology
RB	**R**adio **B**earer
RMSE	**R**oot **M**ean **S**quared **E**rror
RNN	**R**ecurrent **N**eural **N**etwork
SARIMA	**S**easonal **A**uto **R**egressive **M**oving **A**verage
SGD	**S**tochastic **G**radient **D**escent
SVM	**S**upport **V**ector **M**achine
TC	**T**raffic **C**lass
TCN	**T**emporal **C**onvolutional **N**etwork
TE	**T**erminal **E**quipment
UMTS	**U**niversal **M**obile **T**erresterial**N**etwork
UTRAN	**U**niversal **T**ersterrial **R**adio **A**ccess **N**etwork
WCDMA	**W**ideband **C**ode **D**ivision **M**ultiple **A**ccess

Chapter 1

Introduction

1.1 Background

The growing demand to access services of the cellular network in day to day activities have resulted in a tremendous traffic growth globally. This growth is intensified especially after the emergence of digital cellular systems and user equipment which are capable of delivering PS(Packet Switched) service in addition to the existing CS(Circuit Switched) service. According to the forecast done by several organizations like Cisco, the UMTS Forum and Ericsson, this trend will continue in the foreseeable future. According to Ericsson, the global total mobile data traffic was estimated to reach around 51Exabyte(EB) per month by the end of 2020 and is projected to grow by a factor of around 4.5 to reach 226EB per month in 2026 [1]. The main drivers behind the traffic growth are: video usage, service proliferation, evolution in usage, application uptake, Machine-to-Machine(M2M), enhanced screen resolution/content offering, cloud computing, the shifting demography: increased urbanization trend, population growth in developing countries, the continued growth of audio-visual media streaming and others [2].

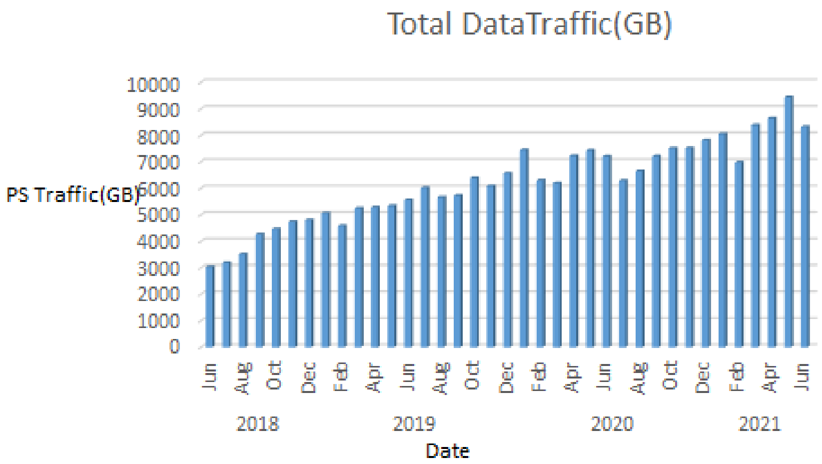

Figure 1.1: Jimma Town monthly Data Traffic Trend

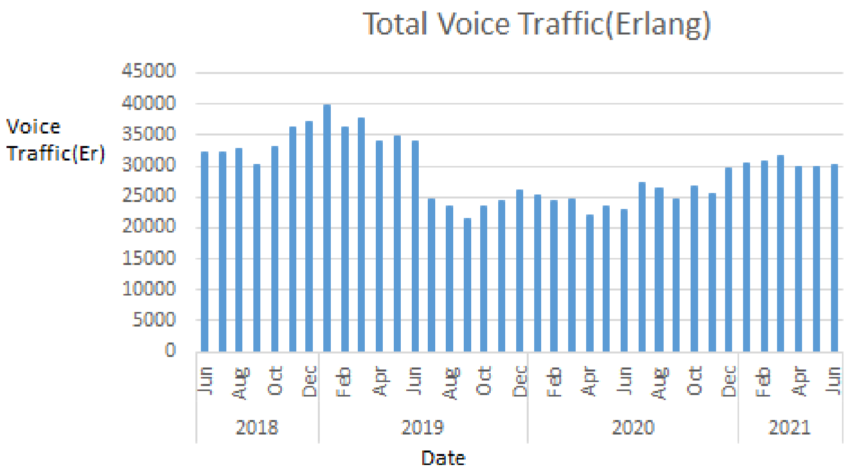

Figure 1.2: Jimma Town monthly Voice Traffic Trend

The explosive growth of traffic demand has left MNOs and equipment manufactures to deal with the challenges and opportunities of delivering the required QoS for the ever growing cellular traffic demand. QoS is the minimum quality of a requested service as perceived by the user of the service [3]. The operator can proactively manage the underlying QoS during the radio network dimensioning, operation and optimization process. Operators are sensitive to revenue, as a result, they expect good return on infrastructure invested which will be one of the factors for under dimensioning. Under-dimensioning will impact the QoS which will be the cause for customer dissatisfaction, churn rate and consequently loss of revenue. On the other hand, over dimensioning will result in excess infrastructure and energy cost which will negatively impact the return on investment.

For MNOs to strategically manage the QoS of their cellular network proactively and invest wisely on network equipment, it is crucial to have knowledge of the maximum forecasted traffic at a given point of time in advance. The maximum traffic is measured at busy/peak hour period. As a result, telecom operators have to use a proper traffic forecasting model to strategically perform: the planning and design of new cellular network, to consider advanced technology like LTE-A and 5G, the shutting down of legacy technology like 2G and the expansion of existing infrastructure. During design an operator can deploy an extra resource to be consumed by the growing traffic demand or have an easily scalable scenario of 'upgrade as demand' to meet the expected QoS.

As the number of users increase and the move from feature phone to the deployed network will be more congested degrading the expected network QoS unless the proactive activities used to handle the rapidly growing network traffic are performed properly. In Jimma town cellular network as well there is an increasing traffic demand. In addition to managing the existing network QoS, using a traffic forecasting model for companies like Ethio Telecom will also help to think about the newly emerging high data rate technologies like LTE-A and 5G that can easily meet the QoS, and validate the shutdown of legacy technology:2G.

Ethio Telecom primarily focuses on population coverage over geographic coverage which aims at demand-based service provisioning. Traffic analysis and forecasting is at the center of activities to be conducted to achieve the planned strategic road-map[4]. Figure 1.3 shows the three years foretasted subscriber growth on each available RATs the company deployed.

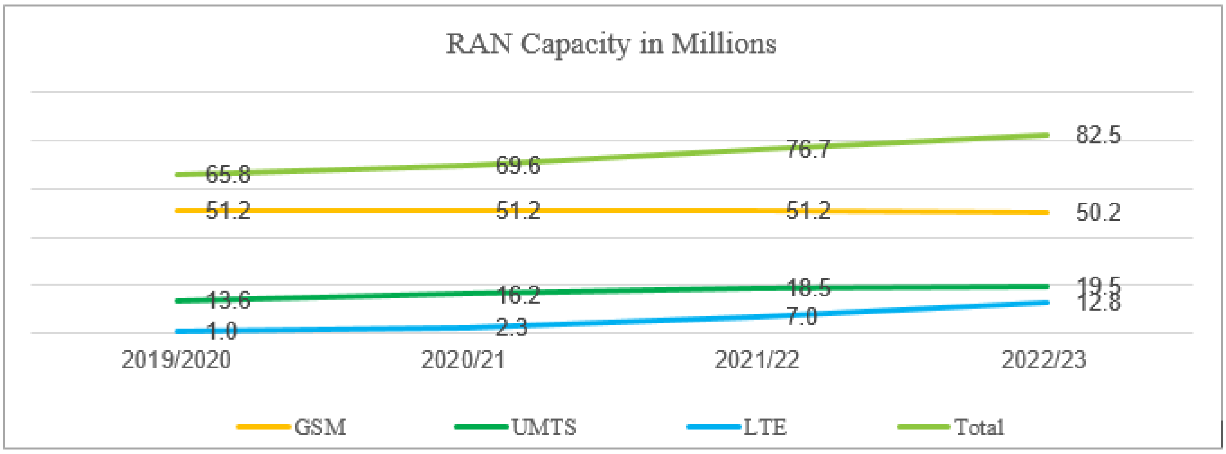

FIGURE 1.3: Ethio Telecom target subscription capacity per technology for years 2020-2023 [4]

1.2 Statement of the Problem

Cellular network traffic forecasting is used by MNOs to understand the future traffic demand in terms of volume and mix. The forecasted traffic is applied to make important decisions during radio network planning, optimization and to business plan preparation. In the current situation where the cellular network traffic is rapidly growing, knowledge of the forecasted traffic is being used to have a cellular network that can cater the requested service and satisfy the customer.

Like the global trend, In Ethiopia as well, Ethio Telecom's annual performance reports show that every year there is a significant growth in traffic demand. A very high radio resource utilization and the associated congestion is also reported which is a cause for QoS degradation[4]. Congestion occurs when the network is not capable of carrying the requested service and this happens when there is unexpected amount of traffic demand[3]. As a result, it is crucial to have a traffic forecasting mechanism that is capable of capturing the important characteristics of the cellular network traffic from historical data and accurately predict the future trend.

1.3 Objectives

1.3.1 General Objective

The general objective of this thesis is to develop a cellular network forecasting model using RNN LSTM for proactive QoS management using a dataset from Ethio Telecom Jimma town live network.

1.3.2 Specific Objectives

The specific objectives of the research are:

- Review related works on cellular network traffic forecasting and RNN LSTM based time series forecasting.
- Collect, analyze and pre-process network level busy/peak hour CS, PS uplink and PS downlink traffic data from Jimma Town UMTS network.
- Fit the selected deep learning model(LSTM)on the training dataset.
- Make a multistep prediction on the test dataset.
- For each traffic category evaluate the model prediction performance using Mean Absolute Percentage Error(MAPE) and the Root Mean Squared Error(RMSE).
- Select the best fit model per each traffic category based on performance evaluation metrics.

1.4 Methodology

In this thesis an RNN-LSTM technique is proposed to predict the CS and PS network traffic. First the total busy hour traffic is collected from a live SWR

Jimma UMTS network. Since Ethio Telecom currently uses both CS and PS radio resources to provide voice and data services, each traffic categories are independently considered for prediction. Next the dataset is pre-processed.

Data collection and Pre-processing

The dataset used for this thesis which is a daily busy/peak hour voice and data traffic is collected from Ethio Telecom network using SAP performance monitoring and data analytics tool. The dataset for both data and voice traffic collection is done for a duration of June 1, 2018 to June 30, 2021. In pre-processing, we performed data cleaning, normalization, splitting in to training and test set used to train and test the deep learning model.

Model Training and Performance Evaluation

The model is trained on a training and validation set, and tested on a test with a performance evaluation metrics until the best possible prediction accuracy is achieved. Model hyperparameters are continuously tuned during the training process to minimize the prediction error.

Tools, Metrics and Algorithms

The LSTM deep learning forecasting technique is used to develop the model. By using MAPE and RMSE we evaluated each model with a given model hyperparameter combination to select the one with the least prediction error. For dataset pre-processing and model simulation, we used python Keras, TensorFlow, R statistical software are used. For analysis, SAP, IBM SPSS and Microsoft Excel are used. Figure 1.4 demonstrates summary of the activities performed in the methodology

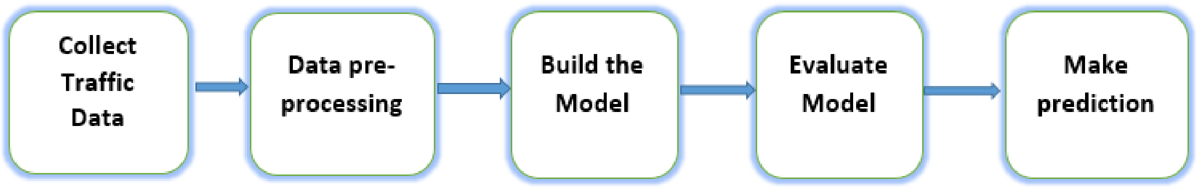

FIGURE 1.4: Methodology flow

1.5 Scope and Limitation of the Research

1.5.1 Scope

Without loss of generality, the purpose of the research is to develop a cellular network traffic forecasting model by training an RNN LSTM architecture using a traffic data collected from Ethio Telecom Jimma town live cellular network.

1.5.2 Limitations

- The model did not consider the backhaul and backbone network related issues.
- The dataset used in the model training and testing is from Ethio Telecom's South West Region Jimma UMTS network.

1.6 Significance of the Thesis

The study of cellular traffic forecasting in order to develop an efficient and accurate model to predict both long and short term future demand is still among one of the attractive research areas. This work uniquely approaches the subject using the case of Ethio Telecom commercially deployed mobile network for purposes of long term resource allocation, dimensioning and optimization in order to proactively manage the expected QoS by using the RNN-LSTM technique for time series forecasting in order to predict future traffic demand per service/resource type of the town. The expected significance is that knowledge of the forecasted traffic demand per resource type will help MNOs radio network optimization and planning professionals, and high level managers:

- To do proper optimization by reconfiguring/upgrading the available resources in order to meet the QoS requirement per requested service,

- To dimension expansion and new radio resource capable of providing a level of service to proactively manage future QoS requirement,
- To help understand trend in service type and subscriber behavior which can be used to make high level strategic decision.

1.7 Thesis Outline

The rest of this paper is organized as follows: Chapter 2 explores related works. In Chapter 3, QoS and traffic forecasting in cellular network is discussed. Chapter 4 introduces important deep learning architectures that are suitable for time series forecasting tasks discuses the selected model implementation. The findings are discussed in Chapter 5. Conclusions are drawn and recommendations are made in Chapter 6.

Chapter 2

Literature Review

Cellular network traffic demand forecasting has attracted several researchers with an intent to come up with a technique that can accurately anticipate future demand. This thesis will address few of them related to the proposed title and technique or either.

In [5], LSTM RNN forecasting is used to study, implement and evaluate traffic in cellular networks at base-station level. The main reason the authors preferred LSTM over other types of Neural Networks, is because it has the ability to learn and remember over long sequences and does not rely on a pre-specified window of samples as input. Four months' subscriber traffic data per base-stations were presented by Vodafone. Each base station performs the training and prediction to completely automate strategic resource planning and allocation like energy and bandwidth in advance to improve QoS provided to subscribers. The paper exploits the ability of LSTMs to offload part of its computation to a centralized system, the embedded hardware/server in the base station. By comparing the forecasted result with real values using MSE, LSTM based technique outperformed all the relative forecasting models: ARIMA, SARIMA and RBF(Radial Basis Function), in both accuracy and execution time. This paper shows how good LSTMs perform for time series forecasting to instantly predict traffic and allocate resource in each base station, but didn't address a long term traffic forecasting case.

