Mobile Network Congestion Prediction Using Machine Learning: The Case of Ethio Telecom

A Thesis Presented

by

Betelehem Alemayehu Hailu

to

The Faculty of Informatics

of

St. Mary's University

In Partial Fulfillment of the Requirements
for the Degree of Master of Science

in

Computer Science

January, 2022

ACCEPTANCE

Mobile Network Congestion Prediction Using Machine Learning: The Case of Ethio Telecom

By

Betelehem Alemayehu

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

Dr. Alemebante Mulu
Internal Examiner

Dr. Sileshi Demesie
External Examiner

Dr. Alemebante Mulu
Dean, Faculty of Informatics

January 2022

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been duly acknowledged.

Betelehem Alemayehu Hailu

Full Name of Student

____

Signature

Addis Ababa

Ethiopia

This thesis has been submitted for examination with my approval as advisor.

Dr.Asrat Mulatu

Full Name of Advisor

____

Signature

Addis Ababa

Ethiopia

January 2022

ACKNOWLEGMENTS

First and foremost, from bottom of my heart thanks to God who endowed me with the strength to tackle problems with support and being with me throughout my life? I would also like to express my deepest gratitude to my advisor Dr.Asrat Mulatu in which the thesis work would not be possible without his motivation, enthusiasm, continuous as well as constructive supervision, and encouragements during the entire process of working this research.

I want to acknowledge Ato Anteneh Amsalu and Ato Alexander Behailu from Ethio Telecom for their interest in helping to provide information and their sincerely. I wish to thank my whole family members specially my mother and my father for their encouragement and constant moral support. Finally, I would like to thank all of you who support me to complete this thesis work as well as my study

TABLE OF CONTENTS

LIST OF ACRONYMS AND ABBREVIATIONS

ET	Ethio Telecom
2G	Second Generation
3G	Third Generation
4G	Fourth Generation
AI	Artificial intelligent
QOS	Quality of Service
LV	Last Value Predictor
LTE	Long-Term Evolution
GPS	Global Position System
GSM	Global System for Mobile
VAS	Value Added Service
EDA	Exploratory data analysis
BTS	Base Transceiver Station
ANN	Artificial Neural Network
SVM	Support Vector Machine
MA	Windowed Moving Average
LSTM	Long Short-Term Memory
DES	Double Exponential Smoothing
KPI	Key Performance Indicator
TRX Av	Transceiver Availability
CSSR	Call Setup Success Ratio

HOSR	Handover success rate
MLP	Multi-Layer Perceptron
AR	Auto Regression
SIM	Subscriber Identity Module
PRF	Premium Rate Fraud
PBX	Private Branch Exchange
TCH	Traffic Control Congestion
NMS	Network Management System
ARMA	Autoregressive Moving Average
ETC	Ethiopian Telecommunication Corporation
ETA	Ethiopian Telecommunication Agency
IBTE	Imperial Board of Telecommunications
SDCCH	Stand-alone Dedicated Control Channel
SACCH	Slow Associated Control Channel
MLP-NN	Multilayer Perception Neural Network
PSTN	Public Switched Telecommunication Network
PEMS	Performance Measurement System

LIST OF FIGURES

LIST OF TABLE

ABSTRACT

A mobile network, also known as a cellular network, is a radio network that is distributed over land areas known as cells, each of which is supplied by at least one fixed position transceiver, also known as a cell site or base station. Congestion, fraud, and delay on international calls are among issues that these networks confront. For practically all telecom service providers across the world, these issues are severe threats and customer churn issues. In the context of Ethiopian telecom network data, this paper seeks to handle mobile network congestion problems using machine learning techniques termed multilayer perceptron neural networks. The network data used in this article was obtained from Ethiopia Telecom's key performance indicator database over a six-month period. For the aim of constructing the machine learning models, a total of 3080 data sets with 15 attributes are employed after removing unnecessary data, formatting the data organization, and clustering the data into three independent data sets for each site.

By conducting performance analysis of Multilayer Perception Neural Network models with different combinations of training algorithms, activation functions, learning rate, and momentum, it was found out that Multiple Layer Perception Neural Network model having 15 hidden layers each having 200 neurons with Adam optimizer training algorithm and Relu activation function delivered the lowest mean absolute error of 0.272 while another Multilayer Perception Neural Network model having 10 hidden layers having 200 neurons in each layer, the same activation function and training algorithm had the mean absolute error of 0.345. The results of this research showed that performance analysis of Multilayer Perception Neural Network models is a crucial process in model implementation of Multilayer Perception Neural Network for mobile network congestion prediction and a multilayer perceptron having 15 layers can give a comparable prediction of the real mobile network congestion situation.

The lack of sufficient data and enough expert knowledge of the performance parameters of the network were of the major challenges faced during the craft of this research paper. Finally, through the results found in this paper we recommend ethio telecom to implement this mobile network congestion prediction techniques and avoid such types of irregularities throughout the network which will improve user experience and reduce customer churn.

Keywords: Relu, Congestion Prediction, MLP_NN, QoS, Machine Learning.

CHAPTER ONE

INTRODUCTION

1.1. Background of the Study

This section aims to provide some background information on predicting mobile network congestion networks using machine learning in the context of Ethiopian telecom. The chapter discusses the problem statement and related work on predicted mobile network congestion networks, as well as the thesis' main objective and methodology.

Ethiopia was the first country in Africa to introduce telecommunications in 1894.It is Africa's oldest public telecommunications company [1].Then the widespread open wire line system connecting the capital with all of the country's key organizational locations was set out in those years, the technological plan contributed to the inclusion of Ethiopian society [2].

"In 1941, Ethiopia reorganized the Telephone, Telegraph, and Postal services at the end of the war against Italy, during which the telecommunication network was destroyed." By proclamation No. 131/52 in 1952, the Imperial Board of Telecommunications (IBTE) was founded. The Board was in charge of providing and expanding telecommunications services in Ethiopia and had complete financial and management autonomy. By proclamation 49/1996, the government formed a distinct regular organization, the Ethiopian Telecommunication Agency (ETA), and the Council of Ministers established the Ethiopian Telecommunication Corporation the same year by regulation 10/1996. (ETC). Under the decree no 197/2010, ETC has been replaced by 'Ethio telecom' since November 2010. ET's major responsibility, under the supervision of the ETA, is to maintain and grow the network [3].

The Ethiopian government has decided to focus on the development of telecommunication services, considering them a key lever in the development of Ethiopia, as a continuation of the 2005/06-2009/10 five-year plan, and after concentrating its efforts on education, health, and agriculture, the Ethiopian government has decided to focus on the development of telecommunication services, considering them a key lever in the development of Ethiopia. Ethio telecom is reborn, on November 29, 2010, from this ambition of supporting Ethiopia's steady growth [2].

Telecommunications services play an important part in a country's political, economic, and social growth. It is also crucial in society's day-to-day activities. People may now connect immediately across long distances, share information, and do business thanks to extraordinary advancements in telecommunication technology. Furthermore, the availability and dependability of telecommunication services have an impact on the success of commercial and social interactions that have become reliant on the service provider.

Predicting network congestion is critical for increasing network performance and security. Different machine learning algorithms for traffic analysis are discussed in this study. Network traffic has increased, as has development. For the next generation cellular network, predictive study of mobile network congestion is becoming increasingly important. Knowing the user's demands ahead of time allows the system to allocate resources efficiently. Accurate data traffic prediction in telecom networks is a difficult task for better network management. It enhances resource allocation and power management in a dynamic manner. Long Short-Term Memory and neural networks are used in this thesis (LSTM).

Ethio telecom's major goal is to supply customers with high-quality service (QoS). QoS can be measured from both a network and a user standpoint (through a network management system) (Via drive test system). Ethio telecom now uses a classic simple statistical method to analyze data retrieved from the Mobile Network Management System database, which is utilized to assess network performance, service quality, and make decisions.

When providing telecom services, the telecommunications industry generates and retains a massive amount of data. These data comprise call detail data, which describes the calls that pass through telecommunication networks, network data, which defines the health of the network's hardware and software components, and customer data, which pertains to telecommunication consumers. This massive amount of data must be accurately handled for a variety of purposes, including fraud detection, network performance analysis, call drop analysis, congestion analysis, customer churn prediction, report generation for higher officials, network planning and optimization, and decision support [4].

Ethio Telecom (ET) is an Ethiopian telecommunications firm that offers voice, Internet, and data services to the general public through fixed lines, mobile networks, and satellite communication. ET has a 44.5 million-strong mobile customer base. [5].Mobile communication service generates the majority of ET's revenue, making it one of the largest and most important services [2]. This mobile network/cellular network currently serves customers using 2G (second generation), 3G (third generation), and 4GLong Term Evolution (LTE) network equipment [6].

Network communication components such as network switches, connectors, and transmission lines or channels produce mobile network congestion when the network is unable to satisfy the objectives of network performance for connection requests. Marketing techniques and pricing schemes also play a role. These variables cause the channel to become overburdened with packets (calls), resulting in a high number of call losses owing to block and drop calls, as well as a reduction in QoS [7] [8]. In the GSM network, congestion is a problem. It is a situation that occurs when the volume of calls terminating from a network exceeds the network's capacity at a given time. Call signals are queued on the transmission channel as a result of this [9]. Ethio Telecom is a 100% state-owned service provider in Ethiopia that offers a variety of telecom services, including the Public Switched Telephone Network (PSTN), the Global System for Mobile (GSM) cellular mobile, and data communication services. [1].

1.2 Statement of the Problem

Ethio Telecom offers a wide range of telecom services across the country. Cellular mobile service is the most common source of consumer complaints about service quality among these services. Customers have been complaining about the low quality of cellular mobile service, despite the Company's tremendous achievement in establishing contemporary information and communications technology network infrastructure. The problem of network congestion is getting unsolvable. When there is congestion, there is a lengthier latency, a lot of jitter, and a lot of packet losses. As a result, network QOS deteriorates, and end-users will not be satisfied [9] [10] [11] .

Residents in Ethiopia, particularly in Addis Ababa, are relocating to new areas that are not part of the ET plan. There is a rise in the number of customers in that area. ET devises a strategy for establishing a cellular network based on the number of residents. Congestion can be caused by a power loss, a fiber problem, a configuration problem, or an equipment fault in a cell. As a result, the network is congested, the equipment's capacity is limited, and customer usage is unsuitable. Using historical observations to estimate future network traffic is a critical step in understanding and planning the capacity of Ethiopia's telecom infrastructure to tackle these issues. To anticipate mobile network congestion networks based on machine learning, most studies employ statistical prediction models. A good data traffic prediction model should be able to capture important traffic characteristics including long-term dependence and self-similarity on a larger time scale, as well as multi-factuality on a smaller time scale. However, these statistical network models do not take into account all of the properties of machine-learning-based mobile network congestion networks, such as long-term reliance and self-similarity.

Dereje tries to uncover secret information from the Ethio telecom GSM mobile network in order to calculate the call establishment success rate. Yard uses machine learning to try to forecast the cause of dropped mobile calls. Using data mining techniques, Lulu seeks to analyze the quality of service of mobile telecoms networks. Gebremeskel and Jember attempted to assist in the identification of mobile fraud on Ethio telecom mobile services. As far as we know, no research has attempted to anticipate mobile network congestion using Ethio telecom network data. The majority of local researches focus on customer classification, telecom fraud detection, and predicting the reason of call drops and fault incidence during the examination of connected operations. As a result, there is a need to investigate the data created by telecommunication mobile networks using dynamic prediction models to predict mobile network congestion. The findings of this study aid the network provider in improving the mobile network's service quality.

1.3 Research Questions

1. Which attributes are best to identify and classify the model used to Predict Mobile Network Congestions?

2. Which machine learning algorithms can be more suitable for the purpose of predicting Mobile Network Congestions?

3. To what extent a multilayer neural network is able to predict in the given dataset?

4. How many perceptron and layers can provide optimal error minimization while avoiding over fitting in this model?

1.4 Objective of the thesis

1.4.1 General Objective

The main objective of this study is to predict mobile network congestion using machine learning algorithms in the Ethio Telecom cellular network**.**

1.4.2 Specific Objectives

In ordered to achieve the main objective of this thesis the following specific objectives considered are

- To assess and predict mobile network congestion using machine learning network traffic. Ethio telecom data is collected, pre-processed, analyzed, and transmitted.
- Determine the key performance indicators (KPIs) for assessing the quality of mobile network congestion.
- Create a machine learning model for predicting mobile network congestion.
- Pick a machine learning model that works well with the network.
- Evaluation: Finally, utilizing the model and experiment data, conduct an evaluation.

1.5. Significance of the Study

Cellular network operators throughout the world are grappling with the issues of enhancing QoS while expanding capacity and launching new services. As a result of insufficient provision of needed resources or underutilization of available resources, many networks have become congested, resulting in QoS degradation. The massive and complex data must be managed so that useful data may be extracted and exposed. The research proposed an ML model to assess the complicated data generated by mobile network systems in this regard. The findings of this study aid the network provider in improving the mobile network's service quality. In general, this research is significant because it has the potential to reduce income losses caused by mobile congestion.

1.6 Scope and Limitation of the study

This thesis compiles data from the Ethiopian telecom network and 3G voice data over a four-month period. Before the revolution in machine learning analytics can truly benefit the large amount of data, many difficulties must be overcome. Customer behavior modeling and considering data as multivariate are two elements that can be explored to improve the accuracy of mobile network congestion network forecast. Because of the model's complexity and time constraints, only uni variate data, i.e. the total volume of data transmission, was employed. Furthermore, the row data used is restricted to ethio telecom.

1.7. Thesis Organization

Chapter 2 a literature review of related works in the rest of the thesis is organized as follows gives an introduction to the relevant technologies considered in this thesis, as well as further details on related works.

Chapter 3 presents a general overview of the proposed solution and delves further into the assumptions considered in this thesis. Analysis of the data considered is also presented.

Chapter 4 shows the results and experiments performed. The results are then discussed, the factors responsible for the numerical results, and details the effect of using certain features over others

Chapter 5 concludes this thesis and list main contributions

CHAPTER TWO

LITERATURE REVIEW

2.1. Introduction

Introduction this chapter deals with the definition of terms and related works relevant to the thesis problem. It also carries out the ideas and concepts that other scholars have put forward concerning with predict mobile network congestion networks.

2.2. Overview of mobile Network congestion

Network Congestion: is a situation when the network exchanging/carrying more data than it can comfortably handle. For example when the quantity of calls originating or terminated from a particular network exceeds the network's capacity at a given moment." In a region where there is no network, calls cannot be made or received. Calls may be difficult to make or receive if a network exists but has poor connectivity. Poor call connectivity, on the other hand, can be caused by a variety of variables that affect network quality, one of which being congestion. **[17].**

2.2.1. Causes of Network congestion

Network Congestion occurs when a network is not able to adequately handle the traffic flowing through it. While network congestion is usually a temporary state of a network rather than a permanent feature, there are cases where a network is always congested signifying a larger issue is at hand.

In this section, we will discuss five (5) common causes of network congestion including:

- Over-subscription
- Poor network design/mis-configuration
- Over-utilized devices
- Bandwidth hogs
- Security attack

Over-Subscription

Have you ever had a web browsing experience that was consistently speedier at some times of the day than at others? For instance, you are much more likely to have a better browsing experience at night than during the day. This is due to the fact that the network has more users during the day (peak period) than at night (off-peak period). It's like taking the subway during rush hour instead of when everyone is at work. This is frequently the outcome of Over-Subscription, which occurs when a system (such as a network) is managing more traffic per time than it was meant to handle. It's vital to keep in mind that over-subscription is frequently done on purpose to save money.

Poor Network Design/Mis-Configuration

Poor design or device mis-configuration is a more major cause of network congestion. Consider a broadcast storm, which occurs when a huge amount of broadcast and/or multicast traffic is seen on the network in a short period of time, causing significant performance deterioration. Due to the fact that broadcasts are kept within subnets, the larger the subnet, the more severe the impact of a broadcast storm. As a result, a network built with huge subnets and without sufficient consideration for broadcast storms might cause network congestion.

Layer 2 loops are another example of broadcast storms. Broadcast messages are used to discover unknown MAC addresses in a layer 2 segment. If the network is in a loop, the same broadcast message can be delivered back and forth.

Over-Utilized Devices

Routers, switches, and firewalls are all built to manage a specific amount of network traffic. The Juniper MX5 has a 20Gbps capacity, for example. This is the maximum capacity, as well as a theoretical figure (capacity in the production environment will be slightly lower). As a result, consistently sending 20Gbps of data through that device will cause it to be over-utilized, resulting in excessive CPU use and packet losses, causing network congestion.

Bottlenecks are another issue connected to over-utilized devices that can cause network congestion. As with other hierarchical designs in which numerous devices feed into a higher-level device, it's important to make sure the higher-level device can handle all of the traffic from the lower-level devices. If this isn't the case, the higher-level device may become a bottleneck, producing network congestion. Consider a four-lane motorway merging into a two-lane roadway.

Bandwidth hogs

By mistake or on intent, a bandwidth hog is a device or user that consumes much more data than other devices. Depending on the device/user, the difference between regular data usage and the hog's usage can be little or substantial. An NPM, on the other hand, may detect when a device is using more bandwidth than it should. Some NPMs enable you to monitor bandwidth usage in real time, allowing you to see when a bandwidth hog is using resources.

Security Attack

Even with the 4Mbps connection provided by their ISP, a network of roughly 10 users in another organization for which I consulted had a dismal browsing experience. Following an inquiry, it was determined that one of their servers had been hacked, and that the attacker was utilizing it to host unlawful content, resulting in a massive quantity of traffic being transmitted to and from it. The crowded network was once again "free" for normal user traffic when this server was cleaned up. Viruses, worms, and Denial of Service (DoS) assaults are examples of other security threats that can cause network congestion.