Traffic forecast is an important tool for both the operators and infrastructure vendors to build their business plan in accordance with market demand which includes traffic volume and technology penetration [6]. In this research the authors propose a mathematical modeling framework based on technology penetration and per-subscriber traffic growth for the mobile broadband traffic growth to be employed in mobile network evolution studies. A quantitative mobile broadband traffic forecast model, mathematically based on the Gompertz function is proposed. The case study used to demonstrate the proposed traffic modeling framework in a dense-urban network deployment scenario of a large European mobile network operator. The authors forecasted the total broadband traffic from HSPA and LTE Release 8 subscriber penetration, subscriber base and subscriber growth functions. The authors pointed out that the proposed traffic forecast modeling framework was shown to be useful for the analysis of network capacity evolution scenarios including assumptions on the operator available network upgrade options. The Gompertz mathematical function used to develop the model in this research is a regressive technique where it depends on the fit of the experimental data and a prior parameter selection is required. Since mobile traffic is a not straight forward data where prior assumption of impact its predictability and also complexity such parametrization, the RNN performs better than the mathematical regression based technique in terms of learning and accuracy.

In [7], Dereje Gemechu used a NAR (Non-linear Autoregressive) ANN technique to forecast voice and data traffic of Addis Ababa City for UMTS network capacity planning for 1 year and 5 years in advance. The purpose of the paper is to develop a forecasting model that can be used for proper planning and rollout of a UMTS network capable of handling a growing traffic demand. The time series neural network model prepared for this study uses non-linear autoregressive (NAR) method. The paper used a 14 consecutive month UMTS traffic data of the City exported from the Performance Reporting System (PRS) and averaged over 24 hours. The forecasting model is used to determine busy hour per user traffic profile pattern of UMTS network ahead of time for Addis Ababa City: CS traffic per user in Erlang, traffic ratio per user, uplink and downlink traffic and traffic ratio per user. These parameters are used to calculate required number of UMTS cells for the City. The author pointed out that the NAR based forecasting model used in the thesis is suitable for predicting time series data learning from the given data without any additional information that can bring more confusion than prediction effect. This research did properly define the impact of traffic forecast on network dimensioning and planning, and its effect on the QoS afterwards. The NAR technique used in this paper can't perform as accurate as LSTM and also the model practically failed to forecast the demand of the town in the time step mentioned in the paper.

In [8], the authors developed a short term forecasting model for UMTS voice traffic using three machine learning techniques of Support Vector Machine(SVM), Multi-layer Perceptron(MLP) and Random Forest(RP) and a statistical forecasting technique of Holt Winters(HW) for commercially deployed UMTS network. A dataset saved every hour between October 24th, 2015 and January 19th, 2016 used for model training and testing. The dataset is divided into two parts. The first part contains 2/3 of the data which is used for model training, whereas the rest of 1/3 data is utilized as a test set. The performance of each forecasting models has been evaluated by calculating MAPE and the authors concluded SVM performs better than the three other techniques. The traditional machine learning techniques are now outperformed by several deep learning techniques such as state of the art LSTM RNN.

In [9], a feed forward ANN used to determine busy hour and develop a forecasting model for a GSM network that can be used for proper equipment dimensioning in the north-central part of Nigeria to proactively respond to QoS degradation due to explosive traffic growth. Five months' traffic data collected from the operator's core network is used for training. The training is done by further dividing the dataset into train, test and validate components and the concluded that ANN has the advantage of learning any network and also does the prediction as long as correct data are supplied. This paper gives a very good insight about how a proper forecasting could help the mobile operator to anticipate QoS degradation and act proactively but there are techniques which perform better than the feed forward ANN technique.

In [10], the authors predicted UMTS data traffic for purpose of design and management of cost effective and high quality network. They used the SARIMA model as an alternative way of forecasting UMTS data traffic of the city of Addis Ababa, Ethiopia. After analyzing and discovering the seasonality of daily UMTS data collected they proposed it is appropriate to use the linear model of SARIMA to develop the forecasting model. The authors forecasted a one month ahead prediction and used MAPE and APE to evaluate the model accuracy against true value. As per the literature a MAPE of 1.17% and APE of 2.62% prediction accuracy is achieved which is believed to be fair.

In [11], the strengths of ARIMA and LSTM in medium to long-term traffic prediction is investigated. The performance of these predictors is evaluated using an RMSE. It is observed that when the length of observation increase beyond a given threshold the LSTM predictor outperforms the ARIMA. Furthermore, it is also observed that the LSTM outperformed the ARIMA techniques when the length of future prediction increase.

In [12], the authors examined the performance of SARIMA, CNN and LSTM time series models for predicting stock market movements. using the dataset from Nifty-500 indices. Outcome of the comparison shows the deep learning based models especially LSTM to be promising over the tradition SARIMA based technique. MSE is used as a metrics to compare the performance and the result shows 0.003842, 0.003369 and 0.003300 MSE value for SARMA, CNN and LSTM respectively.

Minimum performance improvement: MSE = -0.000069

In [13], the performance of LSTM, NNAR, and SARIMA time series models to predict the shoreline variations from surveillance camera images in multistep predictions is investigated. R, RMSE, MAE and MAPE are used to evaluate the performance of each model and it is demonstrated that as prediction step grows LSTM outperformed all the other models. For the longest prediction step in the literature which is 50 the R, RMSE, MAE and MAPE of SARIMA: 0.101, 12.693, 11.146 and 72.603%, NNAR: 0.163, 12.275, 10.715 and 68.708% and LSTM: 0.088, 12.224, 11.239 and 63.004%.

Minimum performance improvement: R = -0.075 RMSE = -0.469, MAE = +0.093, MAPE = -5.704%,

In [14], the authors employed LSTM, Logistic Regression and Random Forest time series forecasting models to predict stock data which is considered complex, fickle, and dynamic, as a result, challenging to predict. The models are evaluated based on RMSE and MAE. The models average shows that RMSE and MAE for LSTM: 0.799 and 0.537, Logistic Regression: 1.431 and 0.996, and Random Forest: 1.395 and 0.971.

Performance improvement: RMSE = -0.632, MAE = -0.459

In [15], the research compares the difference between performance of the well-known RNN techniques: GRU and LSTM. The difference in performance is evaluated based on prediction accuracy between the two models. It is demonstrated that LSTM outperforms GRU as the size of training dataset grows. Model performance is done based on F1 and AUC (Area Under the Curve) and AUC and F1 respectively of LSTM: 96.04

Performance improvement: F1 = +9.06%, +5.95%

In [16], A time series forecasting models are developed using LSTM deep learning to predict single step and multistep ahead short-term electrical load. GRNN and ELM machine learning based techniques are used to evaluate the evaluate the performance of LSTM. In both single step and multistep predictions LSTM is found to outperform both techniques interns of MAE, RMSE and MAPE. It is demonstrated that MAE, RMSE and MAPE respectively of LSTM: 17.11, 22.65 and 1.52 in single step ahead and 55.42, 63.81 and 4.79 in multistep, ahead, GRNN: 34.35, 45.98 and 3.05 in single step ahead and 61.62, 68.45 and 5.33 in multistep ahead, ELM: 36.59, 44.52 and 3.44 in single step ahead and 73.82, 80.56 and 6.86 in multistep ahead.

Minimum performance improvement single step ahead: MAE = -17.24, RMSE = -23.33, MAPE = -1.53 Minimum performance improvement multi step ahead step ahead: MAE = -6.2, RMSE = -4.64, MAPE = -0.54

In [17], the authors explored how deep learning based algorithms for time series forecasting are superior to the traditional algorithms. The study is conducted models based on LSTM deep learning model and ARIMA based traditional model. It is shown that LSTM based algorithms outperforms ARIMA based algorithms in time series forecasting. Data sets from several discipline/industry are used to develop the models their performance is evaluated using RMSE. The average model performance over the various datasets using RMSE is found to be 0.936 for LTSM and 5.999 for ARIMA.

Performance improvement: RMSE = -5.063

The above reviewed researches provided an excellent insight on developing a traffic forecasting model using various time series forecasting techniques and how traffic forecasting can help improve service provisioning and build market based business plan from both MNOs and equipment vendors point of view. With all the strengths of the reviewed papers, the proposed research uniquely approached the case from one or, combination of two or more of the ideas: better technique, different purpose and different type of traffic data for training, validation and prediction. Furthermore, the explored related work on RNN LSTM showed that significant performance improvement is observed when using LSTM based time series forecasting over the platforms considered for comparison.

Chapter 3

Quality of Service Management and traffic forecasting in Cellular Networks

The two aspects that the operator must consider to manage the required/expected QoS are ability of the network to support the requested service with an assured service level and what the end user really perceives. The first one is knowledge about the required amount and mix of traffic that the network is able to support at a given point of time considering all the impairments meet the required QoS. In order to 'know' the amount and mix of traffic in advance, having an appropriate traffic forecasting tool is required[3].

Cellular network traffic forecasting is a technique to predict future traffic and service trend. Understanding the dynamics of the traffic demands in a mobile network in advance is a fundamental importance to MNOs and equipment vendors since it drives the fundamental business planning, network dimensioning and optimization. As a result, mobile traffic forecasting is often of tremendous commercial value[6]. In this section QoS and traffic forecasting in cellular networks will be discussed.

3.1 Overview of the cellular Network

The cellular network technology started as an analog system for voice only services. The technology evolved in to delivering text and data services in addition to voice. Cellular network technology evolution rapidly evolved derived by the emergence and proliferation of data hungry applications and devices. The most important change between the various generations of cellular networks is the technology implemented over the air interface and the network architecture[18].

The first generation mobile networks emerged in the 1980s and the late 2000s marked the end of these networks. The second generation(2G) cellular system is the first digital telephony which introduced text and data services. GSM is the widely known 2G system standardized by European Telecommunications Standards Institute(ETSI). The GSM technology is based on a combination of FDMA and TDMA techniques. It supported data services by the GPRS technology and later due to increased demand for data services evolved to EDGE which supported a theoretical maximum 384Kbps DL user throughput[18].

The third generation cellular system(3G) which is called UMTS significantly improved the data rate offered and marked the introduction of new services such as video telephony, multimedia, audio streaming, email and internet browsing to the rapidly growing telecommunications ecosystem. In January 1998 the European standardization body ETSI decided upon WCDMA as the third-generation air interface [19]. Detailed standardization work has been carried out as part of the 3GPP standardization process.

In UMTS each licensed band is formed of 5MHz width that supports 3.84 Mega chips per second chipping rate within each time slot. With these design decisions, data transfer for downlink connection is 384 Kilobit per second (Kbps) for the first release, and a maximum of 21.6 Megabit per second (Mbps) per single cell/cerrier in later releases where the HSPA technology was introduced for the downlink. For uplink, the data transfer ranges from 384 Kbps to 5.76 Mbps theoretically. 3GPP specified important evolution steps on top of WCDMA: HSPA and then HSPA+

(advanced HSPA) for improved data rate delivery. In the ideal scenario of user equipment category and radio condition a single user can achieve a peak data rates up to 57Mbps in the downlink and 5.76Mbps in the uplink, using a combination of air interface improvements as well as multiple cell and MIMO (multiple input multiple output) features in the latest release of 3G/UMTS [20].

The all IP networks: LTE, LTE-A and 5G emerged to deal with the growing demand for mobile data traffic. In addition to the change on the air interface technologies these two system simplified the network architecture. The LTE/LTE-A systems used Orthogonal Frequency Division Multiple Access (OFDMA) in the downlink and Single Carrier Frequency Division Multiple Access (SCFDMA) in the uplink to achieve a maximum of 100 Mbps data throughput with a single carrier. In addition to OFDMA the 5G systems used power domain for multiple access to achieve a downlink data throughput of 1Gbps[**?**].

3.2 QoS in Cellular Networks

Mobile networks which by nature have finite resources are designed to support wide range applications with different quality of service requirements[3]. Mobile/-cellular network resources which include the radio spectrum and the back-haul transmission networks are expensive and must be shared between multiple users. The dramatic increase in the number of mobile devices trying to access the mobile networks in the past have increased the amount of traffic to be carried over the networks as well. Due to the above reasons it is economically impossible to have a radio network that is capable of carrying all the traffic requested at any point in time. Allowing all users to access the network infrastructure based on their will, will result in congestion and when the requested amount exceeds the network capacity it will cause blocking of other requests resulting in customer dissatisfaction.

To overcome this problem 3GPP came up with an idea of controlled portioning of the available radio resources for fair-use policy that limit service abuse by few users [3]. This is achieved by assigning each service a given network defined QoS profile used when setting-up, modify and maintain connections. In this way the operators can ease network congestion, improve service quality and even to create a frame work for business model.

The 3GPP defined QoS generically refers to the quality of a requested service as perceived by the user of the service [21]. In its technical specification 3GPP defines QoS requirements for various services to be carried by each mobile radio access technologies(RATs) i.e. GSM, UMTS. LTE, LTE-A and 5G. In the 3GPP Release 99(R99), first release with QoS functionality, a new QoS handling is introduced which aligned with the emerging UMTS network defined QoS. The R99 platform provided QoS for both voice and data, with priority given to voice. It allowed operators to have a radio resource management algorithm in each RATs endowed with QoS differentiation to enable operators to offer cost effective services and improve service quality. The R99 UMTS QoS functionality maps services with different requirements onto distinct QoS profiles defined by a subset of the bearer attributes, which are bit rates, priorities and traffic classes(TC)[21].

In 3GPP release 15 QoS functionality, it is made possible to apply QoS control on a per service data flow basis. The specific realization is achieved by having an architecture to support control of QoS reservation procedures for the existing IP-CAN (IP Connectivity Access Network) and the emerging 5G(NR) technology. The specific QoS flow will be mapped on to IP-CAN bearer based on its QoS attributes/criteria such the QoS subscription information, service based policies, and/or predefined internal policies [22].

3.3 Essentials of Cellular Network Traffic forecasting

Cellular network traffic forecasting is a time series forecasting problem where one or more future values are forecasted based on historic values as independent input variables[23]. The purpose of time series forecasting is to estimate how the sequence series will behave into the future. Time series data is a set of quantities that are collected over even intervals in time and ordered chronologically. The time interval at which data collected is generally referred to as the time series frequency and any time interval for which measurement does not exist is set to the missing value [24].

3.3.1 Characteristics of Time series

Analyzing the characterstics and patterns of the time series data can be a great help in building a reliable model. Time series data can exhibit a variety of patterns, and it is often helpful to decompose a time series into several components, each representing an underlying pattern category[24]. Time series are usually characterized by additive or multiplicative combination of three components: trend, seasonality, and irregular components, also known as residuals [25].

Trend: It is the general movement that the time series exhibits during the observation period, without considering seasonality and irregularities. It is a general systematic linear or nonlinear component that changes over time and does not repeat. Although there are different kinds of trends in time series, the most popular are linear, exponential, or parabolic ones [24],[25].

Seasonality: This component identifies variations that occur at a fixed and known frequency. It might provide useful information when dealing with the time series data. It is a general systematic linear or nonlinear component that changes over time and does repeat [24],[25].

Residuals: Residuals are random fluctuations or can be considered as noise components. Once the trend and cyclic oscillations have been calculated and removed, some residual values remain. Sometimes the random fluctuations might be high enough to mask the trend and the seasonality. In this case, the term outlier is used to refer these residuals and. Sometimes these can be fluctuations that can make the prediction almost impossible in which case a proper residual analysis and modelling has to performed [19].

In order to estimate the effects of each these components a decomposition can be applied to the time series.

If we assume an additive decomposition, then we can write [24]

$$y_t = S_t + T_t + R_t \tag{3.1}$$

where y_t is the data, S_t is the seasonal component, T_t is the trend-cycle component, and R_t is the residual component, all at period t. Alternatively, a multiplicative decomposition would be written as[24]

$$y_t = S_t * T_t * R_t \tag{3.2}$$

The additive decomposition is the most appropriate if the magnitude of the seasonal fluctuations, or the variation around the trend-cycle, does not vary with the level of the time series. When the variation in the seasonal pattern, or the variation around the trend-cycle, appears to be proportional to the level of the time series, then a multiplicative decomposition is more appropriate [24]. Time series component analysis and modelling will be very important especially if the technique going to be deployed is statistical.

The dataset used for this research is decomposed and in both traffic categories it shows that there is a weekly seasonality. The voice traffic experiences high residual level which will obviously makes it less predictable.

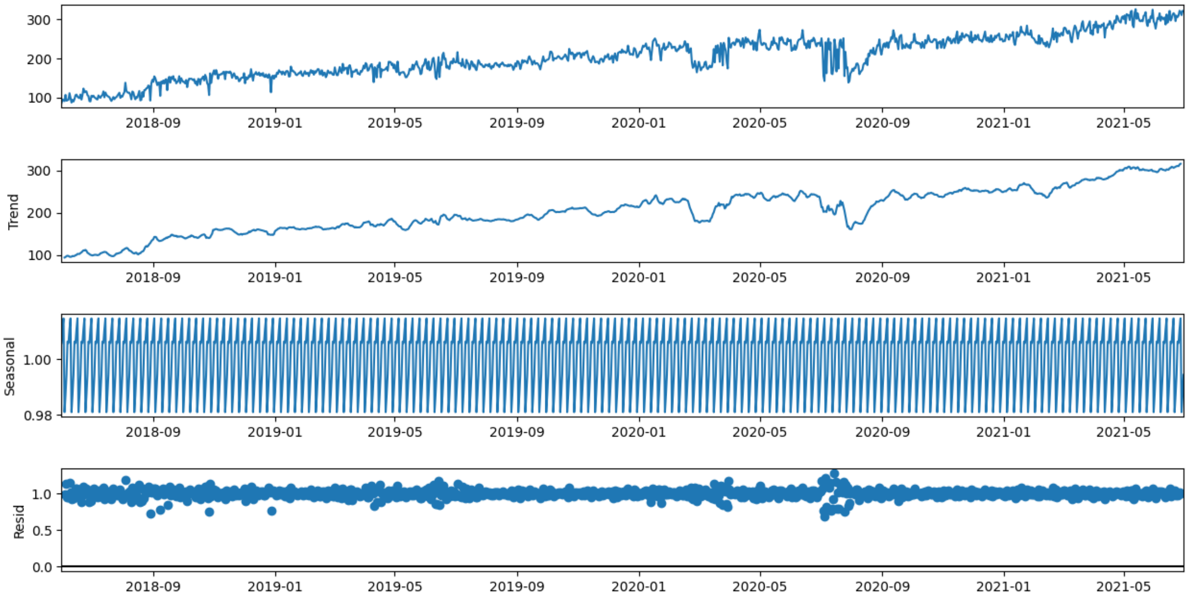

FIGURE 3.1: Time series component decomposition of Jimma town data traffic

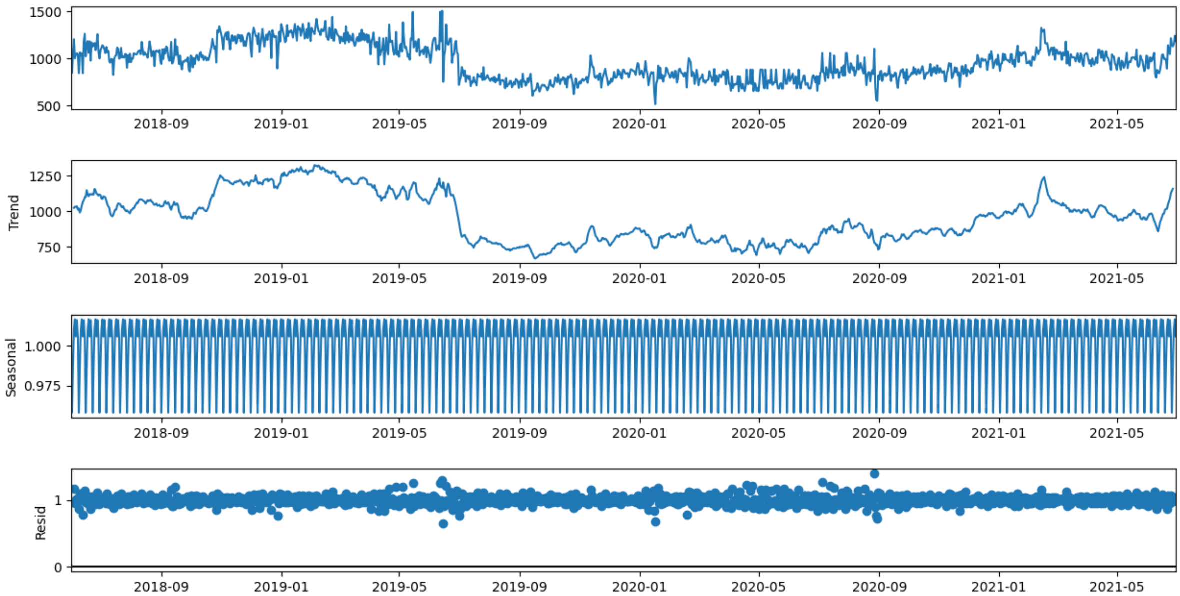

FIGURE 3.2: Time series component decomposition of Jimma town voice traffic

3.3.2 Time Series modeling

Time series forecasting requires the ability to understand temporal dependencies between the series. When we came up with time series forecasting problem, it is crucial to narrow it down and decide the technique used to model the problem. Time series analysis is about identifying the intrinsic structure and extrapolating the hidden traits of your time series data in order to get helpful information from it [25].

In time series data set, the time column is a primary structure used to order a data set. Due to its chronological order, this primary temporal structure makes time series problems more challenging when there is a need to apply specific data preprocessing and feature engineering techniques to handle time series data. Missing data and outliers are two the most important challenges related the temporal structure of time series that have to be taken in to account when modelling the time series. Data collected at irregular time interval time intervals is also another challenge for prediction .Before getting started with building their forecasting solution, it is highly recommended to analyze and define the following points [26]:

The inputs and outputs of your forecasting model – Inputs are historical time series data provided to feed the model in order to make a forecast about future values. Outputs are the prediction results for a future time step. [26].

The univariate or multivariate nature of your forecasting model – Time series forecasting can either be univariate or multivariate. Univariate involves the analysis of a single independent variable during modelling. Multivariate analysis examines multiple independent variables are observed simultaneously in the time series. Multivariate modelling incorporates the past observation of other independent variables. [26]

Single-step or multi-step structure of your forecasting model – Single step structure is used to predict the observation at the next time step. Multiple-step or multi-step time series problems, where the goal is to predict a sequence of values in a time series [26].

Contiguous or noncontiguous time series values of your forecasting model – A time series that present a consistent temporal interval between each other are defined as contiguous. On the other hand, time series that are not uniform over time may be defined as non-contiguous. the reason for time series to be noncontiguous is missing data due to several reasons. Since noncontiguous time series data are difficult to model the missing points are imputed most of the time. Below are the three most common reasons why data goes missing [26]:

- Missing at random: This happens when when the data point to be missing is/are not related to the unobserved data rather it is related to some of the observed data.
- Missing completely at random: This case is when the missing are is/are independent of the observed and unobserved series. This means there is no relationship between a certain missing value and its hypothetical value and also with the other observed and unobserved values.
- Missing not at random: The possible reasons for this type of missing data is that the missing value is/are related to the unobserved data or variable which are not measured. This type is a non ignorable since the cause is not known.

In the first two cases the observed data will contain series that will represent the information in the missing ponts as a result, it is safe to remove the missing values depending on the amount/percentage of occurrences , while in the third case removing observations with missing values can produce a bias in the model. When dealing with missing data imputation using a carefuly selected technique is always a preferred choice over removing. There are different techniques to impute the missing variables depending on the kind of problem we are trying to solve, and it is difficult to provide a general solution [26].

Chapter 4

Deep Learning Based Cellular Network Traffic Forecasting

Cellular network traffic forecasting is concerned mainly with studying user level, cell level, node level, service level or network level future traffic demand with the aim of improving service provisioning. This is achieved by having an appropriate traffic forecasting model capable of predicting future demand with a required level of accuracy based on history data.

Mobile Network Operators have to use a forecasted traffic to decide on network upgrade path and evolution which is paramount for QoS service provisioning and achieve the optimum ROI. A reliable forecasting will help operators to proactively manage user satisfaction and make sound strategic decisions. In this thesis, a cellular network traffic forecasting model is developed that can be used to determine a network level traffic pattern on both PS and CS traffic categories. The historical data which is a network level daily busy/peak hour traffic data used to train and test the model is collected from Jimma own UMTS cellular network. Using the historical data, a deep learning based traffic forecasting model is developed.

4.1 Deep Learning Architectures

In order to handle the increasing variety and complexity of forecasting problems, many techniques have been used to develop a model that can handle the sequential dependencies of time series attribute. Each has its special use, and care must be taken to select the correct technique for a particular application. The selection of a specific technique depends on many factors such as the resources and data available, the accuracy of the competing models, the way in which the forecasting model is to be used, the time period to be forecasted and the cost/ benefit of the forecast to the company [24]. The model to be used in forecasting depends on the resources and data available, the accuracy of the competing models, and the way in which the forecasting model is to be used

There are wide range of forecasting techniques to choose from when selecting a model. Among these are the statistical techniques of SARIMA (Seasonal Autoregressive Integrated Moving Average) and ARIMA (Autoregressive Integrated Moving Average), the traditional machine learning techniques such as SVM (Support Vector Machine) and MLP (Multilayer Perceptron) and neural network/deep learning based techniques like ANN-RNN and NAR (Non Linear Auto-regression) can be mentioned. The main drawbacks of the statistical techniques are the poor robustness to the rapid fluctuations of the time-series since they assume linear relationship between past values. Drawbacks of traditional machine learning techniques are that they require human intervention to learn, are not good with large training data, and sensitive to hyper parameters [25]. The ANN based deep learning techniques which have a programmable network that enables machines to make accurate decisions without help from humans and are capable of nonlinear modeling and have outstanding performance as the scale of training data increases [15],[27]. The deep learning techniques are also preferable for time series forecasting in many areas due to their ability to capture complex and nonlinear patterns in the data.

The ANN is a popular machine learning subfield that simulate the mechanism of learning in biological organisms. Neurons are the fundamental computational unit of ANN where the processing/mapping of inputs to outputs is performed. The connection between the neurons are called weights which scale each input and affect the computed value at each activation function evaluated in the neurons. The relationships between the neurons of two consecutive layers are modeled by the weights, which are calculated during the training phase of the network [28]. This is done during the optimization process performed to minimize the loss function.

The basic idea of a neuron model is that an input, x, together with a bias, b(optional) is weighted by, w, and then summarized together. The bias, b, is a scalar value whereas the input x and the weights w are vector valued, i.e., $x \in R^n$ and $w \in R^n$ with $n \in N$ corresponding to the dimension of the input. Note that the bias term is not always present and is sometimes omitted. The sum of these terms, i.e., $z = w^T x + b$ forms the argument of an activation function, φ , resulting in the output of the neuron model[29]. Mathematically:

$$y = \varphi(z) = \varphi(w^T x + b) \tag{4.1}$$

Neurons perform a two stage process to map inputs to outputs: calculation of a weighted sum of the inputs to the neuron and then passing the result through a non-linear function called 'activation function' that maps the results of the weighted sum score to the neuron's final output value called 'activation value'. An artificial neural network computes a function of the inputs by propagating the computed values from the input neurons to the output neuron(s) and using the weights as intermediate parameters [28],[29]. This process is called forward pass.

Architecture of a neural network can be single layer where a set of inputs is directly mapped to an output by using a generalized variation of a linear function or a multi-layer neural networks where the neurons are arranged in layered fashion in which the input and output layers are separated by a group of hidden layers made up of neurons and weights for each connections. A standard neural network consists of many connected neurons each capable of evaluating a function and producing a sequence of real valued activations [27],[28]. Mathematically:

$$z = \sum_{i=1}^{n} w^T x + b \tag{4.2}$$

$$\bar{y} = \varphi(z) \tag{4.3}$$

Where x is the input(independent variable), w is the weight associated with each input to the neurons in the T^{th} layer, φ is the non-linear activation function and y is output(dependent variable) of the neuron.