2.3 Network Congestion prediction

Predicting network congestion is a critical subject that has recently piqued the interest of the computer network community [13]. Network traffic prediction is a common issue that is important for network monitoring, network security, avoiding congestion, and enhancing network performance. Researchers employ a variety of strategies to forecast network traffic. We have divided these techniques into four broad groups: linear time series model, nonlinear time series model, hybrid model, and decomposition model. This gives an overview of the four categories in which diverse network traffic prediction techniques are classified.

2.4 Application of Machine Learning in Telecommunication Industry

The telecoms sector produces and keeps massive amounts of data. Call detail data, which describes the calls that cross telecommunication networks, network data, which defines the condition of the network's hardware and software components, and customer data, which describes telecommunication customers, are among these data. Because there is so much data, manual data analysis is challenging, if not impossible. The growth of the telecommunications industry is one of the most important markers of a country's social and economic progress on a global scale. Furthermore, the advancement of the communication sector is critical to the total development of all sectors of social, political, and economic activities. The nature of innovation and dissemination in this industry is quite dynamic. Machine Learning is used in the telecommunications industry for a variety of objectives. The following are a few examples of machine learning applications in the telecommunications industry.

Use of Machine Learning to identify and restart Sleeping Cells

Unfortunately, a cell tower across the radio network may crash our PC or laptop. This can have a significant impact on service, particularly in congested areas or during peak times of day. Machine learning can be used to analyze network performance data, learn from it, and locate sleeping cells, which can then be automatically restarted. It is now a manual procedure, but it is being heavily worked on to automate it in order to fully exploit the potential of Machine Learning.

Using machine learning to identify Potential Churners

All types of consumers are now a typical occurrence for network operators, as competition has increased and new deals are constantly being introduced to the market. Many businesses already have rudimentary pattern matching software in place to assist them identify new consumers. These aren't ideal, though, and they need to be maintained on a regular basis. Machine learning algorithms are being created to continuously learn from new data in order to better understand why subscribers leave and generate a new customer retention strategy.

Using machine learning to improve service application

As we all know, the telecom sector is notorious for developing user profiles in order to offer new services to specific groups of people. However, the number of user profiles that can be detected, maintained, and kept up-to-date has a limit. To help with this, unsupervised machine learning algorithms that are fed data on user behavior across the network can aid in detecting new users who were previously ignored. The information is then utilized to figure out which bundle is best for which consumer.

Using Machine Learning for Detecting Frauds

Another important machine learning use in telecommunications is detecting fraud and ensuring income, both of which have a direct impact on the bottom line. Currently, the majority of network operators employ fraud mitigation software that employs rules-based logic to detect fraudulent activity. However, supervised machine learning methods may now be used to detect such actions. Machine learning may potentially have the capability of detecting early trending disparities in criminal behavior.

2.5 Review of related works

The thesis has conducted scientific literature review to assess the major issues and concepts in the predict and control mobile network congestion based on deep neural networks (LSTM), ANN,AI and Machine learning prediction telecom service and other related concepts are considered to confirm the impact of the research. An assortment of different journals, books, papers and articles are grasped from the internet, library, manuals and e-books. In this section different works which are relating with Machine Learning for telecommunication sector is presented. The paper presented are extracting data from mobile network and different study in telecom sector using LSTM, **ANN,AI** and Machine Learning techniques are selected and existing based on the significance, similar this work.

Sophia &Olatokun [14] investigated congestion control mechanisms and patterns of call distribution in GSM Telecommunication network in Nigeria in MTN network. The major data acquired was automated data from MTN's data logging system, which was augmented with unstructured interviews. Sophia and Olatokun respond to six researcher questions that are stated and tested in accordance with the goals. The data is analyzed in MS Excel, and the results are presented using descriptive statistics. The study found that, in addition to the MTN network's carrying capacity, other factors such as the use of phones for data transfers and multimedia activities contribute to traffic congestion on the network.

Simone Manganite, Michael Schapiro [7] study congestion control for future mobile network. MORC is presented a novel rate-control protocol for mobile networks. MORC uses the PCC protocol design framework to provide a good balance of latency and throughput. They reviewed various deployment scenarios for MORC. MORC is a significant first step, but it is still far from fulfilling the full potential of next-generation mobile networks, according to them. They also go over some of the current and future work in these areas. MORC outperforms traditional TCP congestion control and the recent BBR protocol in lab tests and early field tests, achieving faster file download times, higher resiliency to network changes, better bandwidth utilization, and improved video client quality of experience.

Sneha Kumar Kasera, Ramachandran Ramjee, Sandra R. Thuel [15]Studied Congestion Control Policies for IP-Based CDMA Radio Access Networks. The subject of congestion control in a CDMA wireless access network's IP RAN was investigated, and three control techniques dubbed admission control, diversity control, and router control were used to maximize network capacity while preserving high speech quality. We demonstrate how the three alternative control techniques may let the IP RAN elegantly manage 10-40% congestion overload. First, we propose two novel CDMA call admission control upgrades that take into account a unified view of both IP RAN and air interface resources. Next, we offer diversity control, a novel technique that takes advantage of CDMA networks' soft-handoff feature and removes selected frames from several soft-handoff legs to smoothly degrade voice quality during congestion. Finally, they investigate the effects of router control in the presence of an active queue.

Guiomar Corral et al.developed Congestion Control in ATM Networks using Artificial Intelligence Techniques. They propose a system to reduce short-term congestion in ATM networks. This system use Artificial Intelligence algorithms to forecast future network congestion so that less extreme actions can be taken ahead of time. Following a thorough review of prior studies, a UPC algorithm based on buffer use was adopted. This algorithm has been tweaked to help with congestion control. The major purpose of the new approach is to anticipate future buffer consumption. The findings of a formula provided by AI will be used to make this forecast. Several questions about this work remain unanswered in order to improve AI application benefits. The AI application formula, the methodology utilized to get this prediction formula, the use of various traffic sources, and the threshold policy used are all among the questions.

Pragyan Verma et al., [17]studied Dropping of Call Due to Congestion in Mobile Network. Examine the causes of call drop page and the different states of call drop page in mobile communication (Mobile to Mobile and Mobile to Landline). It offers a variety of documents based on books, the Internet, and research papers to present a summary of call dropping and ways for reducing it. It may be useful to a variety of persons working in the telecom industry, particularly those involved in operations and maintenance, in identifying the many causes of call dropouts and proposing solutions for fixing them. Almost every telecom operator in the globe is experiencing call drop issues, and they are improving their networks at both the hardware and software levels to reduce call drop as much as possible. Because it is closely tied to consumer happiness and network operator revenue. According to the ITU and 3GPP, if a call is made.

A. Ozovehe et al., [18]investigate busy hour traffic congestion Analysis in Mobile Macro Cells. For traffic congestion study, real-time traffic data from an integrated GSM/GPRS network was used. The study focused on ten congested cells, including data from the network management system (NMS) spanning three years. The TCH congestion correlation research during rush hour revealed that congestion is influenced by CSSR and BH traffic. For the cells, the CSSR and BH traffic have average correlation coefficients of 0.9 and 0.6, respectively. Because of the strong association, CSSR and busy traffic knowledge can be utilized to predict TCH congestion, which is important for cellular network optimization and resource management. When choosing input for congestion prediction models, the correlation test is crucial. The ten most congested areas were studied, and it was discovered that slow responses were the most common.

Min Liu et al., [19]enveloped prediction of Congestion Degree for Optical Networks Based on BP Artificial Neural Network. By creating BP-ANN, I proposed and implemented an approach for predicting the degree of congestion in optical networks. The fundamentals are discussed, and simulation results show that their proposed method is capable of predicting network congestion levels. They choose four critical request attribute values as input for BP-ANN learning and training. The volume and quality of example data, as well as the correctness of the network model, affect prediction accuracy. By improving them, we will be able to make more accurate forecasts. BP-ANN. Because the trained BP-ANN has a strong nonlinear conversion mapping ability, it can make the most of huge data in the real-world network environment.

Khan et al., [8]study early detection of congestion using specific location or area and usually call attempts made in the location. In this study Multilayer Perceptron Neural Network is used to detect congestion. The trained network was used to anticipate traffic congestion in a specific location using daily traffic data and a multilayer network model. They investigated applying this on the data set, which resulted in output prediction of Stand-alone Dedicated Control Channel (SDCCH) failure attempts in particular area using Levenberg-Marquardt as a training method in neural networks for output prediction that takes less time in training.

Raheem and Okeene [9] study a neural network approach to GSM Traffic Congestion Prediction. Using twelve months of real traffic data, they present a GSM congestion prediction model based on multilayer Perceptron neural networks (MLP-NNs) with sigmoid activation functions and Levenberg-Marquardt [9] Algorithms (LMA). The trained network model was used to forecast traffic congestion on a particular route. The correlation coefficient between expected traffic congestion volumes and actual traffic congestion volumes is 0.986, according to regression analysis. This result demonstrates the utility of artificial neural networks in traffic congestion prediction and control.

Siddiqui and Choudhary [20] developed Telecom voice traffic prediction for GSM using Feed Forward Neural Network. The study is being carried out to estimate peak-hour voice traffic congestion. They check QoS reports on a regular basis to verify that voice traffic is being used to its full potential. This QoS report includes resource parameters such as the number of traffic channels (TCH), stand-alone dedicated channels (SDCCH), and number of SDCCH seizure attempts, SDCCH success calls, SDCCH block calls, and SDCCH drop calls, total calls, TCH assign, TCH success calls, TCH availability rate, TCH drop, Incoming handoff (HO) success rate, Outgoing handoff (HO) success rate, Half rate (HR), and Mean holding time. Siddiqui and Choudhary's research focuses on utilizing a neural network to forecast voice traffic using the above-mentioned QoS criteria on a daily basis and working with real-world data for quality assessment.

Elisha Didam Markus et al, [21]Studied Predicting Telephone Traffic Congestion Using Multi-Layer Feed forward Neural Networks. It shows how to estimate traffic congestion in a telephone network using an artificial NN model. To represent telephone traffic, the design technique employs a multilayered feed forward NN with back propagation algorithm. All simulations were conducted in the Mat lab. The correlation coefficient between projected and real traffic congestion levels was 87 percent in a regression analysis, demonstrating the utility and effectiveness of Neural Networks in traffic prediction and control. The simulation findings show that after the influential variables that influence congestion are determined, the neural network MLP can forecast network congestion in the short and long run.

J. Alan Bivens et al., [22]developed Network Congestion Abbreviation and Source Problem Prediction Using Neural Network. Focus on using a simple feed forward neural network to predict severe congestion in a network. They use neural networks to predict the source or sources responsible for the congestion, design and apply a simple control method for limiting the rate of the offending sources so that congestion can be avoided. Unlike the usual TCP/IP flow control, the proposed method is applied only to selected nodes and converges to the final rate faster. A learning mechanism can be of great value for a network manager. The generalization power of a neural network particularly is appropriate because of the unpredicted variance of parameters that the network manager encounters. Neural networks are an appropriate Mechanism for decision making in pro-active network management.

Aliyu Ozovehe[23] studied Mobile Soft Switch Traffic Prediction using Polynomial Neural Networks. Based on data from a real GSM/GPRS network, Mobile Soft Switch (MSS) busy hour traffic forecast was performed, and correlation analysis was utilized to determine that busy hour call attempt (BHCA) use and A-interface utilization have a substantial positive linear correlation. The key performance indicators (KPI) of busy hour call attempt (BHCA) utilization and A-interface utilization are employed as inputs into the GMDH prediction model, with BH traffic as the model target. Three statistical performance indices were used to evaluate the model's performance: mean absolute percentage error (MAPE), root mean square percentage error (RMSPE), and goodness of fit (R2) values. Experiments show that the suggested model can reach R2 values of up to 96 percent for both short- and mid-term forecasting.

Quang Hung Do et al., [24]developed Prediction of Data Traffic in Telecom Networks based on Deep Neural Networks. In their study, deep neural networks were used to analyze, assess, and make predictions based on telecommunications activity data released by Viettel Telecom in Vietnam every hour for a year. The ability of the GRU and LSTM models to analyze and predict data traffic was compared and contrasted. Several metrics, such as RMSE, MAPE, MAE, and R, were used to assess the performance of the developed models. According to the findings, deep neural networks could be useful for processing time series data in telecom networks. The GRU and LSTM models outperformed the ANFIS, ANN, and GMDH models on every performance criterion, according to research. The study's findings show how artificial intelligence models can forecast data traffic models and how they can be used to build them.

Pedro Torres et al., [25]investigated Using Deep Neural Networks for Forecasting Cell Congestion on LTE Networks: A Simple Approach combines a dataset of collected downlink throughput samples from one cell in a region where cell congestion is common and a Deep Neural Network (DNN) approach to do short-term cell load predictions provided by a mobile network operator. The existing model could be improved further by fine-tuning the deep neural network technique. The authors are also working on integrating the technique in an LTE system-level simulator to quantify network performance improvements based on cell congestion prediction and SON tactics. The obtained findings suggest that, in comparison to other traditional forecasting approaches, the proposed model can forecast cell downlink throughput with high accuracy if a suitable dataset of samples is provided.

Nishant Kumar and Martin Raubal [26].Studied Applications of deep learning in congestion detection, prediction and alleviation: present the current state of deep learning applications in the areas of congestion detection, prediction, and relief. The goal of detecting, forecasting, and reducing traffic congestion is to improve the transportation network's level of service. Deep learning is becoming more relevant for such jobs as access to larger datasets with higher resolution grows. In recent years, a number of comprehensive survey studies have detailed the deep learning applications in the transportation area. Congestion that occurs on a regular basis and congestion that does not occur on a regular basis are explored separately. These recommendations are made in response to the issues highlighted in the previous section. They emphasize the necessity of data standardization, potential synergies with other fields, and potential synergies between simulation-based and deep learning systems for traffic prediction in this paper.

Sen Zhang et al., [27]developed Deep Auto encoder Neural Networks for Short-Term Traffic Congestion Prediction of Transportation Networks proposes an accessible and general workflow to acquire large-scale traffic congestion data and to create traffic congestion datasets based on image analysis. Develop a dataset called Seattle Area Traffic Congestion Status (SATCS) using traffic congestion map snapshots from the Washington State Department of Transportation's publicly available online traffic service. They then present a deep auto encoder-based neural network model with symmetrical encoder and decoder layers for learning and predicting traffic congestion in a transportation network. Our experiment equipment confined the study to just predicting traffic congestion levels using traffic congestion snapshots from a single data source. Will attempt to improve the computational capabilities of our experiment equipment in order to conduct more extensive trials, experiment with snapshots from other service providers, and fuse numerous types of data from various sources in order to construct a more comprehensive model..

Ogwueleka [28]user profiling and classification are important tasks in data intensive environments where computer assisted decision-making are sought for. The subscriber's call data describes the subscriber's calling behavior, which was used as the basis for modeling in this study. The learning algorithms in this study were designed to learn user profiles from phone data in order to make fraud detection choices. The techniques used in this work learn to detect fraud from partially labeled data, in which a call account was known to have been scammed but not when. As a result, the data is a combination of authentic and false information, with an unclear mixing method. Probabilistic models and neural networks were used in the fraud detection process. These models' ability to learn from data was regarded as a valuable feature, as was their ability to deal with ambiguity.

According to **Kusaksizoglu** [29] the telecom business loses a substantial amount of money due to fraud. Telecom providers may save a lot of money by using effective fraud detection and analysis systems. Automated fraud detection systems allow operators to respond to fraud by detecting it, denying service, and prosecuting perpetrators. The goal of this study was to use data mining techniques to investigate call detail records (CDRs), demographic data, and payment data of mobile users in order to construct models of normal and fraudulent behavior. First, they conducted exploratory data analysis (EDA) on the data set and identified that some characteristics, such as Account length, Package kind, Gender, Type, and Total Charged Amount, had a significant tendency for fraudulent use. They used the k-means cluster approach to group the customers based on their preferences.

Yeshinegus [30] investigate and come up with a prediction modeling for detecting incoming international calls that are terminated using local mobile numbers or fraud detection in telecommunication the case of ethio telecom. The derived model can be integrated with the existing system to detect frauds in telecommunication companies, specifically in ethio telecom. This is done by implementing derived models from data mining tools, techniques and algorithms. Yeshinegus this study CRISP-DM (Cross industry Standard process for data mining) model is used and classification method like J48 and PART from decision tree and multilayer Perceptron algorithms on data collected from ethio telecom. WEKA data mining tool used to design a model for predicting fraudulent activities. For this study prepaid sample voice (call detail record) CDR data has been used along with SMS, GPRS and other data such as pre-paid wallet recharge log from OCS and CCB data warehouse in ethio telecom. The experiment result showed that the model from the PART algorithm exhibited 100% accuracy level followed by J48 algorithm with 99.98%

Dereje [4] discovers hidden knowledge from ethio telecom mobile network data specifically GSM mobile so as to determine call setup success rate. According, to the research to overcome the drawback of simple statistical method they proposed data mining techniques, methods and methodology. In order to discover knowledge from the data they used the divisive hierarchical clustering methods to cluster the data. The K-means algorithm, WEKA tool and CRISP data mining process model are used. As a result, they shows the knowledge which was discovered during analysis of each cluster and the relationship between attribute against CSSR. The data shows the most of the call setup failed and categorized under very poor CSSR category. To enhance CSSR emphasize should be given to attributes used as KPIs. The study result reveals which attribute should enhance to improve the call setup success rate. Enhancing CSSR leas giving QoS to customers and it implies customer satisfaction and increases company revenues. Finally they recommended ET to apply data mining technique using cluster analysis on GSM mobile network data to analyze the data, evaluate the performance of the network, to assess the quality of the service and to make better decision. Yared developed a predictive model that can determine mobile call drops from ethio telecom mobile network data using data mining techniques. They implement data mining process, classification methodology. The data collected from fault management system, to build the model. They used WEKA for their study and four classification techniques such as J48, random forest algorithm from decision tree as well as PART and JRIP algorithm from rule induction are used. As a result, J48 decision tree algorithm with 10-fold cross validation registered better performance and processing speed 95.43% and 0.06 sec respectively.