A layer is the core building block of neural networks, a data-processing module that behaves as a filter for data. An input data goes in, and it comes out in a more useful form by extracting meaningful representations from the input. Multilayer neural networks contain multiple computational layers; the additional intermediate layers (between input and output) are referred to as hidden layers because the computations performed are not visible to the user [24].

4.1.1 Training a Deep Learning Neural Network

Training a neural network is a supervised learning process which involves using input called features and output training dataset to update the trainable parameters. Supervised Learning mainly divided into regression and classification. Regression is the type of Supervised Learning in which labelled data is used to make predictions in a continuous form. Classification is the type of Supervised Learning in which labelled data can use, and this data is used to make predictions in a non-continuous form[26],[28].

The training of neural networks is an iterative process that solves an optimization problem to find for parameters (weights) that result in a minimum error or loss when evaluating the examples in the training dataset. The weights are the primary trainable parameters which contain the information learned by the network from exposure to training dataset.

To understand the training neural networks it is appropriate to discuss the below important concepts:

Supervised Learning

It consists of learning to map input data to known targets, given a set of examples. Generally, almost all applications of deep learning that are in the spotlight these days belong in this category [28]

Activation functions

An activation function is a function that performs non-linear transformation in order to help the network learn complex patterns in the data. In order to get access to a much richer hypothesis space that would benefit from deep representations, we need a non-linearity, or activation function [27],[28]. Two of the most commonly used activation functions deep learning architectures:

Sigmoid: takes any real value(x) and gives an output in the range (0,1). It is the most widely used activation function. Mathematically:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{4.4}$$

Tanh: this function as well takes any real value(x) and gives an output in the range (-1,1). tanh function is very similar to the sigmoid/logistic activation function, and even has the same S-shape with the difference in output range of -1 to 1[28]. Mathematically:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{4.5}$$

Loss function

The quantity that will be minimized during training. It represents a measure of success for the task at hand. It is imperative to select the appropriate loss function for the neural network since the neural networks will be just as ruthless in lowering their loss function, or we'll have to face unintended side effects[27].

There are simple guidelines to follow when choosing the correct loss: for instance, you'll use binary cross-entropy for a two-class classification problem, categorical cross-entropy for a many-class classification problem, mean-squared error for a regression problem, connectionist temporal classification(CTC) for a sequence-learning problem, and so on [28].

Optimizers

Optimizers are algorithms used to tune trainable parameters such as weights of the network in order to minimize the loss function and provide most accurate results possible. It determines how the network will be updated to reduce loss function. It implements a specific variant of stochastic gradient descent (SGD). There are different types of optimizers which have their own characteristics in terms of what type of neural network application they are used, and their advantage and disadvantage. To mention few of them: Stochastic Gradient Descent(SGD), RMSProp, Adam and Adagrad [27][28].

Backpropagation

A neural network function consists of many tensor (data container) operations chained together, each of which has a simple, known derivative. Calculus tells us that such a chain of functions can be derived using the chain rule. Applying the chain rule to the computation of the gradient values of a neural network gives rise to an algorithm called Backpropagation (also sometimes called reverse-mode differentiation) [25]. Backpropagation starts with the final loss value and works backward from the top layers to the bottom layers, applying the chain rule to compute the contribution that each parameter had in the loss value. The back-propagation algorithm first uses a forward phase in order to compute the output and the loss. Therefore, the forward phase sets up the initialization recurrence, and also the intermediate variables that will be needed in the backwards phase [28].

The backward phase computes the gradient of the loss function with respect to various weights. The first step is to compute the derivative the loss function. If the network has multiple outputs, then this value is computed for each output. This sets up the initialization of the gradient computation. Subsequently, the derivatives are propagated in the backwards direction using the multivariable chain rule and the weighs are updated[28].

In chain rule the gradient of error/loss function will be multiplied through the backward phase. The weight which is the training parameter is updated proportional to the gradient. With the conventional backpropagation error signals(gradients) flowing backward in time tend to (1) blow up or (2) vanish. This problem is called exploding or vanishing gradient. Vanishing gradient occurs when the gradient gets smaller and smaller as we go backward with every layer during backpropagation. Exploding gradient occurs when the gradient gets larger and larger as we go backward with every layer during backpropagation[27].

Hyperparameters

A hyperparameter is a parameter whose value is specified by the developer to control the learning process for a give dataset. Hyperparameter values can be arbitrarily set by the user before starting the training process. To mention few of the hyperparameters: number of nodes(neurons), number of hidden layers, number of units in a dense layer, dropout, number of epochs, batch size and optimization techniques[28]. There are a number of deep learning techniques of which their selection depends on the specific use case. This section will discuss selected architectures suitable for time series forecasting.

4.2 Deep Learning Architectures for Time Series Forecasting

4.2.1 Feed Forward Neural Network

A Feed Forward Neural Network(FFNN) is an artificial neural network in which information is only processed in the forward direction. There are no feedback connections in which outputs of the model are fed back into itself. The simplest FFNN is the perceptron with no hidden units and only an input and an output layer. The output units are computed directly from the sum of the product of their weights with the corresponding input units[21]. Practically the perceptron is used to predict only a binary value of 0 or 1.

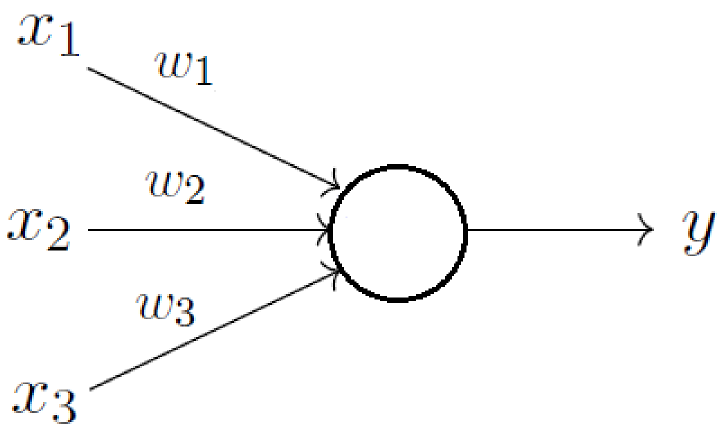

FIGURE 4.1: Basic architecture of a perceptron

The multilayer perceptron(MLP) which is also known as deep FFNN is an ANN made up of many perceptron. In multilayer perceptron successive layers feed into one another in the forward direction from input to output. [26].

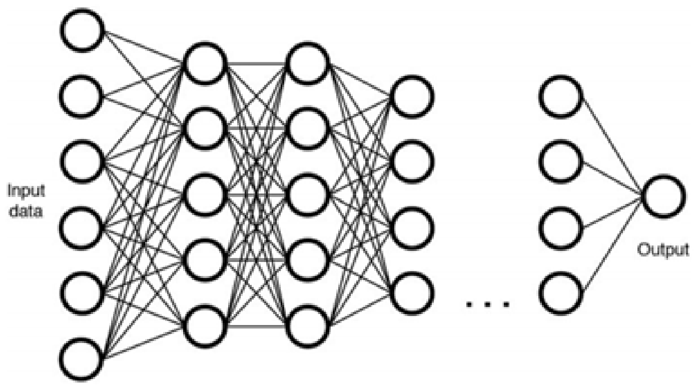

FIGURE 4.2: basic Deep FFNN architecture[26]

Since they do not allow feedback FFNNs are not good in capturing dependencies/information contained in historical information. As a result, they are not efficient with data that requires learning to extract sequential or time-dependent pattern[28].

4.2.2 Recurrent Neural Network

The FFNN are designed for data in which the attributes are largely independent of one another. Recurrent neural networks on the other hand are a class of neural networks that allow previous outputs, in addition to the current input, to be used as inputs while having hidden states. Recurrent neural networks (RNNs) are suitable for data types which contain sequential dependencies such as time series dataset. The basic RNN is defined as:

$$\bar{h}_t = f(w_{xh}\bar{x}_t + w_{hh}\bar{h}_{t-1}) \tag{4.6}$$

$$\bar{y}_t = w_{hy}\bar{h}_t \tag{4.7}$$

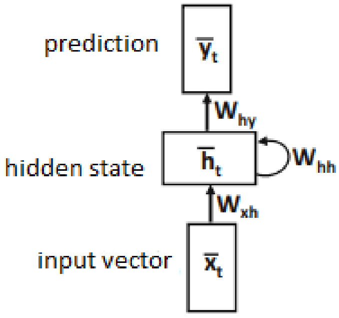

FIGURE 4.3: A recurrent Neural Network [21]

Where f is the activation function, $\bar{h}_t$ is hidden state at time t, $\bar{h}_{t-1}$ is saved hidden state from previous computation, $\bar{x}_t$ is input vector at time t, w_{xh} is the input weight matrix, w_{hh} weight matrix from previous hidden state, w_{hy} is the output weight matrix and tanh is the activation function[27].

The basic RNNs suffer from vanishing and exploding gradient as the layers get deeper or number of layers in the architecture increase. To overcome this problem two specialized versions of RNN were created capable of maintaining information in the memory for the long period of time: LSTM and GRU [25][27].

LSTM

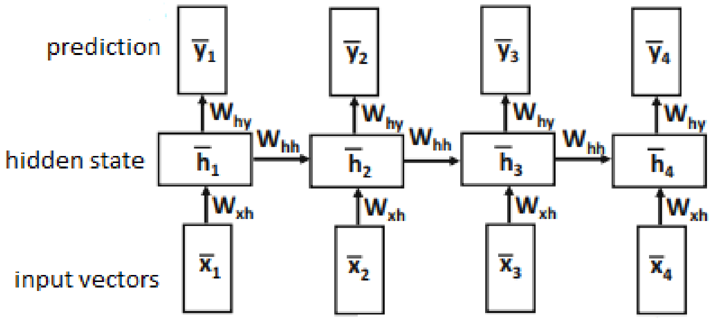

FIGURE 4.4: RNN time layered representations [27]

To address the problem of vanishing and exploding gradient, a solution is to change the recurrence condition for the hidden vector $\bar{h}_t^k$ with the use of long-term memory. In order to achieve this goal, we have an additional hidden vector of k dimensions, which is denoted by $\bar{c}_t^k$ and referred to as the cell state. One can view the cell state as a kind of long-term memory that retains at least a part of the information in earlier states by using a combination of partial "forgetting" and "increment" operations on the previous cell states. The operations of the LSTM are designed to have fine-grained control over the data written into this long-term memory [27].

LSTM contain an internal mechanisms called gates that have the ability to remove or add information to the cell state, carefully regulated by structures called gates. The gates are used to update and persistently store relevant information to the cell state by using a combination of partial "forgetting" and "increment" operations on the previous cell states. One can say the purpose of the gates is to optionally let information through [25][29]. These gates which helps to perform the entire operation are input, forget, and output variables, because of the roles they play in updating the cell states and hidden states [19][21][24]. Each gates have weights associated with them.

In figure 4.5, C_{t-1} is the previous cell state, h_{t-1} is previous hidden state, x_t is current input to the network[28]. The gates and their output are discussed below.

1. Forget gate

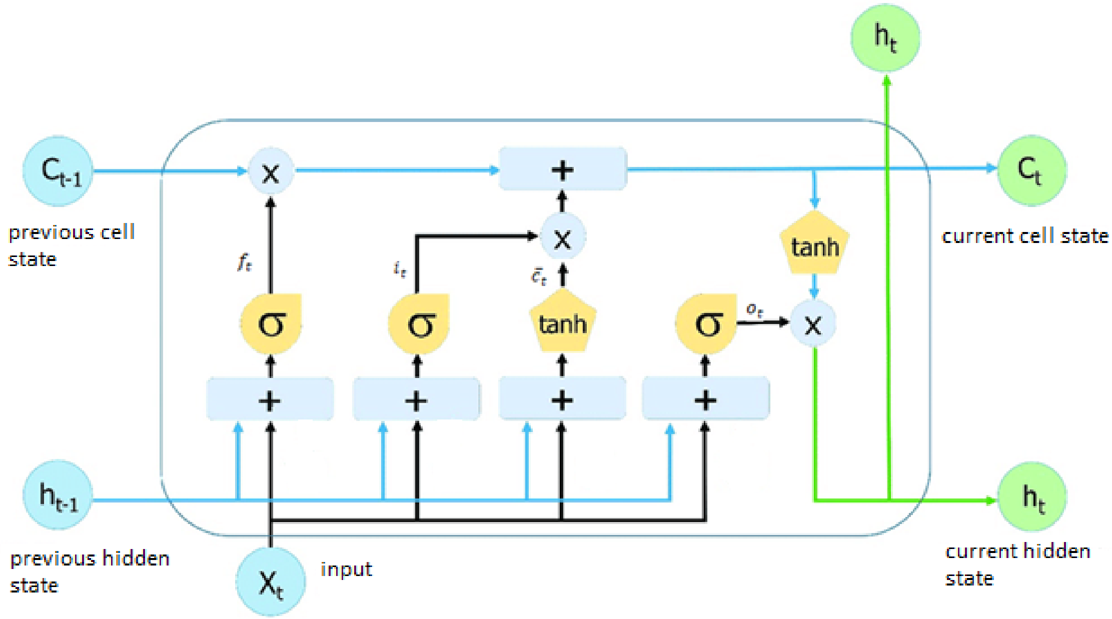

FIGURE 4.5: LSTM architecture [28]

This gate is responsible for throwing away information that is no more required or that is of less importance, or keeping information that is relevant to the system by multiplication of a filter. As seen in the above diagram, first information from the previous hidden state and information from the current input is passed through the sigmoid function. Output of the sigmoid is a value between 0 and 1 which will be multiplied with the previous cell state. Multiplication of the previous cell state with a value closer to 0 means to forget, and the closer to 1 means to keep [25],[27]. Mathematically:

$$f_t = \sigma_g(W_{xf}x_t + W_{hf}h_{t-1}) \tag{4.8}$$

2. Input Gate

The input gate is responsible for updating the cell state. As it can be seen from the diagram above, first we pass the previous hidden state and current input into a sigmoid function. Output of the sigmoid function, a value between 0 and 1, decides information that is relevant and non-redundant. 0 means not important, and 1 means important. Passing the previous hidden state and current through a tanh function helps to squish values between -1 and 1 helps regulate the network.

Then the tanh output is multiplied with the sigmoid output. The sigmoid output will decide which information is important to keep from the tanh output [25],[27]. Mathematically:

$$i_t = \sigma_g(W_i x_t + W_{hi} h_{t-1}) \tag{4.9}$$

$$\bar{c}_t = tanh(W_{xc} x_t + W_{hc} h_{t-1}) \tag{4.10}$$

3. Output gate

The output gate is responsible for selecting useful information from current cell state to decide what the next hidden state should be. First, we pass the previous hidden state and the current input into a sigmoid function. Then we pass the newly modified cell state to the tanh function. We multiply the tanh output with the sigmoid output to decide what information the hidden state should carry. The output is the hidden state. The new cell and hidden state is then carried over to the next time step. The output gate is is used to select useful information from the current cell state. [25],[27]. Mathematically:

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1}) \tag{4.11}$$

4. Cell State

Finally, the information from the forget gate, input gate are used to calculate the cell state. Calculating the cell state is the core concept of LSTM [25],[27]. Mathematically:

$$c_t = f_t \bullet c_{t-1} + i_t \bullet tanh(\bar{c}_t) \tag{4.12}$$

The current hidden state will be computed as:

$$h_t = o_t \bullet tanh(c_t) \tag{4.13}$$

GRU

The GRU is the newer generation of Recurrent Neural networks and is pretty similar to an LSTM. The GRU can be viewed as a simplification of the LSTM, which does not use explicit cell states [27]. GRU's got rid of the cell state and used the hidden state to transfer information. It also only has two gates, a reset gate and update gate [31].

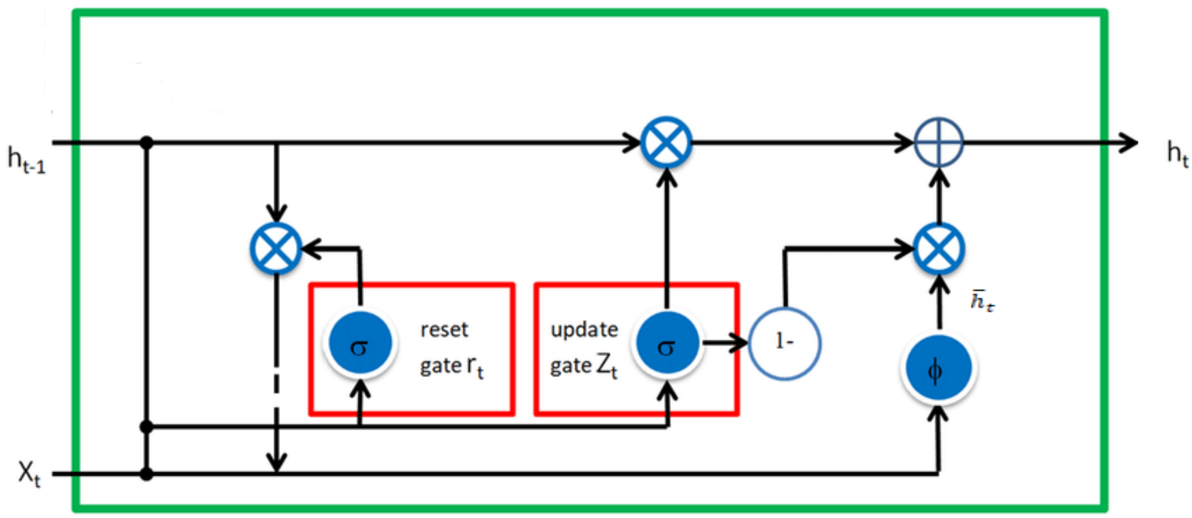

FIGURE 4.6: GRU architecture [25]

In figure 4.6, h_{t-1}, X_t and h_t are previous hidden state, current input to the network and current hidden state respectively.

Update Gate

In GRU the update gate achieves the combined functionality provided by forget and input gate of an LSTM. It decides what information to throw away and what new information to add [31].

$$z_t = \sigma(x_t W_{xu} + h_{t-1} W_{hu}) \tag{4.14}$$

The update gate is responsible for determining the amount of previous information that needs to pass along the next state.

Reset Gate

The reset gate is responsible to decide the amount of past information to neglect [31]. Mathematically:

$$r_t = \sigma(x_t W_{xr} + h_{t-1} W_{hr}) \tag{4.15}$$

The intermediate hidden state:

$$\bar{h}_t = tanh(Wx_t + r_t \bullet U\bar{h}_{t-1}) \tag{4.16}$$

The final hidden state used to transfer information:

$$h_t = (1 - z_t)\bullet\bar{h}_t + z_t \bullet h_{t-1} \tag{4.17}$$

Since GRU's has fewer mathematical(tensor) operations; therefore, they are a little speedier to train then LSTM's.

In addition to feed forward mode the basic RNNs, LSTMs and GRUs can be configured to work in both as a Bidirectional Recurrent Neural Networks (BRNNs) where to predict a value of a data sequence in a given instant of time, information from the sequence both before and after that instant is needed. A BRNN can be seen as two RNNs together, where the different hidden units have two values, one computed by forward and another one by backward[25].

4.2.3 Implementation of the Proposed Deep Learning Based Forecasting Model

The proposed technique for this study uses an RNN LSTM deep learning technique architecture. The model is developed to forecast future values of a time series y from x past values of historical time series data. This is a regression problem where past independent variable is used to predict a future dependent variable. The model is implemented in python Keras as a front end and TensorFlow as a back end.

The deep learning time series forecasting is a supervised learning where a given historical data is used to train a model. As discussed in chapter 2, time series problem can be a univariate or multivariate. In our case since the traffic arrival on the same cellular technology is found out to be uncorrelated the problem is approached as a univariate time series. Let X = x_1, x_2, $\ldots$, x_n, x_{n+1} ,..., x_{n+k} be the given univariate historical data. For supervised learning these data are labeled as samples and target for the deep learning to be trained on. The sample set which contains x =x_1, x_2,$\ldots$,x_n is trained to predict a target k number of time series sequemce y = x_{n+1}, x_{n+2}, $\ldots$ x_{n+k} denoted by $\bar{y}$ with the purpose of reducing the prediction error which is defined by a given loss/objective function. The best prediction is achieved when the predicted value $\bar{y}$ is nearest or equal to the target value.

The deep learning model has input, hidden and output layers to take the input tensors, to perform nonlinear transformation and provide an output time series prediction during training. These separate layers of the system model are indicated in Figure 4.7. The number of LSTM nodes in the input and output layers are automatically decided by the system based on the number of input samples and model type, which is regression in this thesis.

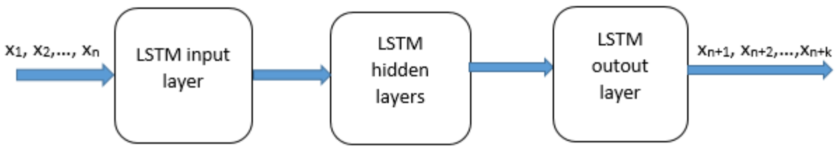

FIGURE 4.7: Model training block diagram

During model fitting, a sample of dataset is held for validation which is called validation set and is used to evaluate model performance. The validation set is believed to be new dataset which the model is not exposed to it before. For a given combination of hyperparameters, the model training is performed until a stable decreasing and converging training and validation loss is achieved and is stopped when the validation loss starts ascending or the two curves no more show a decreasing trend.

To protect the model from overfitting and under-fitting, ultimate care is taken when training the model by carefully cleaning noises/residuals in the time series data and selecting the hyperparameters during tuning so that the difference between training and validation loss is the minimum possible and the training is stopped when the validation error trend changes from decreasing to increasing. Furthermore, the model performance on new dataset is checked against test data.

A combination of hyperparameters are used to train the model across the three traffic categories and the final result is presented in the result and discussion part. Due to the unique characteristics of each traffic type due to user behavior, the result of model hyperparameter tuning are different. In each traffic categories the important hyperparameters used in the model are mentioned. First few of hyperparameters are selected based on the type of learning which is regression in this case and then trial error is used to find the optimum values for the rest of the hyperparameters. Below is list of hyperparameters used to train the model across the three traffic categories:

- Number of LSTM hidden layers
- LSTM hidden states
- Number of epochs
- Number of batch
- Optimization algorithm
- Loss function

During training the model performance with a given combination of hyperparameters is evaluated using a selected accuracy evaluation metrics. Model accuracy evaluation is performed using two of the most popular techniques to evaluate regression: The Mean Absolute Percentage Error(MAPE) and Root Mean Squared Error(RMSE). The MAPE shows us how much inaccuracy we should expect from the forecast on average and it is capable of smoothing out the outlying data issues mentioned. The lower the MAPE the better the model.

$$MAPE = \frac{1}{N}\sum_{N=1}^{N}\frac{A_t - F_t}{A_t} \tag{4.18}$$

Where:

MAPE: Mean Absolute Percentage Error

N: number of times the summation iteration happens

A_t: Actual Value

F_t: Forecasted Value

The RMSE is a measure of squared residuals. Mean squared error is perhaps the most popular metric used for regression problems. It essentially finds the average of the squared difference between the target value and the value predicted by the regression model.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(A_t - F_t)^2} \tag{4.19}$$

Where:

RMSE: Root Mean Squared Error

A_t : ground-truth value

F_t: predicted value from the regression model

N: number of dataset

The model is implemented in python and the findings are discussed in the next chapter. By using the model, a multistep prediction is performed for a different test length and observation is discussed with both numeric graphical visualizations.

Chapter 5

Results and Discussion

In this section, the time series analysis is performed for the peak hour cellular network voice and data traffic, in addition, the model and tools used for the experiment are also well defined. The model training is approached for each traffic type separately i.e. CS, PS downlink and PS uplink. Few of the initial hyperparameter selection and setting is done based on the learning type and dataset size, and rest are selected arbitrarily. The test prediction is performed after model training is done by tuning the hyperparameters through trial and error, and the results are discussed according to the nature of the mobile traffic. The model is trained and tested per each traffic set used as input to train and test the model

This thesis developed a model that is used to predict multistep ahead busy hour data and voice traffic of a mobile network based on history data. Model training is performed using 90% of the dataset and a multistep prediction for one, two and three months ahead are produced against the test dataset and analyzed for both traffic categories.

5.1 Dataset quality assessment

The dataset for both data(PS) and voice(CS) traffic contains 166(14.7%) missing value points and few outliers. The outliers are imputed using mean imputation since they will impact the learning process and model performance. The missing values happened because of service interruption and system related problems. Based on the analysis done using IBM SPSS, the missing points happened at random and distributed throughout the dataset which which ignoring it when training the model impacted model accuracy and learning. The amount of missing data is significantly large that dropping it would impact dataset size and model learning, as a result, it is decided to be imputed. The missing points in the traffic data are imputed using the R statistical software.

5.2 Result and Discussion for Downlink data traffic

As can be seen in the graphic visualization, the peak hour downlink data traffic consists of few outliers which are averaged out for purpose of improved learning and model accuracy. Care is taken when cleaning the outliers not to impact the model performance on new dataset that it is not trained on. Since the model predicts based on a look-back samples, both the training and testing prediction starts after a look-back or a sample length. As a result, the prediction curves are found to be discontinuous for a single look-back length for all traffic categories at the start of the prediction.

The model fitting is performed on the training and validation set and the best performing combination of model hyperparameters is selected.

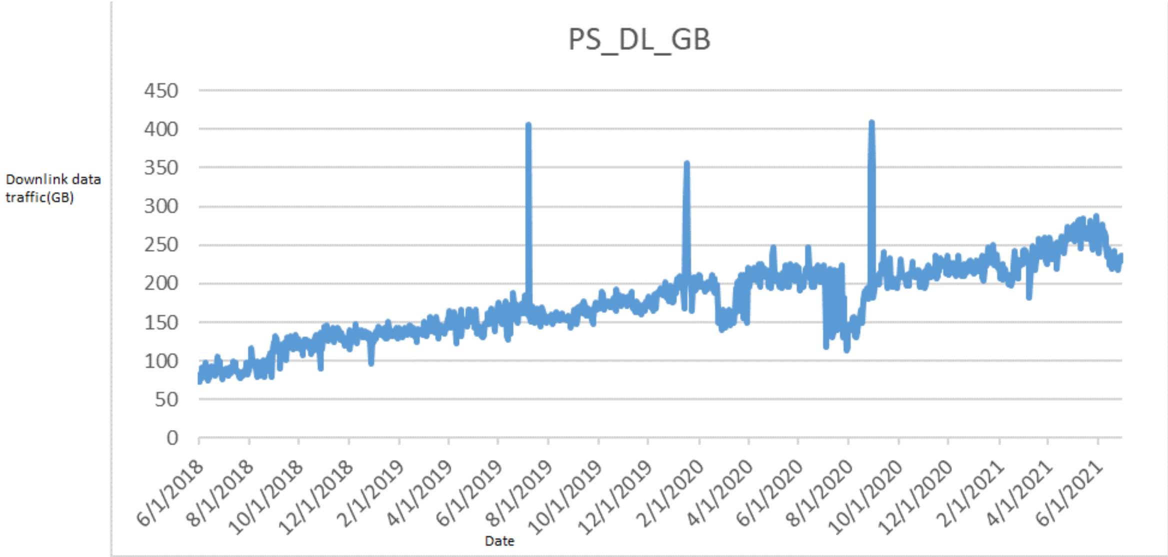

FIGURE 5.1: Downlink data traffic

No.	Hyperparameters	value
1	Number of hidden layers	2
2	LSTM Hidden states	128
3	Number of Epochs	100
4	Optimization Algorithm	Adam(Default)
5	Loss Function	MAE

TABLE 5.1: Model Hyperparameters for downlink data traffic model

The forecasting accuracy of the model for three months ahead is of MAPE = 0.03 and RMSE = 11.2 and the forecast against target data visualization is depicted in Figure 5.3

The model accuracy is also checked with one and two months ahead prediction against a test data. As can be seen for the downlink data traffic the model accuracy for shorter time prediction is better. The RMSE is 8.28 and 10.19 respectively for one and two months ahead prediction. The MAPE for the one and two months ahead prediction is 0.02 and 0.03 respectively.