Jember[12] develop an application of data mining in fraud detection on mobile communication service in the case of Ethio telecom mobile data using (Call detail record) CDR data. The CDR contains a vast amount of data about each call made and is a significant source of data for research to reveal hidden patterns of calls made by consumers, in addition to its normal use for bill processing tasks. He performs data mining with artificial neural networks using MATLAB. As a result, he discovered an accuracy rate of 89 percent and recommends further investigation into other likely causes of fraud on the switch's pure CDR data.

Gebremeskel [11] Detecting unlawful calls from ETC's CDR Switching machine and enabling early identification of those calls are important priorities. For their research, they used neural network techniques and Brain Maker neural network software. CDR was used as the study's data source, with an emphasis on ETC's Pre-paid mobile phone. As a result, fraudulent calls with an accuracy level of 88.465 are archived, but non-fraudulent calls with an error rate of 4.19 percent are not.

Asemelash [31] Research on 3G mobile fault occurrence prediction using Neural Networks was developed. The research was carried out utilizing a case study of Addis Ababa 3G mobile sites and a Nonlinear Auto Regressive (NAR) Neural Network time series prediction method. To train the neural network, they employed the Levenberg-Marquardt method and an iterative strategy of hidden layer neural number selection. As a result, the optimal model with the lowest mean square error of prediction is chosen. The model was further evaluated with actual fault occurrence time, which was not included in the training, and it achieved a prediction rate of 90.71 percent.

Aipeng Guo and Chunhui Yuan [32] Studied Network Intelligent Control and Traffic Optimization Based on SDN and Artificial Intelligence designed a network control and solution mechanism for network intelligent traffic optimization based on SDN and artificial intelligence. Additionally, they also analyzes the objectives of traffic optimization as well as routing calculation algorithms and routing optimization algorithms mainly focused on SDN-based network traffic algorithm optimization and experimental verification. Design a network control mechanism for network intelligent control as well as solutions for traffic optimization based on SDN and artificial intelligence. They analyze operators' network requirements (e.g., the carrying of the 5th generation mobile network (5G) service, multi-protocol label switching virtual private networks optimization, cloud of services and the IP backbone network). The proposed architecture consists of three modules, including a network status collection/perception module, an AI intelligent analysis module and an SDN controller module. future will try to apply network intelligent control and traffic optimization based on SDN and artificial intelligence solutions to operators' actual networks in order to solve real data problems in future work; at the same time, more network intelligent control application scenarios will be studied.

2.6 Chapter Summary

Machine learning is a field of study on computational methods in the learning process and how to apply computer-based learning systems to solve practical problems. As well, it is a way to improve model performances automatically by making use of accumulated datasets from computing machines. Data in machine learning plays a key role, and machine learning algorithms are used to discover and learn knowledge or properties from the data. Therefore, the Machine Learning model requires accuracy, precision and minimum error to have supervised predictive machine learning. In summary, during the review of related works, most of local researches focus on customer classification, telecom fraud detection and predict the cause of call drops and fault occurrence. Many proposals work with branches of Artificial intelligence (AI) such as machine learning, deep learning, or data analysis, in one effort to solve problems easily and predict future system characteristics. The machine learning approach helps with the design and implementation of a system that learns automatically. For instance, a comparison between regression models based on a machine learning approach to predict traffic using a virtual network is presented; in which authors describe better accuracy results using real-life data.

Someone has developed LSTM based prediction models by using machine learning approaches, which involve structure designing or network training designing and prediction and implication. Another goal is to deal with prediction errors that may occur during the prediction process with deep learning methods. The discussed method has been applied to big data that has been collected from the performance measurement system (PEMS).the experiments show that the LSTM model has many capabilities and good performance results as compared to shallow machine learning methods.

The thesis has conducted scientific literature review to assess the major issues and concepts in the predict mobile network congestion based on machine learning, prediction telecom service and other related concepts are considered to confirm the impact of the research. An assortment of different journals, books, papers and articles are grasped from the internet, library, manuals and e-books.

Hence there is a gap to study the data generated from telecommunication mobile network to predict mobile network congestion using dynamic prediction models .The result of this study helps the network provider to improve quality of service of the mobile network. In review of related work the time takes during training for prediction is long but in this thesis model takes only 5.4sec.Most use Mat lab but MLP-NN and python tool using Anaconda software is preferable. The No of layers were tried but no signifance change were observed with the prediction result. The algorithm used in this thesis is Adam optimization which is best NN, because this algorithm minimize the error when mapping from input to output.

Table 2. 1Summary of Review of Related Works

No	Author	Objective	Algorithm(s) Used	Model Used	Metrics
1	**Siddiqui and Choudhary** [20]	The study is under taken predict voice traffic congestion in busy hours. To provide maximum utilization of voice traffic, they observe QoS report at constant time interval.	multilayered feed forward NN with back propagation	feed-forward neural network	Network quality and reduce loss of congestion.
2.	**Simone Manganite, Michael Schapiro** [7]	MORC is presented a novel rate-control protocol for mobile networks.	MORC's online learning algorithm	PCC framework	Bandwidth Utilization
3.	**Sneha Kumar Kasera, Ramachandran Ramjee, Sandra R. Thuel** [15]	The problem of congestion control in the IP RAN of a CDMA wireless access network and examined three control techniques,	Support Vector Regression (vSVR) algorithm.	Mobility Model	user mobility, call generation, call termination, and soft-handoffs
4.	**Min Liu et al.,** [19]	Proposed and implemented an algorithm of prediction for optical networks	Multiple layer perception Neural Network	deep neural networks including Long Short-Term Memory	congestion degree and bandwidth or holding time of services

		congestion degree by constructing BP-ANN.		(LSTM) and Gated Recurrent Units (GRU)	
5.	**Guiomar Corral et al,** [16]	They propose a system to reduce short-term congestion in ATM networks	UPC algorithm	Node, Process model,OPNET	buffer utilization
6.	**Raheem and Okeene** [9]	They propose a GSM congestion prediction model based on multilayer Perceptron neural networks (MLP-NNs) with sigmoid activation function and Levenberg Marquardt Algorithms (LMA) using twelve month real traffic data.	Levenberg-Marquardt Algorithm	A Multi-Layer Perceptron Feed-forward neural network	Network quality and reduce loss of congestion
7.	**Sophia & Olatokun** [14]	Findings from the study showed that apart from the carrying capacity of the MTN network	multilayered feed forward NN with back propagation	feed-forward neural network	Network capability
8.	**Yeshinegus** [31]	The derived model can be integrated with the existing system to detect frauds in telecommunication companies, specifically in ethio telecom	PART and J48	CRISP-DM (Cross industry Standard process for data mining	detect frauds

9.	**Dereje [4]**	They shows the knowledge which was discovered during analysis of each cluster and the relationship between attribute against CSSR	K-means algorithm	WEKA tool and CRISP data mining process model.	performance of the network
10.	**Asemelash**[32]	The research work was conducted base on Nonlinear Auto regressive(NAR), Neural Network time series prediction method using Addis Ababa 3G mobile sites in a case study.	Levenberg-Marquardt algorithm	A Multi-Layer Perceptron Feed-forward neural network	fault occurrence time

CHAPTER THREE

METHODLOGY

3.1 Machine learning

Machine learning is an application of AI that provides systems the ability to learn and improve automatically from experience without being explicitly programed. Which are used for future predictions (based on past data or Big Data*)* and identifying (discovering) patterns in data. Machine learning is itself a type of artificial intelligence that allows software applications to become more accurate in predicting outcomes without being explicitly programmed.

Machine learning is the ability for a computer to output or does something that it wasn't programmed to do. While machine learning emphasizes making predictions about the future, artificial intelligence typically concentrates on programming computers to make decisions. If you use an intelligent program that involves human-like behavior, it can be artificial intelligence. However, if the parameters are not automatically learned (or derived) from data, it's not machine learning [33].

3.2 Understanding of the data

3.2.1 Data Collection

When providing telecom services, the telecom industry generates and retains a massive amount of data. Call detail data, which defines the calls that cross telecommunication networks, network data, which describes the condition of hardware and software components in the network, and customer data, which describes telecom consumers, are among these data. [34, 43].

When a user makes a call, call detail data is generated, and each call's information is kept in the database. All network elements generate network data, such as network element status, call setup information, and so on. Customer data comprises detailed customer information as well as information about the customer's location. [7, 32].

These massive amounts of data are used for a variety of purposes, including fraud detection, network performance analysis, customer churn prediction, reporting to higher-ups for network design and optimization, and decision support.

From the above telecommunications data, on this research we used network data. This data was obtained from the Key Performance Indicator (KPI) database for the 3G wireless mobile telecommunications technology [35]. The data includes network performance parameters recorded for the network for six months from 2021, June 6 to 2021, November 6 in from three sites in Addis Ababa [36]. This data can be extracted to various file formats like Excel, PDF and CSV file type; for this research the data was processed under the Excel (xls) and CSV (csv) formats. The xls format was used to pre-process the data table under the MS Excel and the csv formats have been used to read the data into the jupyter notebook environment and perform data manipulations and feed the data into the proposed models.

The data from the KPI server is only extracted by authorized personnel or domain expert. So, to get the official data permission is needed. Managing the huge KPI data was one of the challenges and time consuming because as the size of the data increases, preprocessing this huge amount data becomes more complex and tedious. After eliminating irrelevant and unnecessary data, a total of 3 datasets for each site containing 1080 rows of data were used for the purpose of conducting this study. We also select 15 attributes out of 20 attributes form the original data based on their relation with mobile network congestions which is our target parameter of prediction.

There are two options to extract the data at Radio access Network from the system.

Extract the data at BSC level: It maintains the radio resources of a collection of Base Transceiver Stations (BTSs) [37, 40]. A BSC is in charge of each controlled Base Transceiver Station's handover or handoffs, frequency hopping, exchange operations, and power regulation.

The other one is at BTS level: It corresponds to the transceivers and antennas utilized in each network cell. It is frequently found in the cell's middle. The size of a cell is determined by its transmission power. Based on the hard ware provider vendor, each BSC in Addis can serve 5.7 million subscribers. Ethio telecom's capacity is configured dependent on the number of customers it has. A license is required to use the BSC's maximum capacity or to upgrade its capacity. The mobile network subsystem, often known as the core network, is made up of a number of different components. It is responsible for the overall control and interfacing of the mobile network.

3.2.2. Description of the collected data

Description of the data is very important in research tasks that involve data analysis and machine learning techniques in order to extract useful information and clearly understand the data. Without such an understanding, useful application of the data cannot be developed. In the previous discussion we have mentioned that this research was conducted by collecting data from ethio telecom 3G KPI database for a six months period.

In this research a Key Performance Indicator (KPI) data for three different sites in Addis Ababa was considered. These datasets consists of 20 network attributes out of which 15 of them will be used for mobile network congestion prediction. Each dataset contains network parameters recorded at network database for a total of 180 days. For each of these days the dataset displays six rows of key network performance indicators. Therefore, for each site the data set presents a total of 1080 rows of parameter records. As Stated above the dataset consisted of various network parameters of which 15 attributes that are related to the mobile network congestion are explained below[38,42].

A, Cell Availability

Cell Availability identifies success rate of radio access network availability in a selected region. Ideally cell availability is greater than or equal to 99%.

Cell Availability report can help the Network Operations team to:

- Identify issues with network access during roaming.
- Identify BTS having higher non-availability of radio network.
- Identify failure reasons for lower cell availability rate.
- Provide cell availability timeline for a selected BTS.
- Provide failure reason timelines to understand variations and pattern in failure parameters

B, Transceiver Availability (TRX Av.)

A transceiver is a transmitter/receiver that comes in one box. While the phrase is most commonly associated with wireless communications equipment, it can also refer to transmitter/receivers in cable and optical fiber networks. As a result, the availability of Transceiver for each user in a cell is quantified by this parameter.

C, Call Setup Success Ratio (CSSR)

The call setup success rate (CSSR) is the percentage of calls attempted that are connected to the phoned number (due to various reasons not all call attempts end with a connection to the dialed number). This number is usually expressed as a percentage of total call attempts.

When a call is attempted, a call setup procedure is initiated, which if successful, results in a connected call. For a variety of technical reasons, a call setup operation may fail. Failed call attempts are the term used to describe such calls. In many circumstances, this description will need to be supplemented with a number of precise details indicating which calls are counted as such.

Which ones were set up successfully and which ones were not. The stage of the call setup procedure at which a call is counted as connected determines this to a large extent. The call setup method in modern communications systems, such as cellular (mobile) networks, can be highly complex, and the point at which a call is regarded properly linked can be characterized in a variety of ways, influencing the calculation of the call setup success rate. The call is counted as successful if it connects successfully but the phoned number is busy.

Blocked calls is another word for call attempts that fail during the call setup phase. In traditional (so-called land-line) networks, the call setup success rate is extraordinarily high, significantly exceeding 99.9%. The call establishment success rate in mobile communication systems using radio channels is lower, ranging between 90% and 98 percent or more for commercial networks. The main causes of failed call setups in mobile networks are a lack of radio coverage (either in the downlink or uplink), radio interference between different subscribers, flaws in the network's operation (such as failed call setup redirect procedures), overload of network elements (such as cells), and so on.

One of the key performance indicators (KPIs) used by network operators to analyze the operation of their networks is the call setup success rate. It is thought to have a direct impact on user satisfaction with the network and its operator's services. The call setup success rate is frequently included in a critical performance indicator known as service accessibility, along with other network technical metrics.

The operators of telecommunication networks aim at increasing the call setup success rate as much as practical and affordable. In mobile networks this is achieved by improving radio coverage, expanding the capacity of the network and optimizing the performance of its elements, all of which may require considerable effort and significant investments on the part of the network operator.

Call Setup Success Rate (CSSR) before diving into the KPI in Telecom, we should discuss some background information.

Call Setup in Telecommunication is a procedure in which a virtual circuit is established across the telecommunications network by using a signaling protocol.

Call Set-up Time can be understood in two ways;

- The overall length of time needed for the purpose of establishing a circuit-switched call
- Data communication- the overall length of time needed for establishing a circuit-switched call, that is, from initiating a call request to the beginning of a call message, between terminals

Definition: Call Setup Success Rate or CSSR

Call Setup Success Rate is one of the KPIs that has a direct impact on customer satisfaction with a network's and operator's telecom service.

It is used by network operators to assess network performance. CSSR is a term that describes the percentage of call attempts that result in a successful connection to the phoned number. The percentage of all attempted calls is the fraction in the procedure.

In terms of telecom, a call that has been successfully connected can be defined in a variety of ways. Let's say the number you dialed is busy, but the call is linked to the dialed number, thus it's considered a successful connection.

In the case of, the CSSR is extremely high, exceeding 99.9%.

D, Stand Alone Dedicated Control Channel (SDCCH)

Slow Associated Control Channel (SACCH) and Stand Alone Dedicated Control Channel (SDCCH) are acronyms for Slow Associated Control Channel and Stand Alone Dedicated Control Channel, respectively. The SDCCH channel is used to send and receive signaling messages between GSM mobile phones and the network (base station). This is used solely for signaling or to establish a traffic connection. In the AGCH response from the base station for the RACH transmitted by MS, SDCCH is assigned to the mobile. The SDCCH allocation is only temporary. The channel has been launched. It's used for location updates, and once that's done, the procedure is declared to be finished, and the SDCCH is switched on is released. If this SDCCH is used to establish the traffic channel connection, the SDCCH channel is likewise relinquished after the connection is formed.

When a dedicated channel, such as a dedicated traffic channel or a dedicated SDCCH channel, is assigned to a Mobile Subscriber (MS). Another route is now essential for exchanging measurement data on a regular basis. SACCH channel is provided in the GSM system to accomplish this. The data pace is extremely sluggish, therefore the name.

A single SACCH slot exists in a multiform structure, and the whole information of SACCH transmitted by a standard burst requires four bursts. As a result, carrying a SACCH message requires four multi frames. This means that a single SACCH message can be delivered in around 480 milliseconds. This is why the channel is said to as slow. One SACCH slot is on the 12th frame and another is on the 25th frame for successive timeslots.

E, TCH Drop Rate (TCH Dr.)

The TCH drop rate refers to the ratio of call drops to successful TCH seizures after the base station controller successfully assigns TCHs to mobile stations (MSs). The TCH call drop rate can be measured from the following aspects

- TCH call drop rate (including handover)
- TCH call drop rate (excluding handover)

The TCH call drop rate is linked to retention ability, which is one of the most important KPIs for telecom carriers. It displays the likelihood of call dropouts after MSs Access TCHs for various reasons. TCH call drop rates that are too high degrade the user experience and contribute to mobile network congestion.

TCH call drop rate (including handover) = (Number of TCH call drops + Number of TCH call drops during very early assignment)/Number of successful TCH seizures * 100%.

F, Handover success rate (HOSR)

In cellular telecommunications, the terms handover or handoff refer to the process of transferring an ongoing call or data session from one channel connected to the core network to another channel.