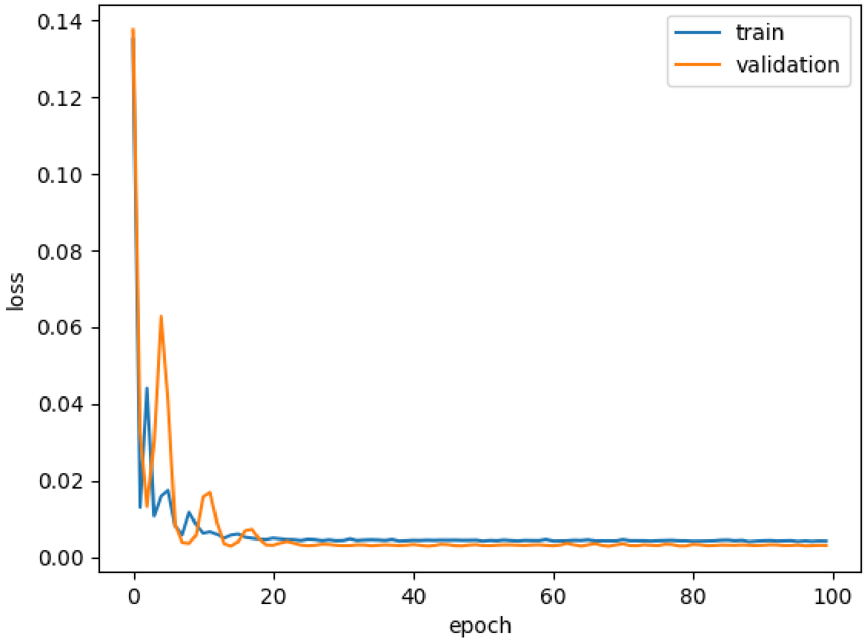

FIGURE 5.2: Training and validation loss for the selected combination of downlink PS traffic model hyperparameters

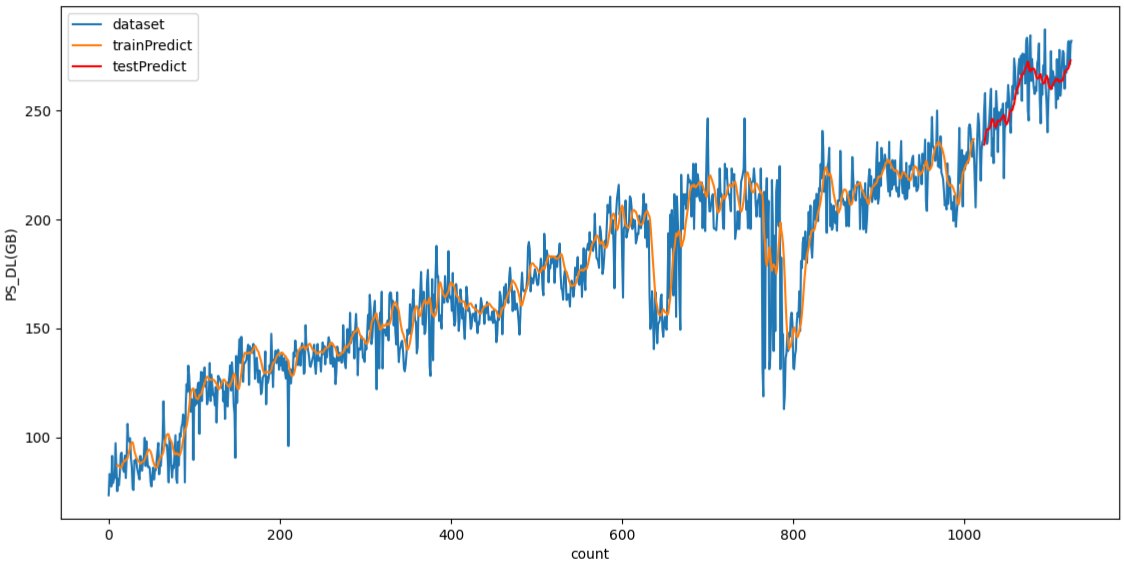

FIGURE 5.3: Three months ahead downlink traffic prediction on test data

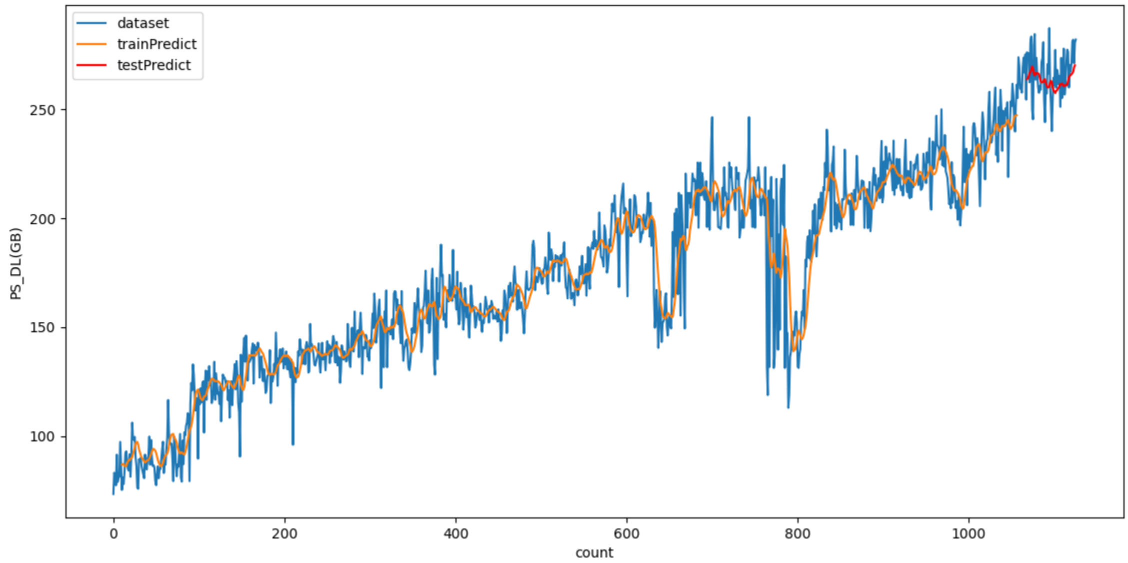

FIGURE 5.4: Two months ahead downlink traffic prediction on test data

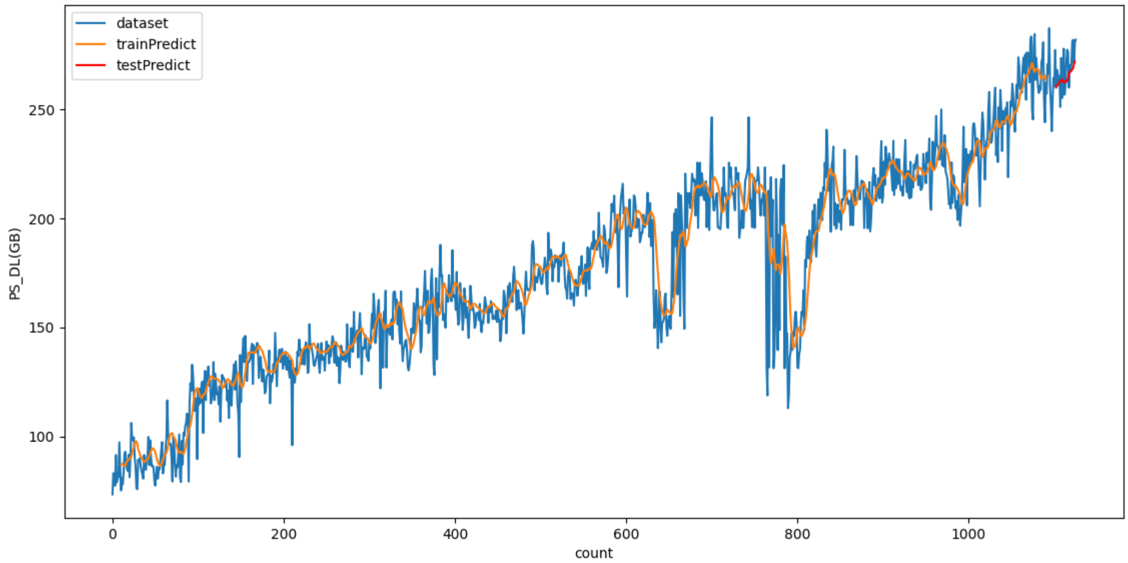

FIGURE 5.5: One month ahead downlink traffic prediction on test data

No of months	Predicted(GB)	Target(GB)
Last one month	261.5278	268.2712
Last two months	262.9233	265.8538
Last Three months	257.9205	259.6738

TABLE 5.2: Comparison of mean of predicted value against target data for the dowlink PS traffic

The mean prediction and target value for the three multi-step prediction shows that the model can actual forecast with a very good accuracy. The mean of predicted value is a little bit smaller than the target the because still there are spikes/outliers not completely removed not to impact the model performance on new datsets. The reason we didn't completely remove the outliers is because our observation shows that volatility is inherent to cellular network traffic the model has to be trained on so that it will not over-fit on new dataset it is not exposed to before.

5.3 Result and Discussion for Uplink data traffic

Due to the asymmetric characteristics of data traffic, the uplink traffic has to be handled separately from the downlink. Just like the downlink data traffic there are outliers which are imputed for the model training. After imputing the missing set the uplink data traffic is visualized as in Figure 5.6.

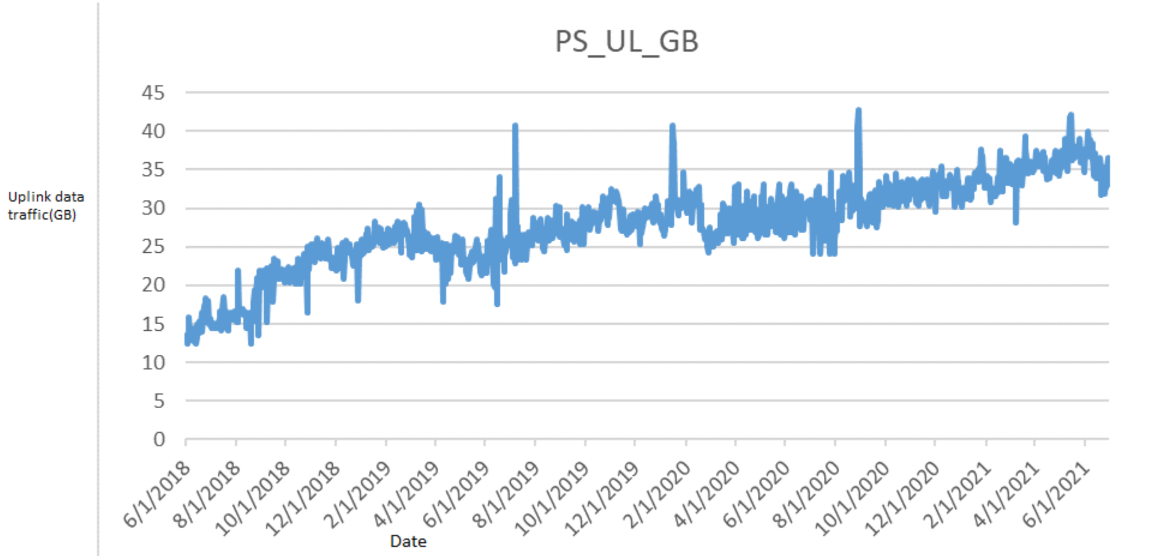

FIGURE 5.6: Uplink data traffic

The model fitting is performed on the training and validation set and the best performing combination of model hyperparameters is selected.

No.	Hyperparameters	value
1	Number of hidden layers	5
2	LSTM Hidden states	128
3	Number of Epochs	100
4	Optimization Algorithm	Adam(Default)
5	Loss Function	MAE

TABLE 5.3: Model Hyperparameters for uplink data traffic

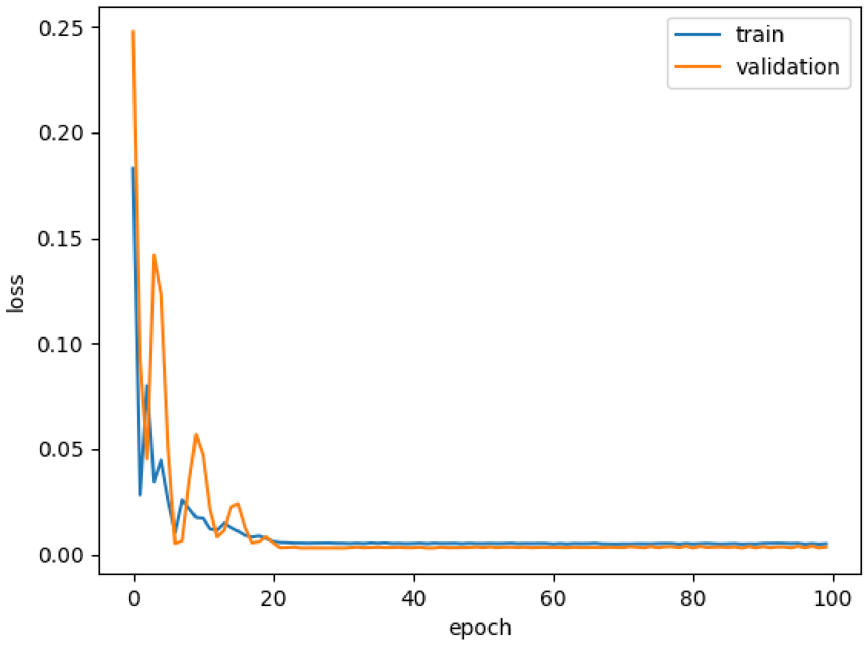

FIGURE 5.7: Training and validation loss for the selected combination of uplink PS traffic model hyperparameters

The forecasting accuracy of the model for three months ahead prediction is of MAPE = 0.04 and RMSE = 1.94 and the forecast against target data is depicted as in Figure 5.8:

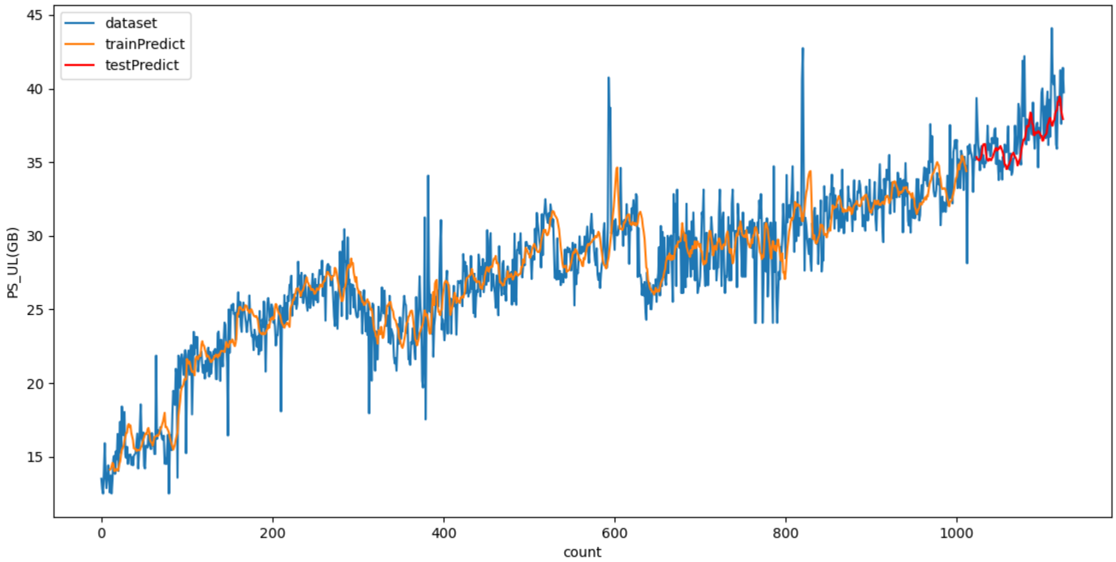

Figure 5.8: Three months ahead uplink traffic prediction on test data

Looking at the original and predicted values of the forecast we can say the model is able to follow the trend with a very good accuracy for both data traffic categories. The model is not perfect in following the extreme values.

A one and two months ahead prediction for this data traffic category is performed. Both the MAPE and RMSE haven't improved as expected when the prediction period is reduced which is found out to be the dataset quality that impacted predictability. The RMSE is 2.69 and 2.3 respectively for one and two months ahead prediction. The MAPE for the one and two months ahead prediction is 0.06 and 0.05 respectively.

No of months	Predicted(GB)	Target(GB)
Last one month	37.2832	39.0057
Last two months	36.6913	38.2385
Last Three months	35.3868	36.9697

Table 5.4: Comparison of mean of predicted value against target data for the uplink data traffic

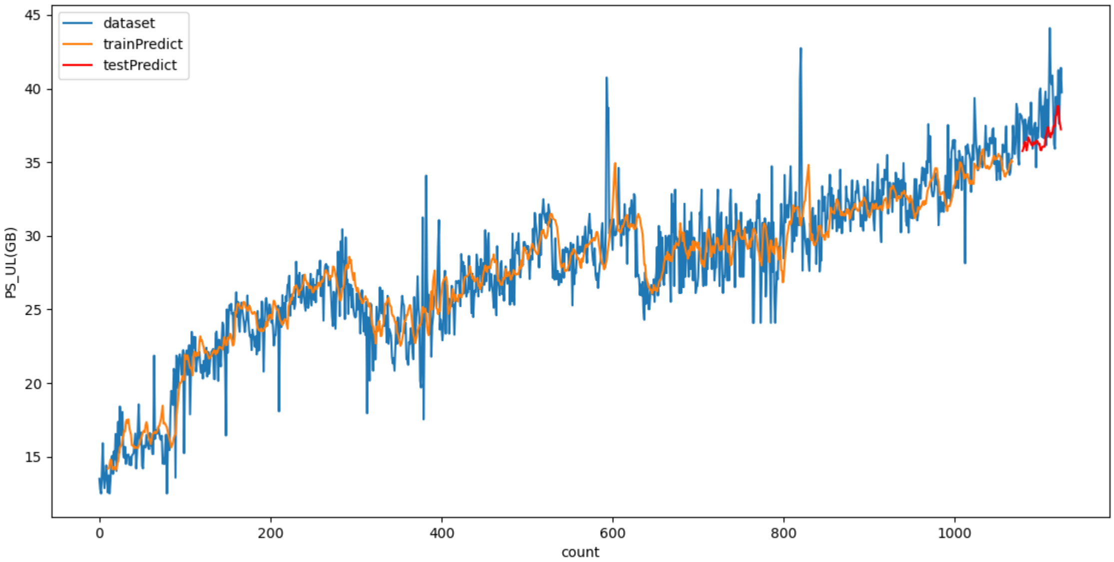

FIGURE 5.9: Two months ahead uplink traffic prediction on test data

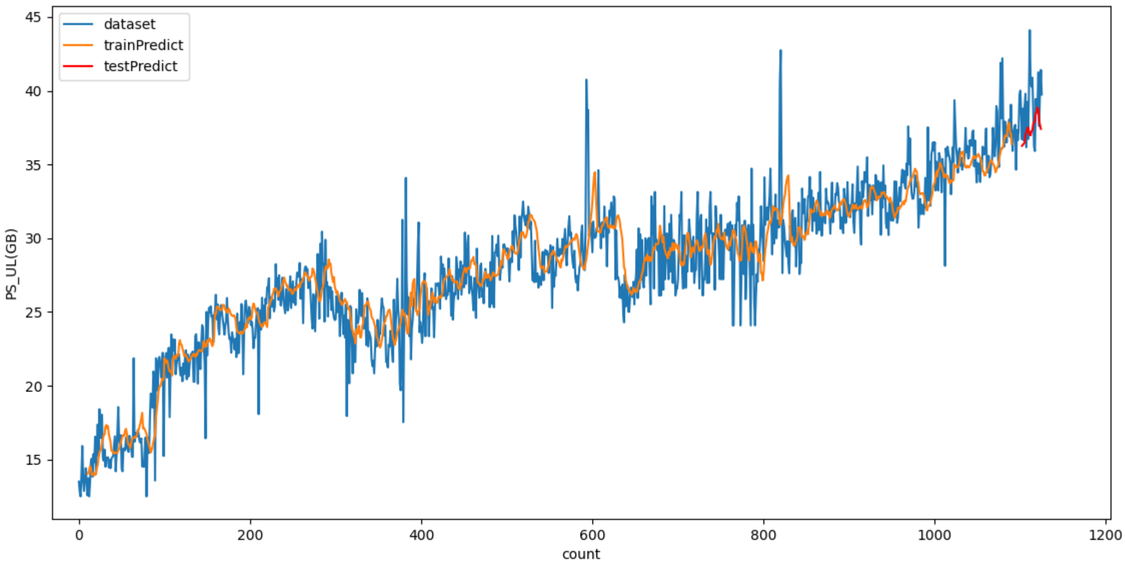

FIGURE 5.10: One month ahead uplink traffic prediction on test data

In the uplink traffic as well the model performed good in multistep forecasting. As mention in the discussion for down link model, the reason mean of prediction is less than mean of the target because of the outliers.

5.4 Result and Discussion for voice traffic

After imputing the missing values figure 5.11 shows voice traffic carried over the network starting June 1, 2018 to June 30, 2021.

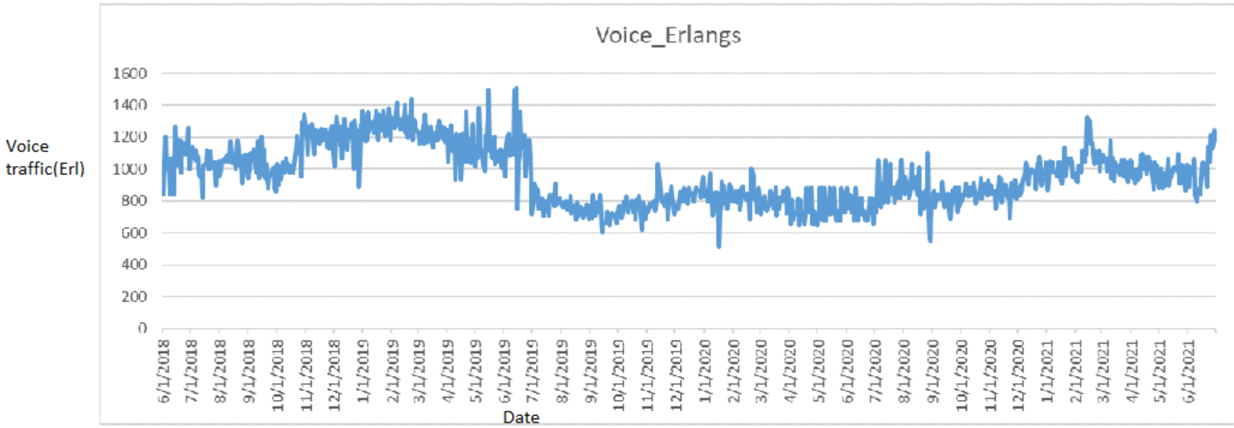

FIGURE 5.11: Voice traffic

The model fitting is performed on the training and validation set and the best performing combination of model hyperparameters is selected.

No.	Hyperparameters	value
1	Number of hidden layers	4
2	LSTM Hidden states	64
3	Number of Epochs	64
4	Optimization Algorithm	Adam(Default)
5	Loss Function	MAE

TABLE 5.5: Model Hyperparameters for voice traffic

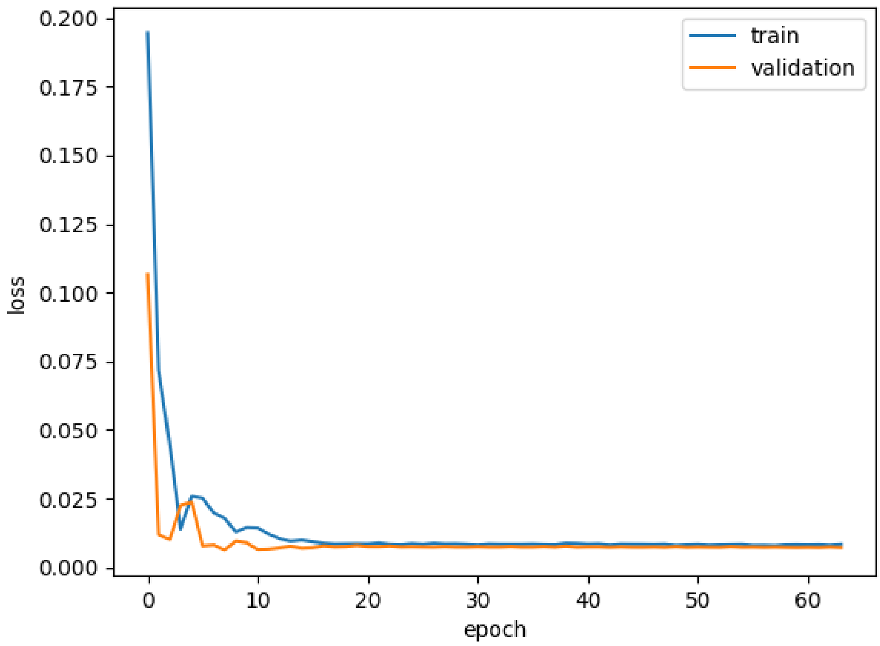

FIGURE 5.12: Training and validation loss for the selected combination of voice traffic model hyperparameters

The forecasting accuracy of the model for three months ahead prediction is of MAPE = 0.07 and RMSE = 80.52 and the forecast against target data visualization is depicted in figure 5.13. The larger RMSE is attributed to the unit of measurement and the amount of traffic carried over the network and the voice data quality. During training, further cleaning of the dataset has resulted in an improved RMSE but this will make model less robust to new dataset. Looking at the original and the predicted values of the forecast we can say the voice traffic forecasting model is able to follow the trend with a very good accuracy. The model is not perfect in following the extreme values.

For the voice traffic as well a one and two months ahead prediction for this traffic category is performed. The model showed a better accuracy for shorter prediction period. The RMSE is 33.4 and 52.3 respectively for one and two months ahead prediction. The MAPE for the one and two months ahead prediction is 0.03% and 0.04% respectively.

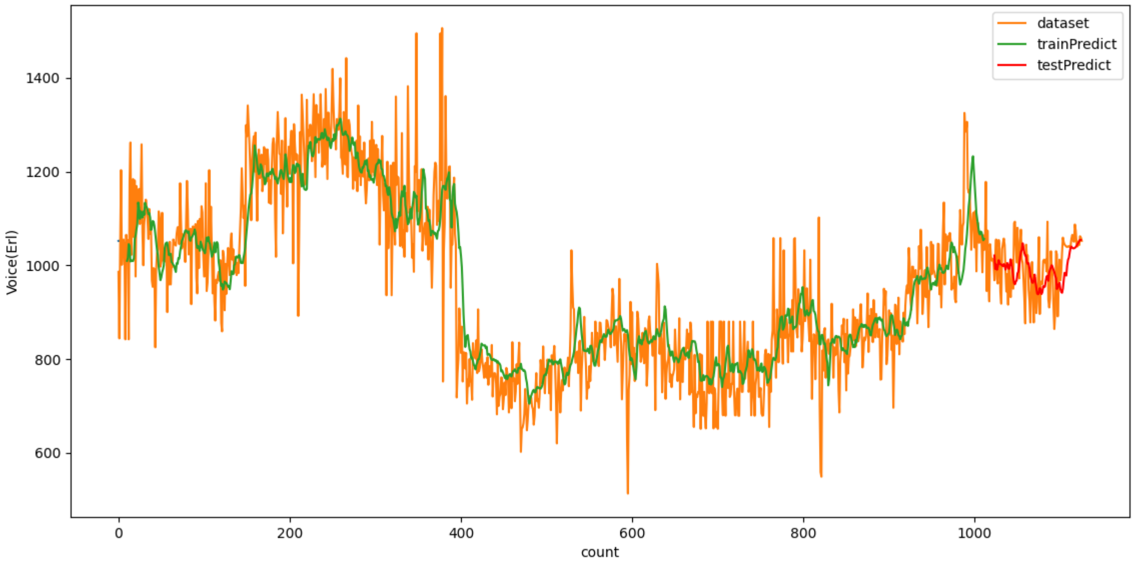

FIGURE 5.13: Three months ahead voice traffic prediction on test data

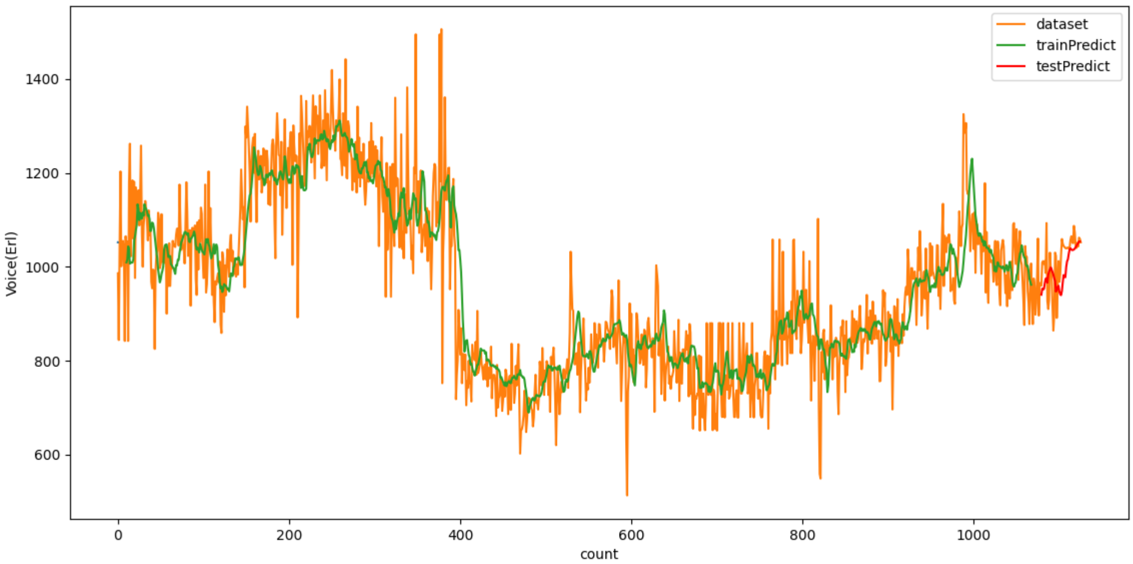

FIGURE 5.14: Two months ahead voice traffic prediction on test data

No of months	Predicted(Erl)	Target(Erl)
Last one month	1027.9087	1049.9565
Last two months	998.61365	1011.7173
Last Three months	972.0947	997.4803