The HOSR is the ratio of the number of successful handovers to the number of handover requests. The major purpose of handover is to guarantee call continuity, improve speech quality, reduce cross interference in the network, and thus provide better services for mobile station (MS) subscribers.

G, Circuit switching (CS Traffic)

Circuit switching was designed specifically for voice communication and is not ideal for data transmission. In circuit switching, a dedicated channel must be created between the sender and receiver before they can speak to one another. Circuit switching is most often seen in telephone systems that require a dedicated, physical path.

Circuit switching, which is set up at the physical layer, sends the entire message through the dedicated channel. This type of switching isn't ideal for data transmission because data is sent and received in streams, meaning the line would remain idle in between transmission spurts. That would be a waste of bandwidth.

Advantages of circuit switching over packet switching

- Decreases the delay the user experiences before and during a call
- The call will be done with a steady bandwidth, dedicated channel, and consistent data rate
- Packets are always delivered in the correct order

Disadvantages of circuit switching:

- Great for only voice communication
- Doesn't use resources efficiently
- Dedicated channels for circuit switching are unavailable for any other use
- There is a higher cost to dedicate one channel per use

H, Packet Switching (PS Traffic)

Unlike circuit switching, packet switching does not require the use of a dedicated channel. Packet-based networks break down a message into smaller data packets which then look for the most efficient route available. For efficiency's sake, each data packet could go a different route.

The header address contains the source and destination nodes. Once all of the data packets reach the correct destination, the packets are extracted and reassembled to create the sender's original message.

Packet switching is most often used for data and voice applications that aren't time-sensitive.

Advantages of packet switching over circuit switching:

- More efficient than circuit switching
- Data packets are able to find the destination without the use of a dedicated channel
- Reduces lost data packets because packet switching allows for resending of packets
- More cost-effective since there is no need for a dedicated channel for voice or data traffic

Disadvantages of packet switching:

- Not ideal for applications that are in constant use, such as high-volume voice calls
- High-volume networks can lose data packets during high-traffic times; those data packets cannot be recovered or resent during transmission
- There is a lack of security protocols for data packets during transmission

While circuit switching and packet switching are the most common methods of transferring data across networks, choosing the right one depends on your specific business needs when it comes to voice and data transfer.

If your goal is to establish clear, reliable voice communication channels, circuit switching may be your best option. If your goal is to facilitate multiple voice and data applications at the same time, then packet switching may be your best option.

I, Traffic channel congestion (TCH)

TCH (Traffic channel) congestion is one of the key performance indicators (KPI) which influence the network performance and customer satisfaction in live GSM network The degree of TCH congestion in the network results in large number of TCH blocking which greatly affects the subscriber satisfaction and revenue of the service provider. The available congestion relief methodologies such as cell splitting, aggressive frequency reuse pattern, microcells and expanding frequency band depend on time and cost factor for implementation. A hybrid model to reduce TCH congestion by balancing the traffic between high and low utilized cells of live GSM network is proposed. The model depends on the radio parameters such as CBQ (Cell Bar Qualify) RXACCMIN (Receive Access Minimum) and TILT which contributes to balance the traffic and maintain congestion level of the network within specified limit.

The properties, together with their data types and explanations, are listed in table 3.1 below, taken from the above-mentioned source. The attributes have different data types like date, string, and nominal and numeric data type.

Table 3. 3 Data types and descriptions

No.	**Attribute Name**	**Data Type**	**Description**	**Missing Values In the Original data**
1	Date	-	The date when the data was recorded.	0%
2	GBSC	String	-	0%
3	Site Name	String	The site of the data record.	0%
4	Cell Name	String	The cell represented by the data	0%
5	Integrity	Float		0%
6	TR373: Cell Availability (%)	Float	Cell Availability identifies success rate of RAN availability in a selected region.	2%
7	TRX Availability Rate (%)	Float	The availability of transceivers in a specific cell.	4%
8	CSSR- Innovis	Float	The fraction of the attempts to make a call that result in a connection to the dialed number.	5%

9	SDCCH Congestion Rate (%)	Float	The congestion rate of the channel used for exchange of signaling messages between GSM mobile and network.	0%
10	TCH Congestion Rate (%)	Float	The rate of traffic channel congestion also called mobile congestion.	0%
11	TCH Drop Rate	Float	The TCH drop rate refers to the ratio of call drops to successful TCH seizures after the base station controller successfully assigns TCHs to mobile stations.	0%
12	Call Drop Rate on TCH	Float		0%
13	HOSR- Innovis	Float	The ratio of the number of successful handovers to the number of handover requests.	0%
14	SDCCH Resource Utilization Rate	Float	The factor of available Stand-alone Dedicated Control Channel Utilization.	0%
15	TCH Resource Utilization Rate	Float	The factor of available Traffic Channel Congestion.	0%
16	CS Traffic (Erl) EG	Float	The set up rate at the physical layer, which sends the entire message through the dedicated channel.	0%
17	PS Traffic(GB)	Float	Refers to the transfer rate of small pieces of data across a specific radio access network.	3%
18	R473:TRX Out-of-Service Duration(s)	Float	Refers to the duration where a subscriber is unable to access transceivers in the site.	2%
19	K3015:Available TCHs	Float	The number of available traffic channels available for the user to hook	1%

			up with while making call attempts.	
20	K3005:Available SDCCHs	Float	The number of available stand-alone dedicated channels in a specific radio band frequency for exchanging control signals while making calls.	0%
21	S3656:Number of available TRXs in a cell	Float	The number of available transceivers in a cell for establishing new connections.	0%
22	CR373:Cell In-Service Duration(s)	Float	The duration for which each cell is active in the course of 24 hours period.	27%
23	R373:Cell Out-of-Service Duration(s)	Float	The duration for which a cell is not active (out of service) in the course of 24 hours.	20%

3.2.3 Preparation of the Data (Data Pre-processing)

Due to their often large size and likely origin from various, diverse sources, today's real-world databases are especially sensitive to noisy, missing, and inconsistent data. Data of poor quality will yield poor mining results. As a result, data preparation is required in order to obtain a data set appropriate for analysis. Data cleaning to reduce noise or handle missing values, relevance analysis to remove irrelevant or redundant attributes, and data transformation, such as generalizing the data to higher-level concepts or normalizing the data, are all examples of preprocessing data in preparation for classification and prediction.

The goal of data preparation is to improve the quality of selected data [39]. Data quality is a complicated issue that is one of the most difficult problems to solve in data mining. It relates to the data's accuracy and completeness. The structure and consistency of the data being studied can also have an impact on data quality. Duplicate records, a lack of data standards, update timeliness, and human mistake can all have a substantial impact on the effectiveness of more complicated data mining algorithms, which are sensitive to little changes in the data. Cleaning data, which might include removing duplicate entries and normalizing the numbers used to represent information in order to improve data quality, is occasionally necessary [41].

Because the data is so large, they put it in moist files, some of the data may have different forms. The data must then be converted into a usable format before it can be used. Because the preprocessing stage's goal is to clean up the data as much as possible and convert it into a format that may be used in subsequent stages.

Data Selection

The data selection step was used to create a dataset that relates 15 input attributes to the mobile network congestion which is the target of prediction. Therefore the new datasets will contain 15 attributes and 1080 rows of data for each of the three sites. To come up with such a clean data set the following attributes from the original data were removed as shown in the table below.

Table 3. 4 Removal Attributes

No.	Attribute Name	Attribute Data Type	Reason for Removal
1	Date	-	No relevant information.
2	Site Name	String	The site name was used to partition the data set into three separate partitions.
3	Integrity	Float	No relevant information.
4	S3656:Number of available TRXs in a cell	Float	No relevant information.
5	CR373:Cell In-Service Duration(s)	Float	No relevant information.
6	R373:Cell Out-of-Service Duration(s)	Float	No relevant information.

The necessity of decreasing the amount of characteristics not only speeds up the learning process, but it also avoids most learning algorithms from being misled into producing a subpar model due to the existence of numerous irrelevant or duplicate qualities. As a result, those traits that are less important for this research, as well as the other features listed in table 3.3, make up the final list of attributes employed in this study. In general, using the Data Selection and Preprocessing approach on our data, three datasets and 15 attributes that are substantially connected to mobile network congestion were chosen for use in this study.

Table 3. 5 Selected Attributes

No.	**Attribute Name**	**Data Type**	**Description**	**Abbreviations**
1	Cell Name	Int	The name of the cell for the data record.	CELL_CAT
2	TR373: Cell Availability (%)	Int	Cell Availability identifies success rate of RAN availability in a selected region.	Cell Av.
3	TRX Availability Rate (%)	Float	The availability of transceivers in a specific cell.	TRX Av.
4	CSSR- Innovis	Float	The fraction of the attempts to make a call that result in a connection to the dialed number.	CSSR
5	SDCCH Congestion Rate (%)	Float	The congestion rate of the channel used for exchange of signaling messages between GSM mobile and network.	SDCCH
6	TCH Drop Rate	Float	The TCH drop rate refers to the ratio of call drops to successful TCH seizures after the base station controller successfully assigns TCHs to mobile stations.	TCH Dr.
7	Call Drop Rate on TCH	Float		Call Dr.

8	HOSR- Innovis	Float	The ratio of the number of successful handovers to the number of handover requests.	HOSR
9	SDCCH Resource Utilization Rate	Float	The factor of available Stand-alone Dedicated Control Channel Utilization.	SDCCH Res.
10	TCH Resource Utilization Rate	Float	The factor of available Traffic Channel Congestion.	TCH Res.
11	CS Traffic (Erl) EG	Float	The set up rate at the physical layer, which sends the entire message through the dedicated channel.	TCH Res.
12	PS Traffic(GB)	Float	Refers to the transfer rate of small pieces of data across a specific radio access network.	PS Traffic
13	R473:TRX Out-of-Service Duration(s)	Float	Refers to the duration where a subscriber is unable to access transceivers in the site.	TRX Out
14	K3015:Available TCHs	Float	The number of available traffic channels available for the user to hook up with while making call attempts.	TCH Av.
15	K3005:Available SDCCHs	Float	The number of available stand-alone dedicated channels in a specific radio band frequency for exchanging control signals while making calls.	SDCCH Av.
16	TCH Congestion Rate (%)	Float	The rate of traffic channel congestion also called mobile congestion.	TCH

Major Tasks in Data Preprocessing:

1. Data cleaning
2. Data integration
3. Data reduction
4. Data transformation

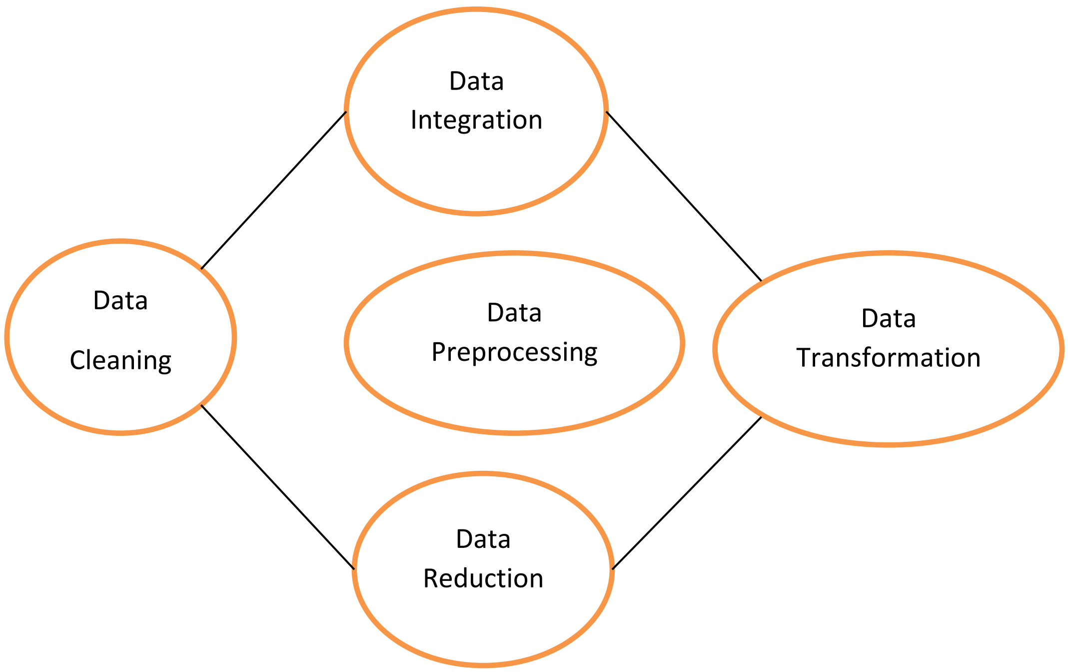

Figure 3. 1 Data Preprocessing Path

Data cleaning:

Data cleaning encompasses a wide range of actions and activities aimed at detecting and correcting data problems. This procedure entails locating and correcting flaws in the dataset that could have a detrimental impact on the information extracted from the data set, resulting in poor model development.

The dataset was scoured for missing rows, erroneous values in each attribute, and data types that did not match the data types defined for each attribute. The removal of these fields was done since there are few records of this nature, and their removal will have no significant impact on the model's generalization for the target variable mobile network congestion. MS Excel 2013 was used to finish this step. This stage included the following major activities:

Table 3. 4 Removal of attributes which have redundant information

No.	Attribute Name	Attribute Value	Reason for Removal
1	Integrity	Numeric	No relevant information and all values are 100%
2	S3656:Number of available TRXs in a cell	Numeric	No relevant information.
3	CR373:Cell In-Service Duration(s)	Numeric	No relevant information.

Table 3.5 Removal and replacement of site names

No.	Site Name in the Original Data	Represented Site
1	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA	Site 1
2	111436_AC+DG_GUL_BSCRNC4.HW.AKKWHS.SAAZ.AA	Site 2
3	111567_H7_AC+DG_GUL_BSCRNC4.HW.KLTMINAGRI.SAAZ.AA	Site 3

Table 3.6 Replacement of Cell Name with integers for each cell in order to feed the model

No.	Cell Name	Cell Category Label(CELL_CA)
1	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA_D1	1
2	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA_D2	2
3	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA_D3	3
4	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA_G1	4
5	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA_G2	5
6	111268_AC_GUL_BSCRNC4.HW.KAAWGOST.SAAZ.AA_G3	6

Data integration:

The process of combining multiple sources into a single dataset. The Data integration process is one of the main components in data management. There are some problems to be considered during data integration.

- **Schema integration**: Integrates metadata (a set of data that describes other data) from different sources.
- **Entity identification problem:** Identifying entities from multiple databases. For example, the system or the use should know student _id of one database and student name of another database belongs to the same entity.
- **Detecting and resolving data value concepts**: The data taken from different databases while merging may differ. Like the attribute values from one database may differ from another database. For example, the date format may differ like “MM/DD/YYYY” or “DD/MM/YYYY”.

Data reduction:

This process helps in the reduction of the volume of the data which makes the analysis easier yet produces the same or almost the same result. This reduction also helps to reduce storage space. There are some of the techniques in data reduction are Dimensionality reduction, Numerosity reduction, Data compression.

- **Dimensionality reduction:** This process is necessary for real-world applications as the data size is big. In this process, the reduction of random variables or attributes is done so that the dimensionality of the data set can be reduced. Combining and merging the attributes of the data without losing its original characteristics. This also helps in the reduction of storage space and computation time is reduced. When the data is highly dimensional the problem called "Curse of Dimensionality" occurs.
- **Numerosity Reduction:** In this method, the representation of the data is made smaller by reducing the volume. There will not be any loss of data in this reduction.
- **Data compression:** The compressed form of data is called data compression. This compression can be lossless or lossy. When there is no loss of information during compression it is called lossless compression. Whereas lossy compression reduces information but it removes only the unnecessary information.

Data Transformation:

The change made in the format or the structure of the data is called data transformation. This step can be simple or complex based on the requirements. There are some methods in data transformation.

- **Smoothing**: With the help of algorithms, we can remove noise from the dataset and helps in knowing the important features of the dataset. By smoothing we can find even a simple change that helps in prediction.
- **Aggregation**: In this method, the data is stored and presented in the form of a summary. The data set which is from multiple sources is integrated into with data analysis description. This is an important step since the accuracy of the data depends on the quantity and quality of the data. When the quality and the quantity of the data are good the results are more relevant.
- **Discretization**: The continuous data here is split into intervals. Discretization reduces the data size. For example, rather than specifying the class time, we can set an interval like (3 pm-5 pm, 6 pm-8 pm).
- **Normalization:** It is the method of scaling the data so that it can be represented in a smaller range. Example ranging from -1.0 to 1.0.

3.2.4. Data formatting

Before starting to work with the data on jupyter notebook, the format of the data was changed in MS Excel where the data should be arranged into input attributes on one side and output (target) attribute on the other side. Then the excel file is just exported into a csv file format that can be easily read and handled by the libraries and functions in jupyter notebook.

3.2.5. Data splitting

Data splitting is commonly used in machine learning to split data into a train, test, or validation set. This approach allows us to find the model hyper-parameter and also estimate the generalization performance. It is meant to get an unbiased estimate of algorithms performance in the real world data. A machine learning model is built upon the data in the training set then hyper parameters are optimized on the validation set as much as possible then after the model is ready; the performance is evaluated on the test dataset.

In this research the three data sets are divided into train and test sets at the proportion percentage of 30% and 70% respectively. The validation data sets were taken out of the training data using = function in the library.

Training Set

The training set is the 70% of the complete data set for each of the three sites that is used to build a model that can best fit into the conditions in each site. The models observe and learn from this data and optimize its parameters. Such that in our case 756 rows of the each dataset are used as a training set to develop two models that are supposed to fit into the sites mobile network congestion situations utilizing 15 attributes.

Validation Set (Cross-Validation Set)

We select the appropriate model or the degree of the polynomial (if using regression model only) by minimizing the error on the cross-validation set.

Test set

The test set is the sample of data which is used to provide an unbiased evaluation of a final model fit on the training dataset. It is only used once the model is completely trained using the training and validation sets. Therefore, test set is the one used to replicate the type of situation that will be encountered once the model is deployed for real-time use. In this research 30% of the data out of each datasets is used as a Test set. Such that 324 rows of the data will be used to test the performance of the model in real world conditions (scenarios).

3.3. Machine learning model

3.3.1. Neural Network Architecture

Neural Network is a machine that is designed to model the way in which the brain performs a specific task or function of interest. "It is usually implemented by using electronic components or is simulated in software on a digital computer. To achieve good performance, neural networks employ a massive interconnection of simple computing" cells referred to as "neurons" or "processing units [44, 45]. A neural network is a massively parallel distribution processor made up a simple processing unit that has a natural tendency for storing experimental knowledge and making it available for use as views a neural network as an adaptive machine.

A well trained artificial neural network can perform many complex tasks such Classification, optimization, control and approximation function [22]. An ANN structure consists of interconnected artificial neurons that usually consist of three layers as defined below.[23]

Input layer: is a layer that communicates with the outside world and does not do any computations, only transferring input signals to other neuronal layers. The system's input data can be numerical or binary. The input layer is a user-defined data vector with a fixed length. [46].

Hidden layer: The intermediate layer between the input and output layer known as Hidden layer. It is used to calculate the weighted sum of the input.

Output layer: This layer receives signals from neurons in the hidden layer. It represents the output containing fixed length vector of data.

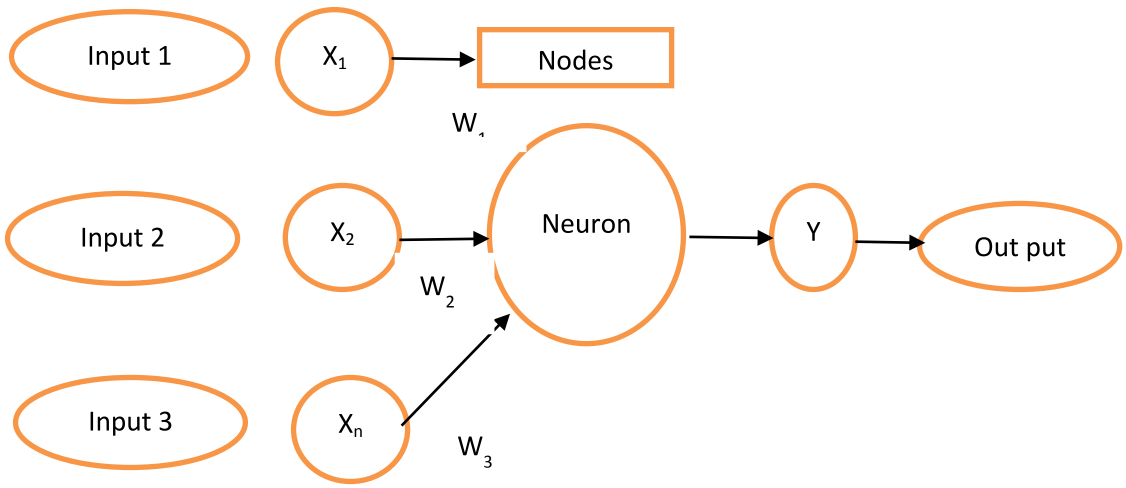

Figure 3. 2 Typical Artificial Neural Network Looks

3.3.2 Types of Artificial Neural Network:

Artificial Neural Networks (ANN) come in a variety of shapes and sizes, and they all accomplish tasks in the same way as human brain neurons and networks do. The majority of artificial neural networks will share some characteristics with a more complex biological partner and will perform admirably in their intended roles, for example Segmentation and categorization. [48].

Feed-Forward ANN:

A feed-forward network is a type of neural network that has an input layer, an output layer, and at least one neuron layer. The intensity of the network can be detected based on group behavior of the connected neurons, and the output is determined by assessing its output by examining its input. This network's main benefit is that it learns how to assess and detect input patterns.

Feed –Backward ANN: The output of this type of ANN returns to the network, allowing it to achieve the best-evolved results internally. According to the Lowell Centre for Atmospheric Research at the University of Massachusetts. Feedback networks send information back into themselves, making them ideal for solving optimization problems. Feedback ANNs are used to fix errors in the internal system. [54].

Convolutional Neural Networks

They're comparable to feed forward networks, but they're more commonly used for image recognition, pattern recognition, and computer vision. These networks use linear algebra principles, notably matrix multiplication, to find patterns in images.

Recurrent Neural Networks

Are identified by their feedback loops. These learning algorithms are primarily leveraged when using time-series data to make predictions about future outcomes, such as stock market predictions or sales forecasting.

Use MLP in Network congestion

MLP-NN is suitable for classification prediction problems where inputs are assigned a class or label. They are also suitable for regression prediction problems where a real-valued quantity is predicted given a set of inputs. Data is often provided in a tabular format, such as you would see in a CSV file or a spread sheet [48].

- To identify Network Congestion Level
- Classification prediction problems
- For perfectly congestion level Estimation
- They are very flexible and can be used generally to learn a mapping from inputs to outputs.

This flexibility allows them to be applied to other types of data. For example, the pixels of an image can be reduced down to one long row of data and fed into a MLP. The words of a document can also be reduced to one long row of data and fed to a MLP. Even the lag observations for a time series prediction problem can be reduced to a long row of data and fed to a MLP.

Primarily leveraged for deep learning algorithms, neural networks process training data by mimicking the interconnectivity of the human brain through layers of nodes. Each node is made up of inputs, weights, a bias (or threshold), and an output. If that output value exceeds a given threshold, it “fires” or activates the node, passing data to the next layer in the network. Neural networks learn this mapping function through **supervised learning**, adjusting based on the loss function through the process of **gradient descent.** When the cost function is at or near zero, we can be confident in the model's accuracy to yield the correct answer

Why MLP?

They are very flexible and can be used generally to learn a mapping from inputs to outputs. As such, since the network data is in tabular dataset it is recommended to use MLP for the following reasons

1. MLPs have the ability to learn and model non-linear and complex relationships, which is really important because in real-life, many of the relationships between inputs and outputs are non-linear as well as complex.

2. MLPs can generalize — after learning from the initial inputs and their relationships, it can infer unseen relationships on unseen data as well, thus making the model generalize and predict on unseen data.

3. Unlike many other prediction techniques, MLP does not impose any restrictions on the input variables (like how they should be distributed). Additionally, many studies have shown that MLPs can better model hetero scedasticity i.e. data with high volatility and non-constant variance, given its ability to learn hidden relationships in the data without imposing any fixed relationships in the data. This is something very useful in network congestion forecasting where data volatility is very high.

3.4 Machine learning method

Machine learning classifiers fall into three primary categories.

A, Supervised machine learning

The use of labeled datasets to train algorithms that reliably classify data or predict outcomes is characterized as supervised learning, often known as supervised machine learning. As more data is introduced into the model, the weights are adjusted until the model is properly fitted. This happens during the cross-validation phase to verify that the model does not over fit or under fit. Organizations can use supervised learning to tackle a range of real-world problems at scale, such as spam classification in a distinct folder from your email. Neural networks, naive Bayes, linear regression, logistic regression, random forest, support vector machine (SVM), and other approaches are used in supervised learning. A training set is used in supervised learning to teach models to produce the desired results.

Over fitting: Excellent performance on training data, but poor generalization to new data. Poor performance on the training data, as well as poor generalization to additional data. A model that over fits the training data is referred to as over fitting. When a model learns the information and noise in the training data to the point where it severely influences the model's performance on fresh data, this is known as over fitting. This means that the model picks up on noise or random fluctuations in the training data and learns them as ideas. Under fitting is a data science scenario in which a data model fails to accurately capture the relationship between input and output variables.49].

When a model tries to anticipate a trend in data that is excessively noisy, it is called over fitting. This is the result of a model that is extremely complicated and has too many parameters. Because the trend does not reflect the reality in the data, an over fitted model is erroneous. This can be determined if the model performs well on the observed data (training set) but badly on the unknown data (test set). A machine learning model's purpose is to generalize well from training data to any data from the domain of interest. This is critical because we want our model to generate future predictions based on data it has never seen before.

In this paper we have used two known techniques to avoid over fitting of the model. These are

1. Simplifying the Model

The first step when dealing with over fitting is to decrease the complexity of the model. To decrease the complexity, we can simply remove layers or reduce the number of neurons to make the network smaller. While doing this, it is important to calculate the input and output dimensions of the various layers involved in the neural network. There is no general rule on how much to remove or how large your network should be. But, if your neural network is over fitting, try making it smaller.

2. Early Stopping

Early stopping is a form of regularization while training a model with an iterative method, such as gradient descent. Since all the neural networks learn exclusively by using gradient descent, early stopping is a technique applicable to all the problems. This method update the model so as to make it better fit the training data with each iteration. Up to a point, this improves the model's performance on data on the test set. Past that point however, improving the model's fit to the training data leads to increased generalization error. Early stopping rules provide guidance as to how many iterations can be run before the model begins to over fit.

Supervised learning can be separated into two types of problems when data mining—classification and regression:

- **Classification** employs an algorithm to assign test data to certain groups with precision. It recognizes certain entities in the dataset and makes educated guesses about how those entities should be labeled or defined. Linear classifiers, support vector machines (SVM), decision trees, k-nearest neighbor, and random forest are some of the most common classification algorithms.
- **Regression** is used to figure out how dependent and independent variables are related. It's widely used to produce forecasts, such as for a company's sales revenue. Popular regression algorithms include linear regression, logistic regression, and polynomial regression. [55].

Supervised machine learning can be applied to many real world applications are Image- and Object-recognition, Speech recognition, Self-driving cars Predictive analysis, Customer sentiment analysis, Email Spam and Malware detection, Text categorization, Face detection, Signature recognition, Customer discovery, Weather forecasting, House price predictions, Stock market predictions, Online fraudulent transaction predictions.

Why supervised Learning?

Supervised learning is a simpler method while unsupervised learning is a complex method. The main advantage of supervised learning is that it allows you to collect data or produce a data output from the previous experience. The following are the attributes of supervised learning that made it the best fit for this problem.

- We will have an exact idea about the classes in the training data.
- Supervised learning is a simple process to understand since we know the relationship between the input and output would look like. In the case of unsupervised learning, we don't easily understand what is happening inside the machine, how it is learning, etc.
- We can find out exactly how many classes are there before giving the data for training.
- It is possible for us to be very specific about the definition of the classes, that is, we can train the classifier in a way that has a perfect decision boundary to distinguish different classes accurately.
- After the entire training is completed, we don't necessarily need to keep the training data in your memory. Instead, we can keep the decision boundary as a mathematical formula.
- Another typical task of supervised machine learning is to predict a numerical target value from some given data and labels.

Unsupervised machine learning

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. Its ability to discover similarities and differences in information make it the ideal solution for exploratory data analysis, cross-selling strategies, customer segmentation, image and pattern recognition. It's also used to reduce the number of features in a model through the process of dimensionality reduction; principal component analysis (PCA) and singular value decomposition (SVD) are two common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, probabilistic clustering methods, and more.

Semi-supervised learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of having not enough labeled data (or not being able to afford to label enough data) to train a supervised learning algorithm.

Unsupervised vs. supervised vs. semi-supervised learning

Unsupervised machine learning and supervised machine learning are frequently discussed together. Unlike supervised learning, unsupervised learning uses unlabeled data. From that data, it discovers patterns that help solve for clustering or association problems. This is particularly useful when subject matter experts are unsure of common properties within a data set. Common clustering algorithms are hierarchical, k-means, and Gaussian mixture models.

Semi-supervised learning occurs when only part of the given input data has been labeled. Unsupervised and semi-supervised learning can be more appealing alternatives as it can be time-consuming and costly to rely on domain expertise to label data appropriately for supervised learning. For this thesis Supervised regressions learning used of labeled datasets to train algorithms that to classify data or predict outcomes accurately.

3.5 Machine Learning Algorithms

Neural networks

Learning algorithm, a method used to implement learning process. Before use NN the weight connecting to neurons and bias values should be adjust. So, that out of the network will match the chosen pattern for specific set of output. The way used for adjusting these weight and parameter to provide such matches are usually referred to as learning algorithms. Learning method also required during training phase, weight can be modified in response to input or output changes.

Primarily leveraged for deep learning algorithms, neural networks process training data by mimicking the interconnectivity of the human brain through layers of nodes. Each node is made up of inputs, weights, a bias (or threshold), and an output. If that output value exceeds a given threshold, it "fires" or activates the node, passing data to the next layer in the network. Neural networks learn this mapping function through supervised learning, adjusting based on the loss function through the process of gradient descent. When the cost function is at or near zero, we can be confident in the model's accuracy to yield the correct answer. **Adam optimization** is solver for the Neural Network algorithm that is computationally efficient, requires little memory, and is well suited for problems that are large in terms of data or parameters or both. Adam is a popular extension to stochastic gradient descent.

Why ADAM Optimizer?

This algorithm has risen to prominence in recent years, particularly in the deep learning field. There are many reasons why Adam optimization algorithm performs is used in this paper. Many of these reasons are listed below

- Straightforward to implement.
- Computationally efficient.
- Little memory requirements.
- Invariant to diagonal rescale of the gradients.
- Well suited for problems that are large in terms of data and/or parameters.
- Appropriate for non-stationary objectives.
- Appropriate for problems with very noisy/or sparse gradients.

Hyper-parameters have intuitive interpretation and typically require little tuning.

Naive Bayes

The Naive Bayes Theorem's premise of class conditional independence is used in the Naive Bayes classification technique. This means that the existence of one feature in the probability of a certain event has no bearing on the presence of another, and each predictor has an equal impact on the outcome. Multinomial Nave Bayes, Bernoulli Nave Bayes, and Gaussian Nave Bayes are the three types of Nave Bayes classifiers. Text classification, spam detection, and recommendation systems are all applications of this technology.

Linear regression

Linear regression is a statistical technique for determining the relationship between a dependent variable and one or more independent variables, and it is commonly used to forecast future results. Simple linear regression is used when there is only one independent variable and one dependent variable. Multiple linear regression [53] is used when the number of independent variables rises. It aims to plot a line of greatest fit, which is derived using the least squares method, for each type of linear regression. When shown on a graph, however, this line is straight, unlike other regression models.

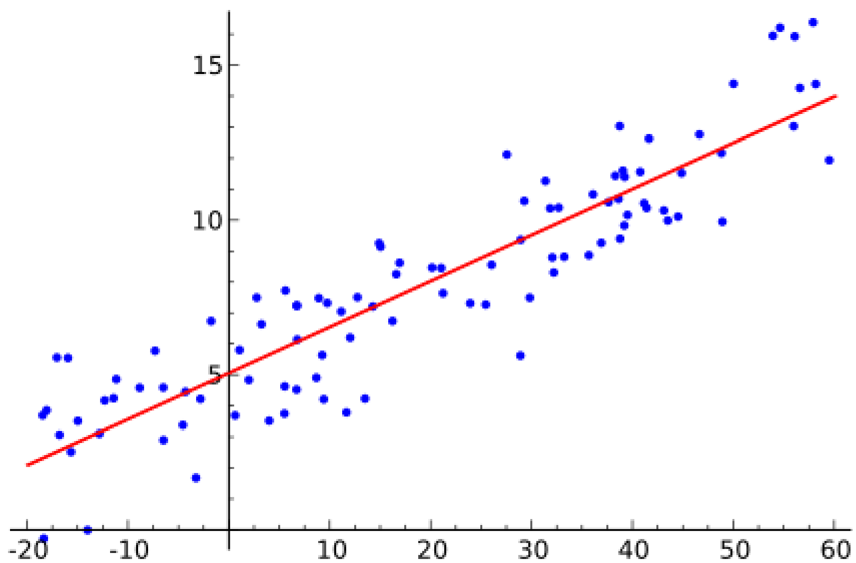

Figure 3. 3 Linear Regression

Fig3.3 shows that linear regression is a method of determining the relationship between two variables. It is assumed that the two variables have a direct link and that this relationship can be represented by a straight line.

Logistic regression

When the dependent variables are continuous, linear regression is used; however, when the dependent variables are categorical, such as "true" and "false" or "yes" and "no," logistical regression is used. While both regression models aim to understand relationships, they do so in different ways. between data inputs, logistic regression is mainly used to solve binary classification problems, such as spam identification.