TABLE 5.6: Comparison of mean of predicted value against target data for voice traffic

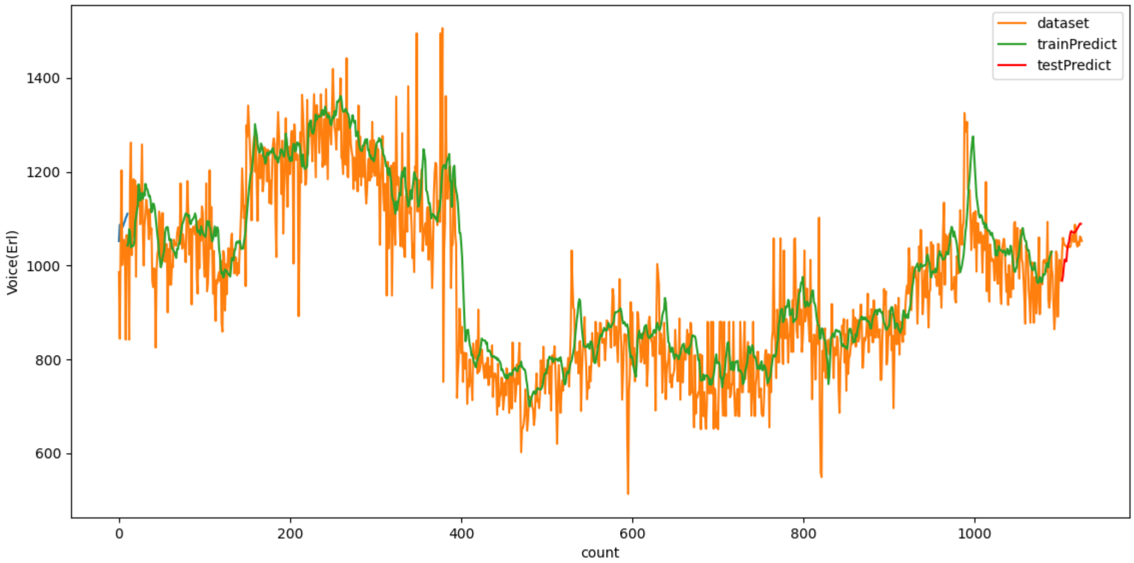

FIGURE 5.15: One month ahead voice traffic prediction on test data

On the voice traffic as well the model performed good in multistep forecasting. As mention in the previous model discussion part, the reason mean of prediction is less than mean of the target because of the outliers.

5.5 Model Validation

Model validation is done by exploring related works which compares the proposed model to other techniques. By using selected performance metrics, the different related works developed time series forecasting model using different platforms to compare LSTM with and validate. In the related works LSTM is compared with traditional machine learning based, CNN based and linear statistical based time series forecasting models. These related works and LSTM performance improvement is summarized in Table 5.7.

In the table, an improved performance can be seen when using LSTM over the selected platforms for baseline comparison. The LSTM based model outperformed the commonly used time series forecasting models based on ARIMA, SARIMA, traditional machine learning and GRU as well, as the training dataset grows. Under the performance improvement column the negative sign shows improved error performance while the positive sign shows improved accuracy performance.

Authors, Publisher	Methodology	LSTM Performance improvement(minimum)
A. Dwivedi, A. Attry, D. Parekh and K. Singla, 2021 ICCCIS, from IEEE Xplore[12]	SARIMA, CNN	MSE = -0.000069
C. Yin et al, IEEE Access[13], 2021	NNAR, SARIMA	R = -0.075, RMSE = -0.469, MAE = +0.093, MAPE = -5.704
O. Kelany, S. Aly and M. A. Ismail, 2020 Technology (IIT), from IEEE Xplore[14]	Logistic Regression and Random Forest	RMSE = -0.632, MAE = -0.459
S. Yang, X. Yu and Y. Zhou, IWECAI, from IEEE Xplore[15]	GRU	F1 = +9.06%, AUC = +5.95%
M. S. Hossain and H. Mahmood, 2020 IEEE Power and Energy Conference[16]	GRNN, ELM	Single-step: MAE = +17.24, RMSE = -23.33, MAPE = -1.53 and multi-step: MAE = -6.2, RMSE = -4.64, MAPE = -0.54
S. Siami-Namini, N. Tavakoli and A. Siami Namin, 2018 17th IEEE International Conference[17]	ARIMA	RMSE = -5.063

Table 5.7: Survey of comparison of LSTM performance to other models

Even if it is not appropriate to compare different models on unrelated datasets, the performance of the forecasting models developed in this research has proved LSTM based models to perform better compared to the linear and traditional machine learning techniques. For instance, considering MAPE which unlike RMSE is capable of smoothing out outliers inherent to the dataset used to train and test, the performance of voice traffic forecasting model 0.07(7%) is a better figure compared to the baseline models in the surveyed literatures. This showed that the performance of LSTM based models developed with the appropriate dataset preprocessing and model fitting is outstanding.

Chapter 6

Conclusion and Recommendation

6.1 Conclusions

In this work, the voice and data cellular network traffic forecasting is investigated taking a dataset from a live UMTS mobile network. The thesis underlines how traffic forecasting is crucial to strategic QoS management of cellular network through planning and optimization and works towards finding a model that is capable of predicting the total voice and data traffic demand. The cellular network capacity to carry a given traffic volume and mix can be checked against the forecasted traffic for tuning and dimensioning of radio resources. As a result, Knowledge of a forecasted cellular network traffic will have a crucial impact on activities performed to maintain Quality of Service provided.

A traffic forecasting model was developed using an LSTM deep learning technique due to its unique capability to capture long term temporal correlation. The prediction accuracy of the forecasting models has been evaluated using MAPE and RMSE for both traffic categories. Model performance analysis shows a higher accumulated error(RMSE) in the voice traffic due to poor quality in the dataset.

6.2 Recommendations

For purpose of strategic cellular network QoS management and planning a business model, it is recommended to investigate the traffic demand for all cellular technologies available. This will help understand evolution of user behavior in terms of volume and mix of service usage. The user behavior will help MNOs to have a business model that can prioritize a cellular network technology capable of satisfying the QoS requirement with an improved ROI. In order to further improve the overall voice(CS) traffic prediction accuracy, we also recommend to additionally consider the traffic carried over the lower technology(2G) which will improve the dataset quality.

Bibliography

[1] Ericsson, Ericsson Mobility Report, Jun. 16, 2021. Accessed on: Jul. 23, 2021. [Online]. Available: https://www.ericsson.com/en/press-releases/2021/6/ericsson-mobility-report-more-than-half-a-billion-5g-subscriptions-by-the-end-of-2021

[2] ITU, IMT traffic estimates for the years 2020 to 2030, Report ITU-R M.2370-0(07/2015), 2015, pp. 3-7

[3] Soldani, David, "QoS Management in UMTS Terrestrial Radio Access FDD Networks," Helsinki University of Technology Communication Laboratory, Technical Report T53, 2005, 234 pp.

[4] Ethio Telecom WND Regional Zonal Strategy Sharing, June 2020, 28 pp.

[5] Dalgkitsis, Anestis, Louta, Malamati, Karetsos, George, "Traffic forecasting in cellular networks using the LSTM RNN," - PCI '18: Proceedings of the 22nd Pan-Hellenic Conference on Informatics, 2018/11/29, pp. 28-33

[6] I. Z. Kovacs, P. Mogensen, B. Christensen and R. Jarvela, Mobile Broadband Traffic Forecast Modeling for Network Evolution Studies, 2011 IEEE Vehicular Technology Conference (VTC Fall), San Francisco, CA, USA, 2011, pp. 1-5

[7] Dereje Gemechu, UMTS Traffic Model Using Artificial Neural Network: The case of Addis Ababa, Ethiopia, A Master's Thesis, AAU Institutional Repository, 2017

[8] Yur, Yasin, Akay, M. Abut, Fatih Short Term Voice Traffic Forecast in 3G/UMTS Networks Using Machine Learning and Statistical Methods. International Journal of Advances in Electronics and Computer Science vol. 3, Barcelona, Spain, 2016, pp. 108-111

[9] E. O. Oladeji, E. N. Onwuka and M. A. Aibinu, Determination of voice traffic busy hour and traffic forecasting in Global System of Mobile Communication (GSM) in Nigeria, 2013 IEEE 11th Malaysia International Conference on Communications (MICC), Kuala Lumpur, Malaysia, 2013, pp. 184-189

[10] S. Medhn, B. Seifu, A. Salem and D. Hailemariam, Mobile data traffic forecasting in UMTS networks based on SARIMA model: The case of Addis Ababa, Ethiopia, 2017 IEEE AFRICON, 2017, pp. 285-290.

[11] Amin Azari, Panagiotis Papapetrou, Stojan Denic, Gunnar Peters, "User Traffic Prediction for Proactive Resource Management: Learning-Powered Approaches," 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1-6

[12] A. Dwivedi, A. Attry, D. Parekh and K. Singla, "Analysis and forecasting of Time-Series data using S-ARIMA, CNN and LSTM," 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS),IEEE, 2021, pp. 131-136

[13] C. Yin et al., "Advanced Machine Learning Techniques for Predicting Nha Trang Shorelines," in IEEE Access, vol. 9, pp. 98132-98149, 2021.

[14] O. Kelany, S. Aly and M. A. Ismail, "Deep Learning Model for Financial Time Series Prediction," 2020 14th International Conference on Innovations in Information Technology (IIT), 2020, pp. 120-125.

[15] S. Yang, X. Yu and Y. Zhou, "LSTM and GRU Neural Network Performance Comparison Study: Taking Yelp Review Dataset as an Example," 2020 International Workshop on Electronic Communication and Artificial Intelligence (IWECAI), 2020, pp. 98-101.

[16] M. S. Hossain and H. Mahmood, "Short-Term Load Forecasting Using an LSTM Neural Network," 2020 IEEE Power and Energy Conference at Illinois (PECI), 2020, pp. 1-6.

[17] S. Siami-Namini, N. Tavakoli and A. Siami Namin, "A Comparison of ARIMA and LSTM in Forecasting Time Series," 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 2018, pp. 1394-1401.

[18] Ajay R. Mishra. Recurrent Neural Networks, in Fundamentals of Network Planning and Optimisation 2G/3G/4G: Evolution to 5G. Second Edition, John Wiley & Sons Ltd (1e, 2004), Ericsson, India2018, Ch. 1, 1-27

[19] Harri Holma, Antti Toskala. UMTS Services, WCDMA for UMTS: Radio Access for Third Generation Mobile Communications. Third Edition, John Wiley & Sons, Ltd, 2004, pp. 1-5, 29-34

[20] 3GPP, Accessed on: Sep.3, 2021. [Online]. Available: https://www.3gpp.org/technologies/keywords-acronyms/99-hspa

[21] ETSI, Universal Mobile Telecommunications System (UMTS); QoS Concept and Architecture (3GPP TS 23.107 version 4.1.0 Release 4), V4.1.0, Jun. 2001

[22] ETSI, Digital cellular telecommunications system (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Policy and charging control architecture (3GPP TS 23.203 version 15.3.0 Release 15), V15.3.0, Jul. 2018

[23] R. Madan and P. S. Mangipudi, Predicting Computer Network Traffic: A Time Series Forecasting Approach Using DWT, ARIMA and RNN, 2018 Eleventh International Conference on Contemporary Computing (IC3), 2018, pp. 1-5

[24] Rob J Hyndman and George, Athanasopoulos. Forecasting data and methods, in Forecasting: Principles and Practice. Third edition, Monash University, Australia, 2021

[25] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, Troncoso. Deep Learning for Time Series Forecasting: A Survey. Feb., 9, 2021 pp. 3-21

[26] Francesca Lazzeri, Time Series Data Preparation, in Machine Learning for Time Series Forecasting with Python. John Wiley & Sons, Inc., Indianapolis, Indiana, 2021, Ch. 9, pp. 61–89.

[27] Charu C. Aggarwal. Recurrent Neural Networks, in Neural Networks and Deep Learning. Yorktown Heights, NY, USA 2018, Ch. 7, 283-309

[28] François Chollet. Deep Learning with Python. first edition, Shelter Island, NY. Manning Publications Co., 2018

[29] Emmert-Streib F, Yang Z, Feng H, Tripathi S, Dehmer M. An Introductory Review of Deep Learning for Prediction Models with Big Data. Front Artif Intell, Feb. 2020

[30] Pedro Lara-Benitez, Manuel Carranza-Garcia, Jose C. Riquelme, An Experimental Review on Deep Learning Architectures for Time Series Forecasting. International Journal of Neural Systems, Nov. 2020

[31] R. Dey and F. M. Salem. Gate-variants of Gated Recurrent Unit (GRU) neural networks. 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), 2017, pp. 1597-1600.