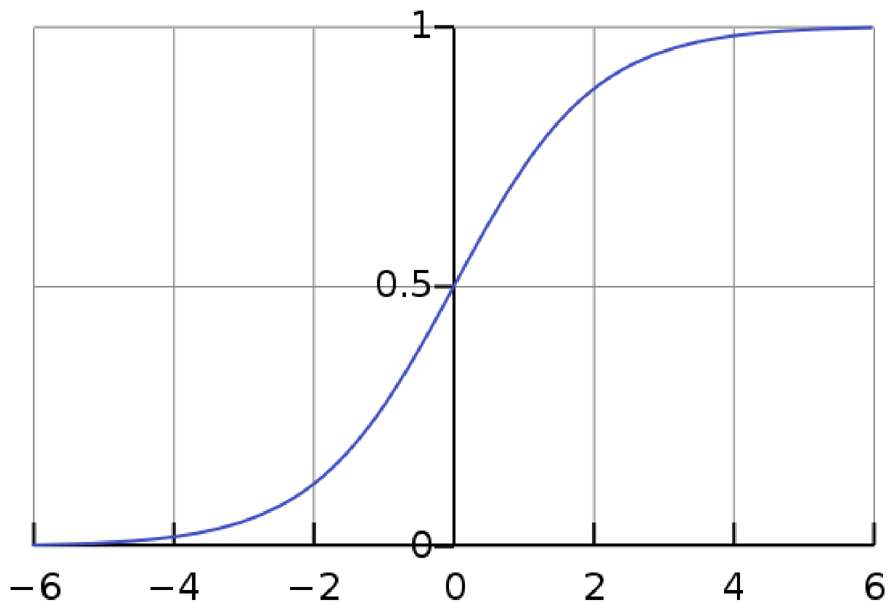

Figure 3. 4 Logistic Regression

Fig 3.4 shows Logistic regression, like linear regression, uses an equation as its representation. To forecast an output value, input values (x) are blended linearly using weights or coefficient values (referred to as the Greek capital letter Beta) (y)

Support vector machine (SVM)

A support vector machine (SVM) is a common supervised learning model created by Vladimir Vapnik that may be used for data classification as well as regression. However, it is most commonly used to solve classification problems by creating a hyper plane in which the distance between two classes of data points is at its greatest. The decision boundary is a hyper plane that separates the classes of data points on either side of the plane (e.g., oranges vs. apples).

K-nearest neighbor

The KNN algorithm, also known as the K-nearest neighbor algorithm, is a non-parametric algorithm that classifies data points based on their proximity and correlation with other data. This technique assumes that data points that are comparable can be discovered close together. As a result, it attempts to determine the distance between data points, which is commonly done using Euclidean distance, and then assigns a category based on the most common category or average.

Data scientists prefer it because of its ease of use and short computation time, but as the test dataset grows larger, the processing time increases, making it less appealing for classification jobs. KNN is commonly used in picture recognition and recommendation algorithms.

Random forest

Random forest is a supervised machine learning technique that can be used for classification and regression. The "forest" refers to a group of uncorrelated decision trees that are then blended to reduce variation and produce more accurate data predictions.

Table 3. 7 Machine Learning Algorithm

Machine Learning			
Supervised Learning		**Unsupervised Learning**	**Reinforcement Learning**
Classification	**Regression**	**Clustering**	Q-Learning
Naive Bayes	Generalized Linear Models	K-means, Fuzzy means	Policy gradients
Support Vector Machines	Logistic Regression	Gaussian mixture	Trust region policy optimization
K-Nearest Neighborhood	Support Vector regression	Hidden Markov model	Proximal policy optimization
Decision tress, Random Forest	Ensemble methods	Spectral Clustering	Hindsight experience replay
Neural Network	Neural Network	Neural Network	Deep Q neural Network

3.6 Development Environment and Tools

Jupyter Notebook

Jupyter Notebook is an open-source web software that lets you create and share documents with live code, equations, visualizations, and narrative text. Data cleansing and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and many other applications are all possible with it. It's a web-based interactive computational environment that lets you create Jupyter notebook papers. Jupyter Book is an open-source project that allows users to create books and papers out of computational data.

This environment had been chosen to work on during the course of this research due to the following reasons of suitability:

- Very easy to host server side, which is useful for security purposes. A lot of data is sensitive and should be protected, and one of the steps toward that is no data is stored on local machines. A server-side Jupyter Notebook setup gives you that for free.
- They're great for showcasing your work. You can see both the code and the results.
- You can run cell by cell to better get an understanding of what the code does. When prototyping, the cell-based approach of Jupyter notebooks is great. But you quickly end up programming several steps - instead of looking at object-oriented programming.

Machine Learning Libraries

A, Pandas

Pandas is a data manipulation and analysis software library for the Python programming language. It includes data structures and methods for manipulating numerical tables and time series, in particular. It's open-source software with a three-clause BSD license. The word "panel data" is an econometrics term for data sets that comprise observations for the same persons over multiple time periods. Pandas is mostly used to analyze data. Pandas can import data from a variety of file types, including comma-separated values (CSV), JSON, SQL database tables or queries, and Microsoft Excel. Pandas supports a wide range of data manipulation operations, including merging, reshaping, and selecting, as well as data cleaning and wrangling.

The following are some of the advantages offered while using pandas:

Data representation

Pandas provide extremely streamlined forms of data representation. This helps to analyze and understand data better. Simpler data representation facilitates better results for data science projects.

Less writing and more work done

It is one of the best advantages of Pandas. What would have taken multiple lines in Python without any support libraries, can simply be achieved through 1-2 lines with the use of Pandas.

Thus, using Pandas helps to shorten the procedure of handling data. With the time saved, we can focus more on data analysis algorithms.

An extensive set of features

Pandas are really powerful. They provide you with a huge set of important commands and features which are used to easily analyze your data. We can use Pandas to perform various tasks like filtering your data according to certain conditions, or segmenting and segregating the data according to preference, etc.

Efficiently handles large data

Wes McKinney, the creator of Pandas, made the python library to mainly handle large datasets efficiently. Pandas help to save a lot of time by importing large amounts of data very fast.

Makes data flexible and customizable

Pandas provide a huge feature set to apply on the data you have so that you can customize, edit and pivot it according to your own will and desire. This helps to bring the most out of your data.

B, NumPy

NumPy library is an important foundational tool for studying Machine Learning. Many of its functions are very useful for performing any mathematical or scientific calculation. As it is known that mathematics is the foundation of machine learning, most of the mathematical tasks can be performed using NumPy. NumPy stands for 'Numerical Python'. It is an open-source Python library used to perform various mathematical and scientific tasks. It contains multi-dimensional arrays and matrices, along with many high-level mathematical functions that operate on these arrays and matrices.

The merits of using Numpy for machine learning project includes the following:

- Consumes less memory.
- Fast as compared to the python List.
- Convenient to use.
- Element wise operation is possible

- Numpy array has the various function, methods, and variables, to ease our task of matrix computation.
- Elements of an array are stored contiguously in memory.

C, Tensor Flow

Tensor flow is an end-to-end open-source platform toolkit for numerical computing and large-scale machine learning that makes gathering data, training models, serving predictions, and improving future outcomes easier with Google Brain Tensor Flow. It has a large, flexible ecosystem of tools, libraries, and community resources that allow academics to advance the state-of-the-art in machine learning and developers to quickly construct and deploy machine learning applications. Tensor flow combines Machine Learning and Deep Learning models and algorithms into a single package. It makes use of Python as a user-friendly front-end and runs it in optimized C++.

Developers can use Tensor Flow to design a graph of computations to run. Each connection in the network represents data, whereas each node represents a mathematical action. As a result, rather than worrying about minor issues like how to connect the output of one function to the input of another, the developer may concentrate on the application's overall logic.

The following are the aspirations behind choosing Tensor flow library for this research:

- Open-source platform

It is an open-source platform that makes it available to all the users around and ready for the development of any system on it.

- Data visualization

Tensor Flow provides a better way of visualizing data with its graphical approach. It also allows easy debugging of nodes with the help of Tensor Board. This reduces the effort of visiting the whole code and effectively resolves the neural network.

- Keras friendly

Tensor Flow has compatibility with Keras, which allows its users to code some high-level functionality sections in it. Keras provides system-specific functionality to Tensor Flow, such as pipelining, estimators, and eager execution.

The Keras functional API supports a variety of topologies with different combinations of inputs, output, and layers.

- Scalable

Almost every operation can be performed using this platform. With its characteristic of being deployed on every machine and graphical representation of a model allows its users to develop any kind of system using Tensor Flow.

Hence Tensor Flow has been able to develop systems like Airbnb, Dropbox, Intel, Snapchat, etc.

- Compatible

It is compatible with many languages such as C++, JavaScript, Python, C#, Ruby, and Swift. This allows a user to work in an environment they are comfortable in.

- Parallelism

Tensor Flow finds its use as a hardware acceleration library due to the parallelism of work models. It uses different distribution strategies in GPU and CPU systems.

A user can choose to run its code on either of the architecture based on the modeling rule. A system chooses a GPU if not specified. This process reduces the memory allocation to an extent.

- Architectural support

Tensor Flow also has its architecture TPU, which performs computations faster than GPU and CPU. Models built using TPU can be easily deployed on a cloud at a cheaper rate and executed at a faster rate.

- Graphical support

Deep learning uses Tensor Flow for its development as it allows building neural networks with the help of graphs that represent operations as nodes.

Tensor Flow acts in multiple domains such as image recognition, voice detection, motion detection, time series, etc. hence it suits the requirement of a user.

D, KeraS

Keras is a deep learning API written in Python, running on top of the machine learning platform Tensor Flow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast as possible is key to doing good research.

Keras is:

Simple: - but not simplistic. Keras reduces developer cognitive load to free you to focus on the parts of the problem that really matter.

Flexible: - Keras adopts the principle of progressive disclosure of complexity: simple workflows should be quick and easy, while arbitrarily advanced workflows should be possible via a clear path that builds upon what you've already learned.

Powerful: -Keras provides industry-strength performance and scalability

E, Matplotlib

Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy. As such, it offers a viable open source alternative to MATLAB. Developers can also use matplotlib's APIs (Application Programming Interfaces) to embed plots in GUI applications.

A Python matplotlib script is structured so that a few lines of code are all that is required in most instances to generate a visual data plot. The matplotlib scripting layer overlays two APIs:

- The pyplot API is a hierarchy of Python code objects topped by matplotlib pyplot

- An OO (Object-Oriented) API collection of objects that can be assembled with greater flexibility than pyplot. This API provides direct access to Matplotlib's backend layers.

F, Scikit-learn

Scikit-learn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, and clustering and dimensionality reduction.
Scikit-learn comes loaded with a lot of features. Here are a few of them to help you understand the spread:

- **Supervised learning algorithms:** Think of any supervised machine learning algorithm you might have heard about and there is a very high chance that it is part of scikit-learn. Starting from Generalized linear models (e.g. Linear Regression), Support Vector Machines (SVM), Decision Trees to Bayesian methods – all of them are part of scikit-learn toolbox. The spread of machine learning algorithms is one of the big reasons for the high usage of scikit-learn. I started using scikit to solve supervised learning problems and would recommend that to people new to scikit / machine learning as well.
- **Cross-validation**: There are various methods to check the accuracy of supervised models on unseen data using sklearn.
- **Unsupervised learning algorithms**: Again there is a large spread of machine learning algorithms in the offering – starting from clustering, factor analysis, and principal component analysis to unsupervised neural networks.
- **Feature extraction**: Scikit-learn for extracting features from images and text.

CHAPTER FOUR

ANALYSIS AND RESULTS

4.1 Model analysis

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

One of the early definitions of machine learning was put forward by Arthur Samuel defining machine learning as a Field of study that gives computers the ability to learn without being explicitly programmed. But these definition lacks some generalization and a more general definition of machine learning was stated by Tom Mitchel as A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at **tasks in T, as measured by P, improves with experience E.**

Machine learning is also an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classifications or predictions, uncovering key insights within data mining projects. These insights subsequently drive decision making within applications and businesses, ideally impacting key growth metrics. As big data continues to expand and grow, the market demand for data scientists will increase, requiring them to assist in the identification of the most relevant business questions and subsequently the data to answer them.

A multilayer perceptron (MLP) is a class of feed forward artificial neural network (ANN). The term MLP is used ambiguously, sometimes loosely to mean any feed forward ANN, sometimes strictly to refer to networks composed of multiple layers of perceptron (with threshold activation). Multilayer perceptron are sometimes colloquially referred to as "vanilla" neural networks, especially when they have a single hidden layer.

An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. **Except for the input nodes, each node is a neuron that uses a**

nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training. Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. It can distinguish data that is not linearly separable.

The basic components of the developed perceptron include Inputs, Weights and Biases, Linear combination, and Activation function. Following is the basic terminology of each of the components.

1. Inputs of a perceptron are real values input.
2. Weights are parameters within the neural network to transform input data.
3. Bias is an additional parameter used to adjust output along with a weighted sum.
4. Linear combination is the merging of input values.
5. Activation values are non-linear transformations of input for specific outputs.

The **input** of the network was a comma separated feature matrix consisting of 15 features that are supposed to bring about mobile network congestion these are cell category, cell availability, transceiver availability, call setup success rate(CSSR), standalone dedicated channel congestion, Traffic channel drop rate, Call drop rate, Handover success rate, Standalone dedicated channel resource utilization, Traffic channel congestion utilization, CS traffic, PS traffic, Transceiver out, Transceiver availability and standalone dedicated channel availability.

The **weights** of the network are to be adjusted by the network since we are in the supervised learning emblem where the network learns from a labeled data and compares its prediction with the already known output of the system to compute an error value which will them be propagated backwards to adjust weights using the back propagation algorithm such that it could minimize that error on the next prediction.

This research makes use of two MLPs that have different number of layers with fixed amount of layers in each of the models.

4.2 MLP having 10 layers (MODEL1)

In our design and modeling part the first model was MODEL1 has an input layer dimension equal to the number of features 15 and an output or target of one node for predicting the mobile network congestion level. This model was internally made up of 10 hidden layers each comprised of 200 neurons for learning whatever the information or feature is passed through its predecessor layers and neurons. The neurons in the input and hidden layers were having a ReLU activation function.

In order to use stochastic gradient descent with back propagation of errors to train deep neural networks, an activation function is needed that looks and acts like a linear function, but is, in fact, a nonlinear function allowing complex relationships in the data to be learned.
The function must also provide more sensitivity to the activation sum input and avoid easy saturation.

A node or unit that implements this activation function is referred to as a rectified linear activation unit**,** or ReLU for short. Often, networks that use the rectifier function for the hidden layers are referred to as rectified networks. Adoption of ReLU may easily be considered one of the few milestones in the deep learning revolution, e.g. the techniques that now permit the routine development of very deep neural networks.

On the other hand since the model is required for fitting (regression) scheme the output layer does not have any activation function. The model MODEL1 is shown in the figure below

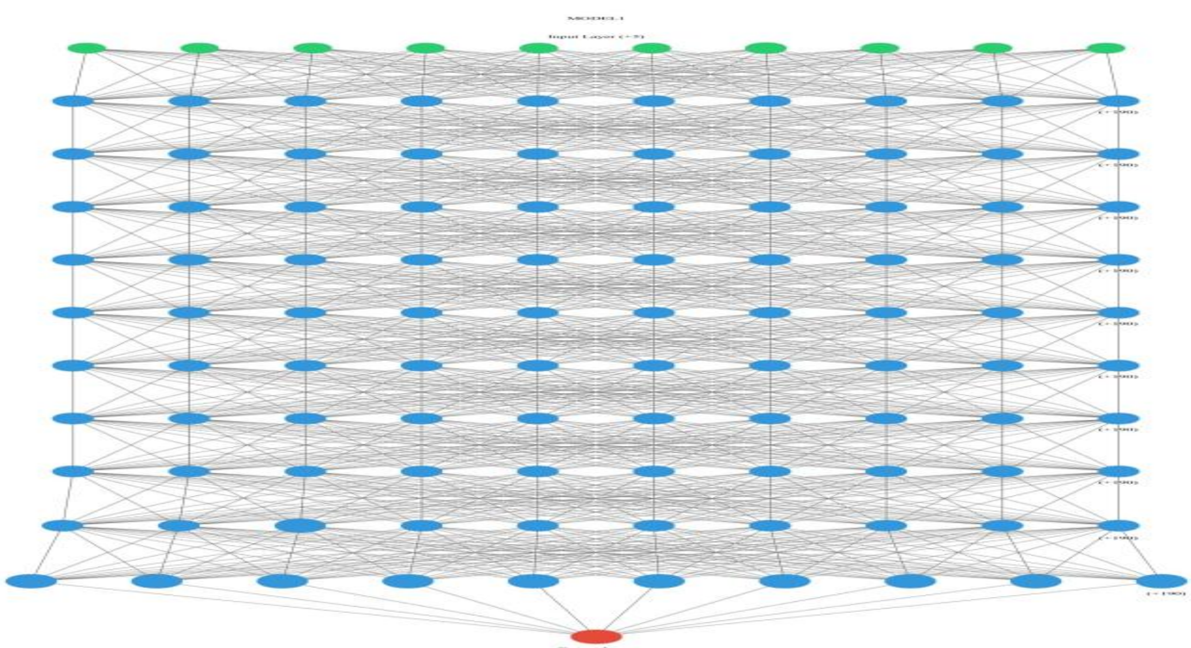

Figure 4. 1 MLP having 10 layers (MODEL1)

As shown in the Figure 4.1 the proposed multilayer neural network has fifteen input nodes where it can accept for the data input features from the dataset. The MLP visualization tool allows as to create a view of how our machine learning model would look like, having this in mind the figure tells that there are five more neurons the are note shown since it only shows ten input neurons. Each of the blue neuron rows refer to a hidden layer where each layer consists of two hundred neurons employing Relu activation function but the MLP visualizer tool in Jupiter notebook is able to view only the number of neurons that can be on the current display screen resolution. But the visualizer also shows that there are one hundred nineteen additional neurons on each layer. The proposed MLP model shows there are ten hidden layers. Finally, using the multilayer neural network for prediction purpose of a single variable a single output layer with no activation function is used to predict mobile network congestion level. The output is therefore a weighted sum of the outputs of the last hidden layer since the model is used for prediction purpose.

4.3 MLP having 15 layers (MODEL2)

In our design and modeling part the second model was MODEL2 has an input layer dimension equal to the number of features 15 and an output or target of one node for predicting the mobile network congestion level. This model was internally made up of 15 hidden layers each comprised of 200 neurons for learning whatever the information or feature is passed through its predecessor layers and neurons. The neurons in the input and hidden layers were having a ReLu activation function.

Traditionally, some prevalent non-linear activation functions, like sigmoid functions (or logistic) and hyperbolic tangent, are used in neural networks to get activation values corresponding to each neuron. Recently, the ReLu function has been used instead to calculate the activation values in traditional neural network or deep neural network paradigms.

On the other hand, since the model is required for fitting (regression) scheme the output layer does not have any activation function. The model MODEL2 is shown in the figure below.

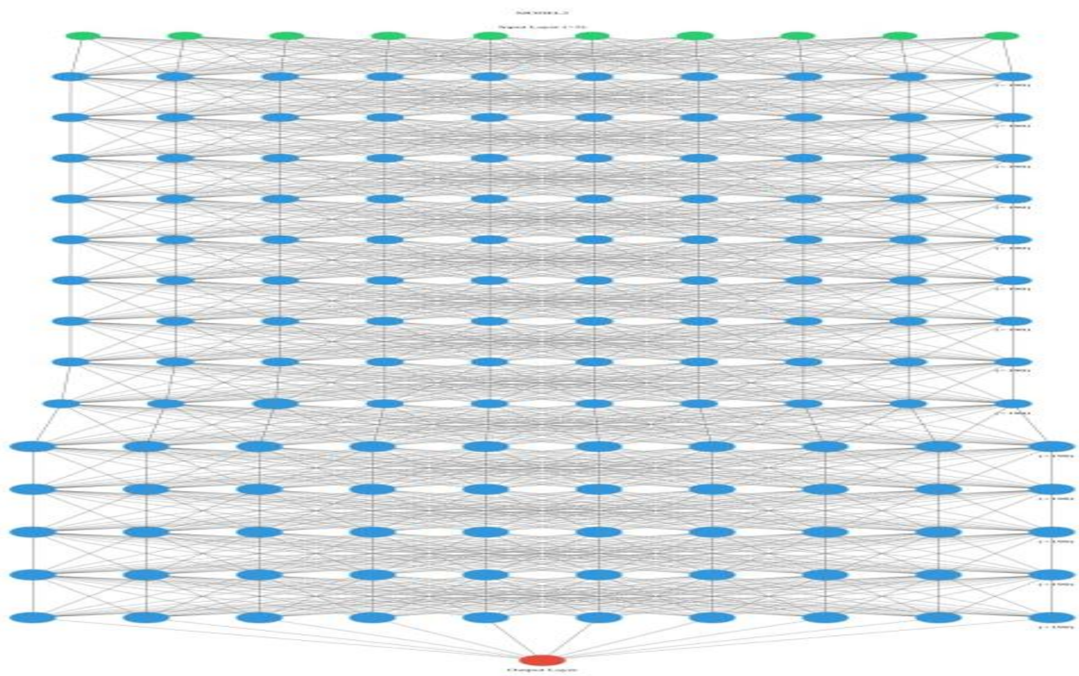

Figure 4. 2 MLP having 15 layers (MODEL2)

As shown in the Figure 4.2 the proposed multilayer neural network is able to obtain fifteen learnable features from the training data which have fifteen predictor parameters. The ann-viz tool in Jupiter notebook allows to visualize what would the proposed multilayer perceptron would look like. Due the screen display area the visualization tool wasn't able to show all the available input neurons thus it puts a notice so that there is five more neuron not shown on the current screen.

The blue neuron rows below the input neurons show that there are fifteen hidden layers in our neural network. Each of these hidden layers has two hundred neurons but only ten of the two hundred neurons present for each layer. Such that a notice is provided showing there are one hundred and nineteen neurons present.

The red neuron denotes the output layer which is used to predict the mobile network congestion parameter. As deducted from Figure 4.2 the output is a weighted sum of the outputs of the last hidden layer since the model is used for prediction purpose.

4.4 Result Discussion

The project aimed to answer the following question: **What are the performance outcomes of a machine learning model when assisting in mobile traffic congestion prediction**? We found that we get an average loss value of 1.192 and mean absolute error of 0.345 for the three sites using a multilayer perceptron having 10 hidden layers and average loss value of 1.2781 and mean absolute error of 0.272 for the three sites using a multilayer perceptron having 15 hidden layers.

In particular, we set out to answer the following questions:

- **To what extent a machine learning technique can be used to predict a mobile traffic congestion in the training dataset**
- **To what extent a multilayer neural network is able to predict in the training dataset**
- **How many perceptron and layers can provide optimal error minimization while avoiding Over fitting of the model**

The questions arise from the nature of the data, the data processing phase, and the nature of the research variable and improvement to common methods of network congestion anticipation and handling algorithms.

The following were the visualized models using the Jupiter notebook model visualization tools

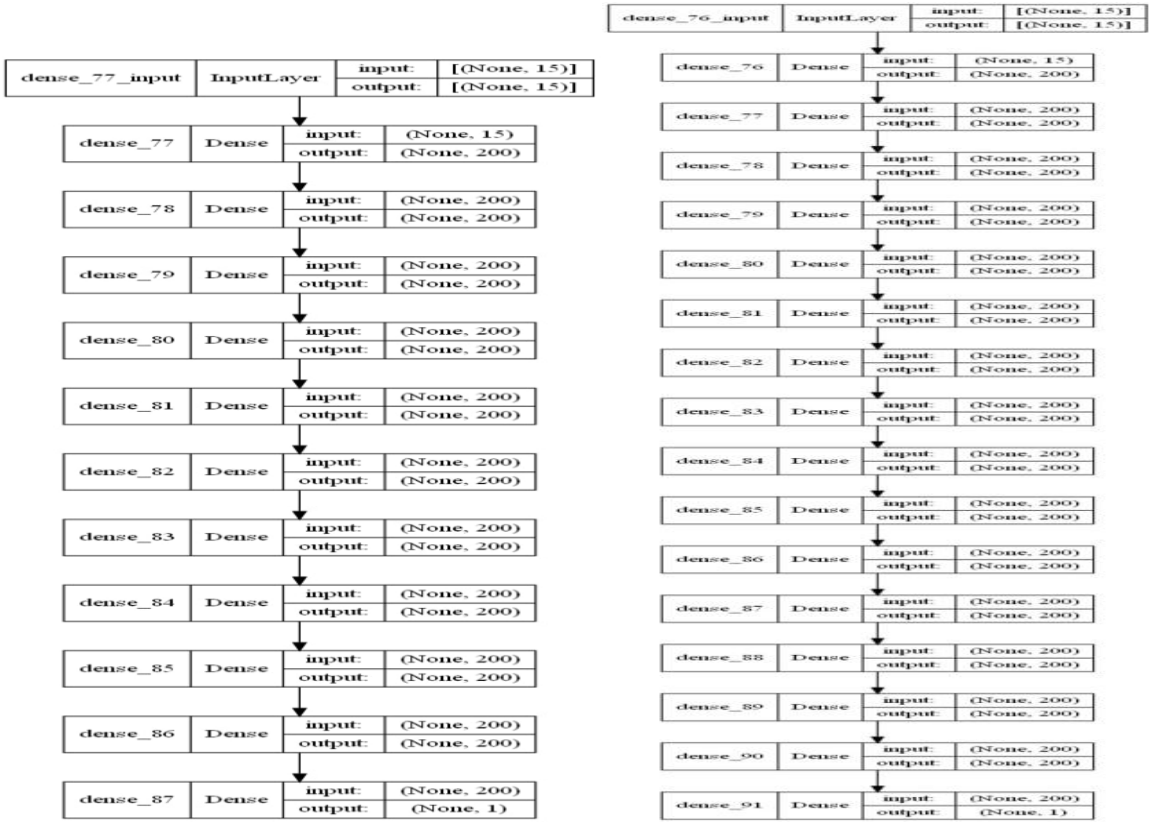

Figure4. 3 Visualization of Model1 and Model 2 Using Plot Model Function

To what extent a machine learning technique can be used to predict mobile traffic congestion in the training dataset

First the uses of machine learning techniques especially multilayer perceptron for prediction of mobile network congestion have shown a good performance over the selected sites of the dataset. As stated above this research employed two MLP models Model1 and Model2 where Model1 contains 10 hidden layers whereas Model2 comprised of 15 hidden layers.

Three train test datasets were used for the three sites. Each dataset was divided into two sets that are the train and test data which the train data is 70% of the total dataset and is used to train the model and the test data were 30% of the total dataset is used to test the performance of the trained model on the data. The training data in turn is divided into train and validation dataset which will be used to **validate** the training process in every epoch. The following figures show the performance measure of training phase of the two models for each of the three sites which uses mean squared error (MSE) as the plot property.

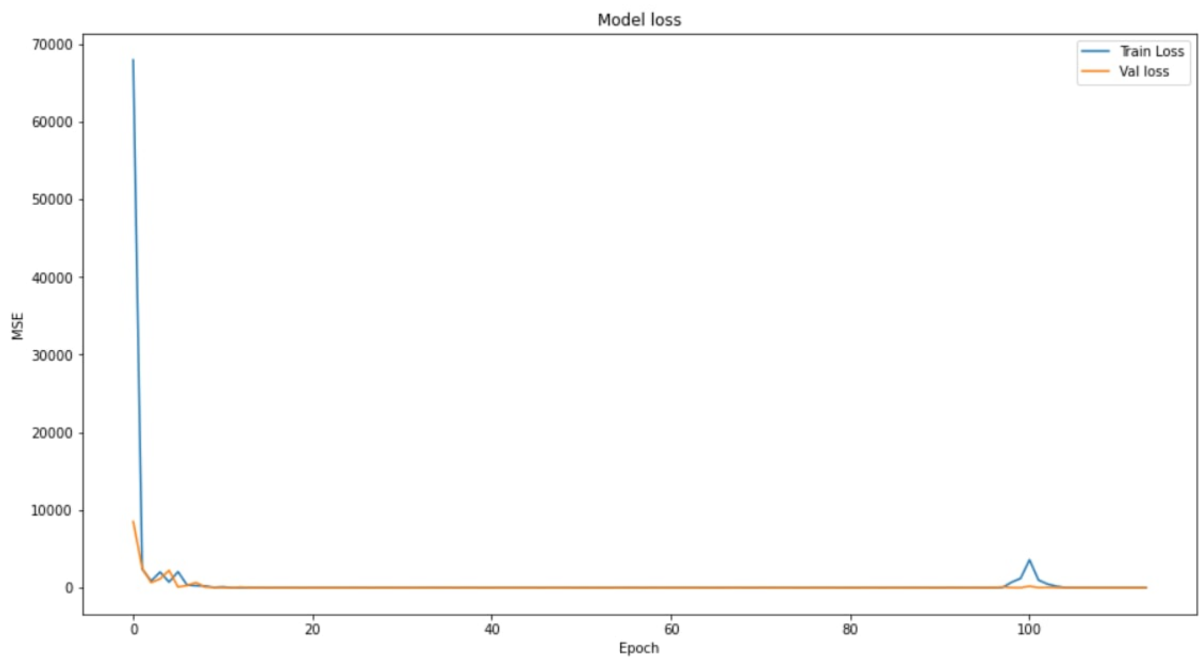

Figure 4. 4 MSE Vs. epoch for site 1 using Model1

Figure 4.4 shows the mean squared error (MSE) vs. epoch for site 1 using Model1. As denoted on the legend of the graph the blue line represents the train loss and the orange line represents the validation loss. The train loss was showing peak values for the first five epochs and decreases instantly afterwards. The validation loss starts at about a MSE of nine thousand which decreases in a few iterations after the start of the train-validate process. As this process goes on the graph is seen to keep a very low MSE around zero which tells that the model performed well both on the training and validation data.

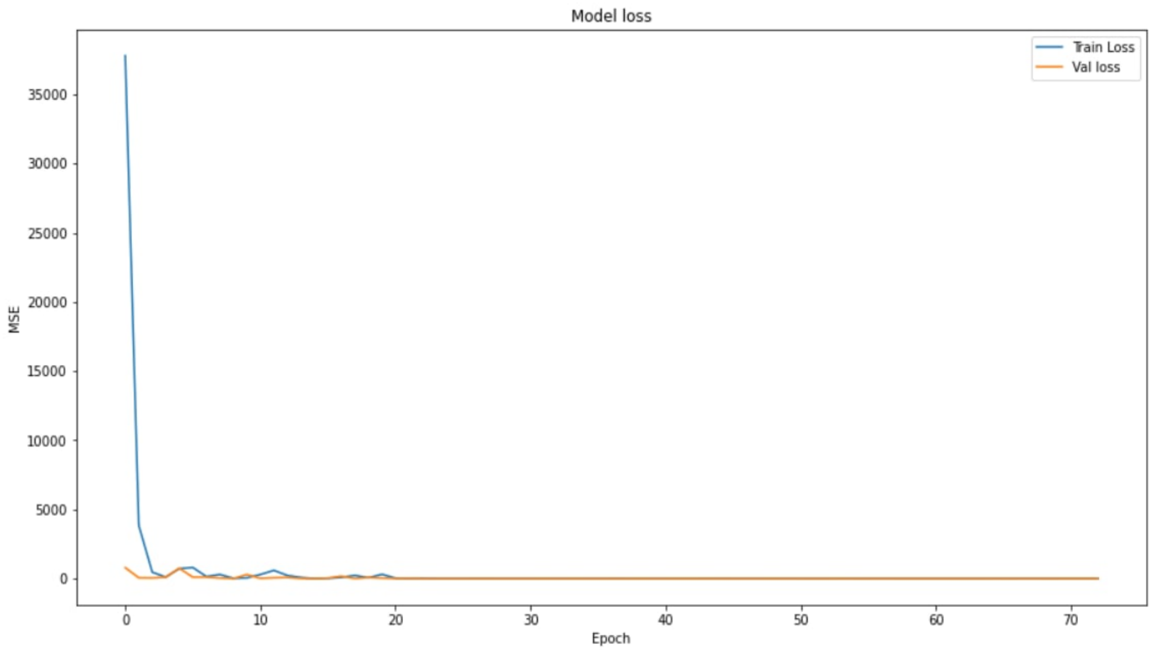

Figure 4. 5 MSE Vs. epoch for site 2 using Model1

As depicted from the graph in Figure 4.5 the mean squared error of the multilayer perceptron model Model1 decreases as the model advances on new epochs. In this figure the line in blue shows the train loss and the line in orange denotes the validation loss. Looking at the first ten epochs we can see that even in situations where the model training MSE values where on their peaks the validation loss is kept minimum which verifies that the model is trying to catch some solid relationships between the input features and the output feature so that being requested to predict on some combination of the validation data the model maintained minimum validation loss. Through the whole training and validation process the model finally approaches to keep some constant and very minimum MSE for both the training and validation data.

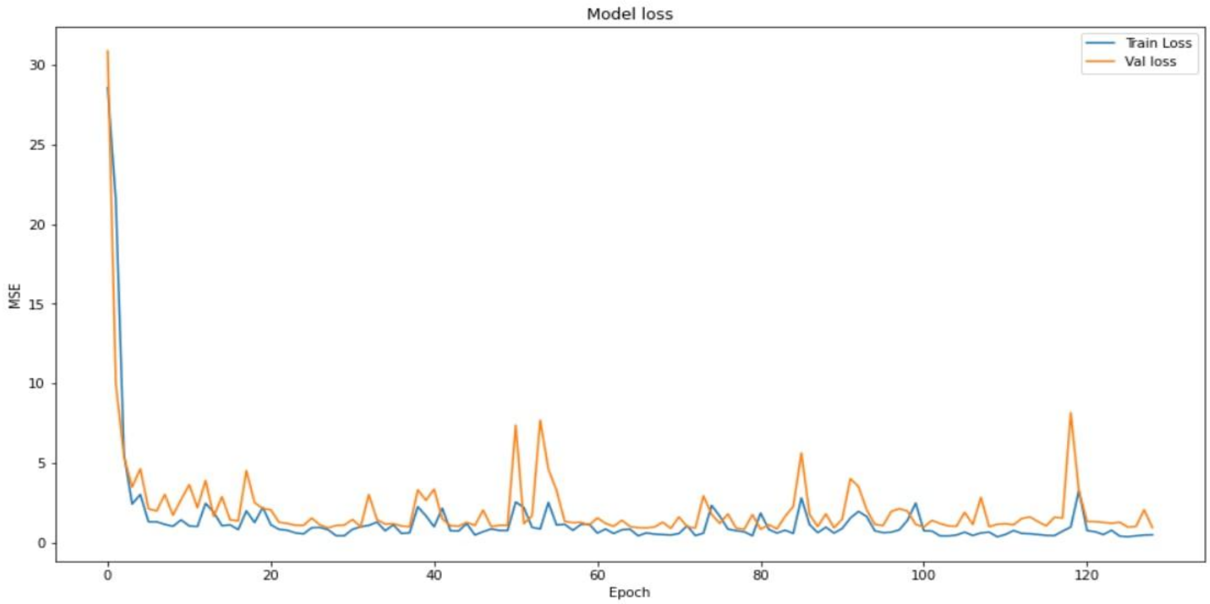

Figure 4. 6 MSE Vs. epoch for site 3 using Model1

The Figure 4.6 is trying to tell the training story of the machine learning model Model1 on data gathered from site 3 which lasted about a session of around 130 epochs. At the beginning of the training process the graph shows both the train and the validation losses were around 30 but as the model goes through the first 10 epochs both of these loss values fall below 5 which can be interpreted that the model is finding some base function that can rule the relationship between the input features and the output features. Around the last epochs of the training process it can be seen that the model approaches to a mean squared error value between zero and one.

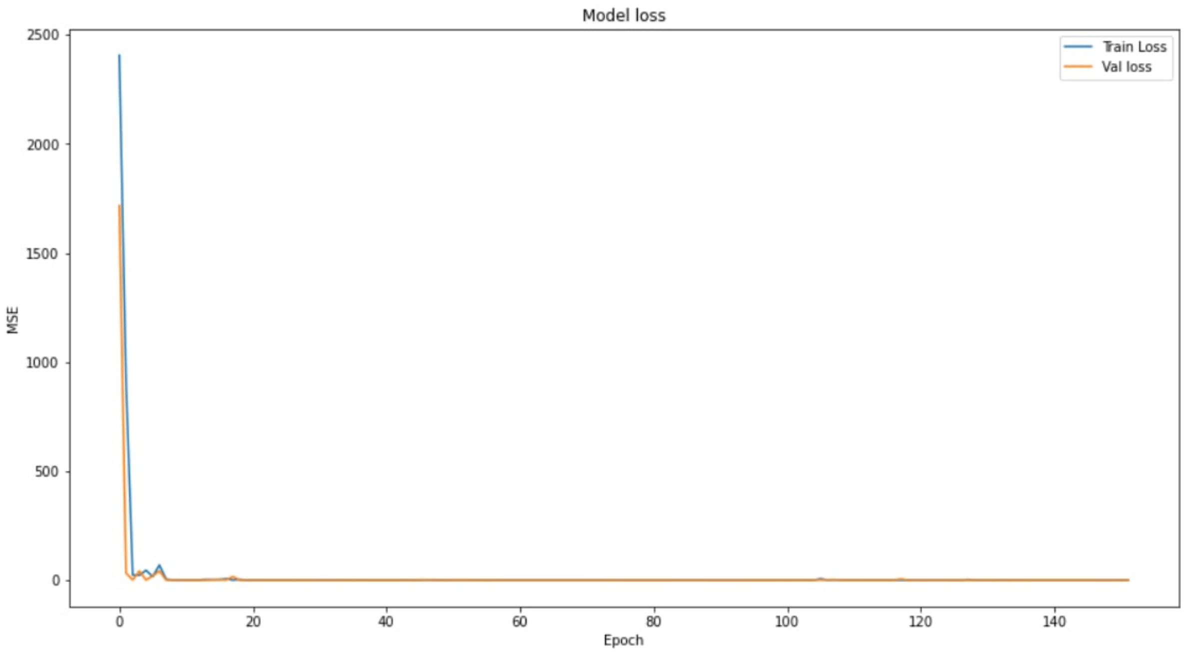

Figure 4. 7 MSE Vs. epoch for site 1 using Model2

Figure 4.7 shows the mean square error value of a machine learning model Model2 in around one hundred fifteen epochs needed for the model so that it can sufficiently adapt in to the network data provided for site 1. As that can be seen from the figure the MSE value of the Model at the first ten epoch was read peak but the value decreases as the model begin to build its experience on a larger and larger data as the epochs proceed. Thus, reaching at the final epochs the model is seen to maintain some constant MSE which means the model is having the same and minimum level of in accuracy for both the training and validation data.

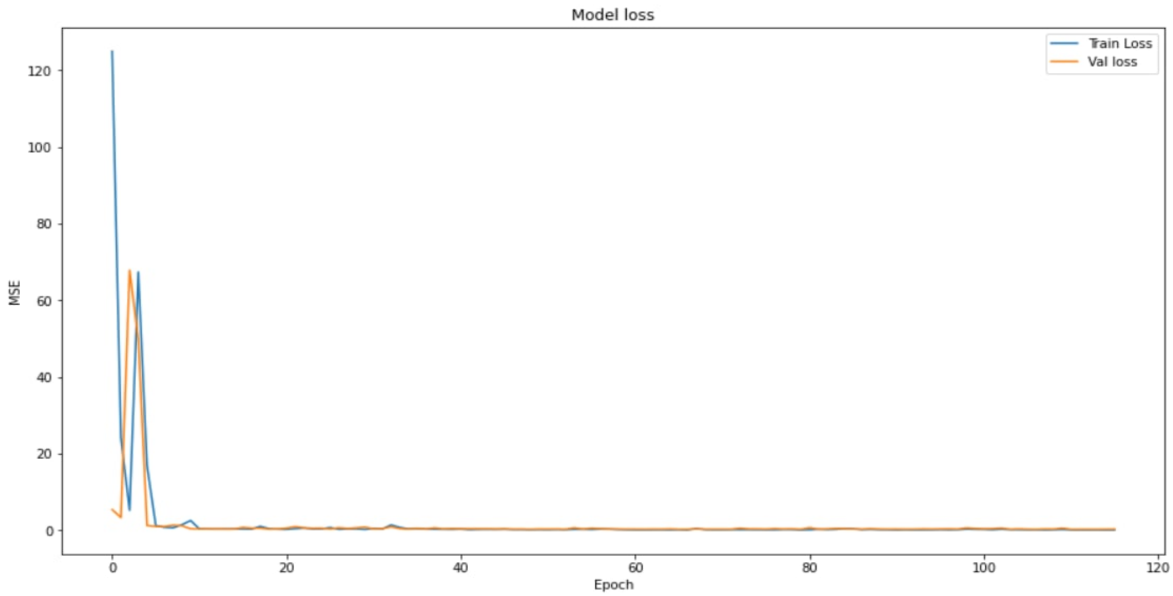

Figure 4. 8 MSE Vs. epoch for site 2 using Model2

The training session shown in Figure 4.8 shows how the multilayer perceptron Model2 responded to a training and validation data in blue and orange color respectively. The training loss is seen to start at some point around 120 which is seen to fluctuate during the early phases of the process in which this pattern is also seen to repeat it's self on the validation data as well. So that the model understands the generalization function from the earliest epochs needed some modification. Due to those modifications made on the model it successfully began to reduce the MSE as the session goes on.

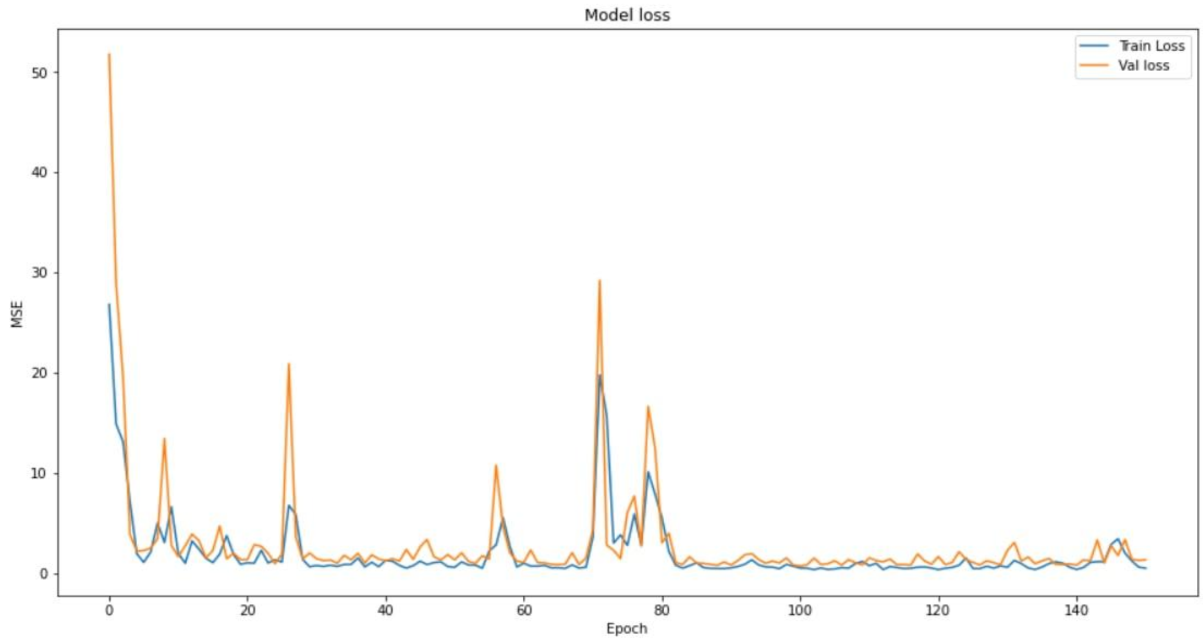

Figure 4. 9 MSE Vs. epoch for site 3 using Model2

Figure 4.9 shows the training session of a machine learning model Model2 on data gathered from site 3 for about one hundred fifty epochs. The graph in this figure can be seen to tell us how the training and validation loss started to fall to zero. In the beginning the model was observed to have higher MSE which shows the generalization function was relatively successful to generalize for the training data but failed to generalize effectively as that on the training data. Thus the learning algorithm can take huge penalties against the model so that it tells how far it is from the required rule. This way as the training progress on epoch the MSE starts to decrease and approach to some constant and minimum value

As that can be induced from the above six graphs the mean squared error of both machine learning models is reduced throughout the ongoing training progress. The average time taken by the six training sessions is 5.34 seconds this shows that we can successfully develop an MLP model for mobile network congestion prediction in shortest possible time.

The MSE results obtained in each of the six training sessions are satisfactory such that the model can progress on to the next testing phase where the model is supposed to perform on a labeled data that it has never been encountered before so that we can assess how it may respond to mobile network parameters collected from a real scenario.

The testing phases of the model development process will be shown using the figures taken from the Jupiter notebook testing environment. In the following figures the data labels (outputs) from the dataset are compared with the network prediction using graphs as shown below

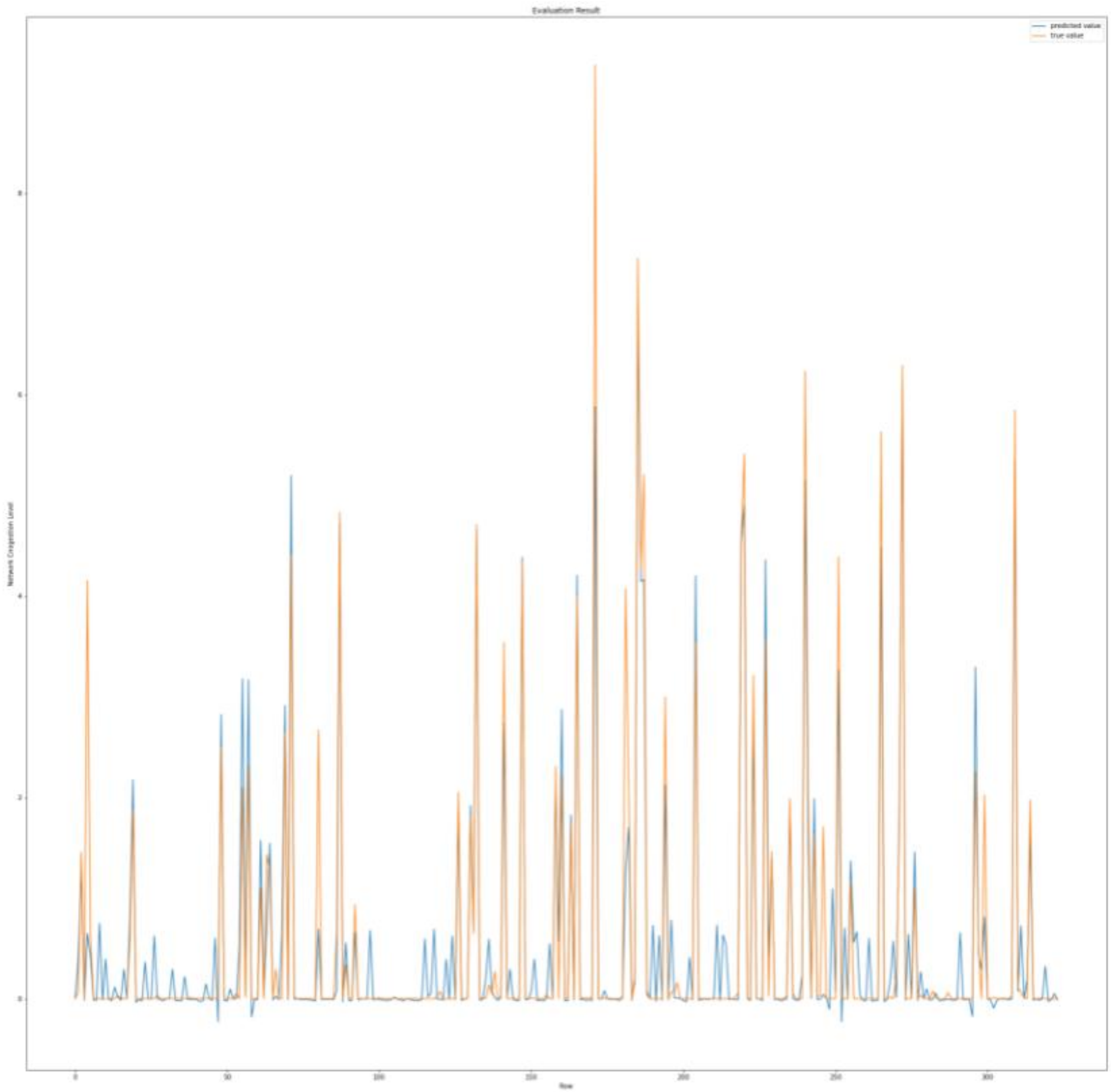

Figure 4. 10 Test results compared with the outputs of the dataset for site1 using Model1

Figure 4.10 shows how the model behaves on a test data. The graph shows the comparison between a real network output data and the output data obtained from the prediction model in orange line and in blue line respectively. As expected, the blue line nearly followed the orange line in a close proximity which tells us that we are getting results that mimic the real network data in orange.

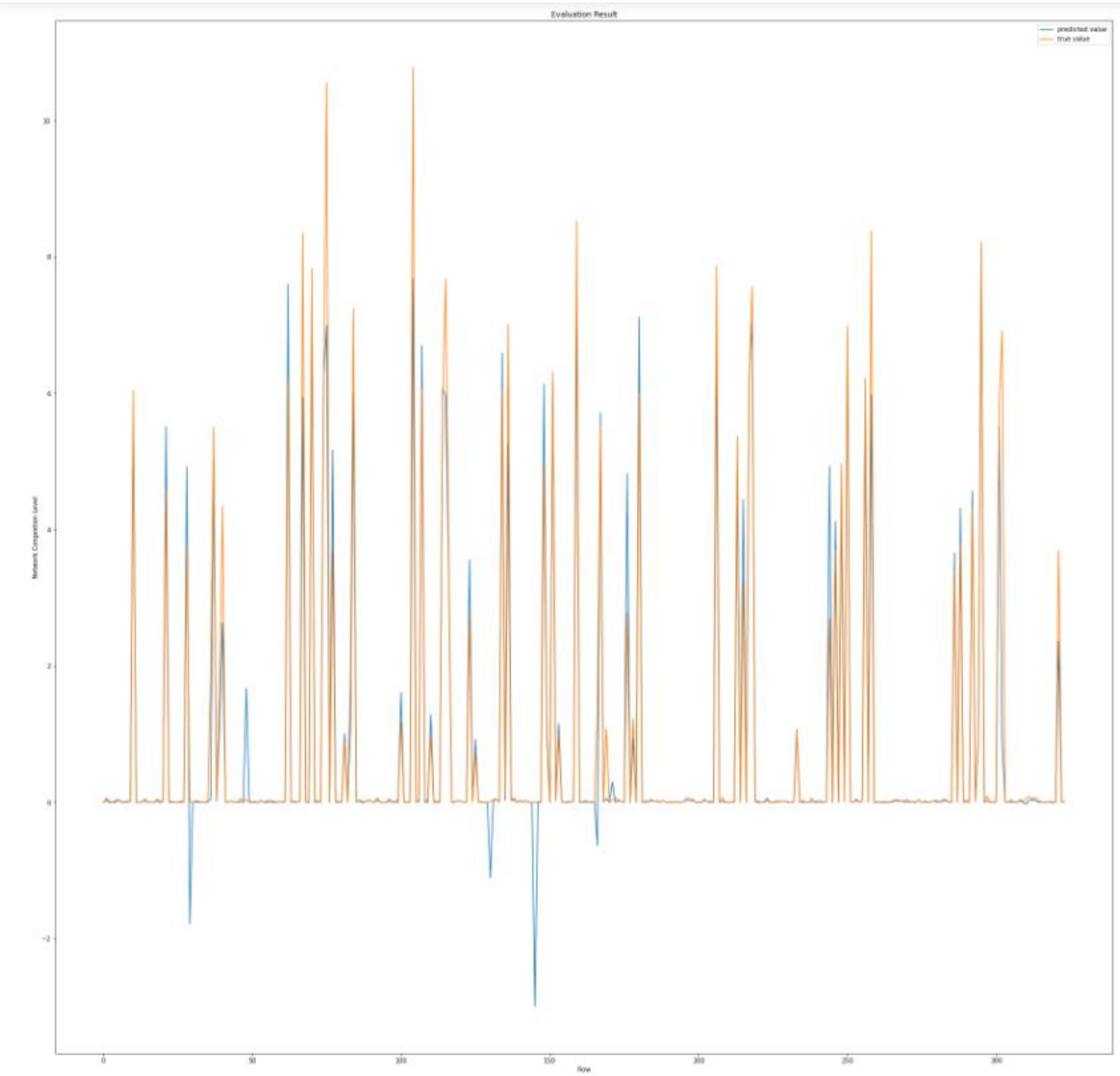

Figure 4. 11 Test results compared with the outputs of the dataset for site2 using Model1

Figure 4.11 uses blue color to represent how the proposed model predicts values from a combination of real network features that it hasn't been trained on and the orange line describes what the real network data would look like. It can be seen that more or less the model was able to copy the output feature pattern observed in the real network output data.

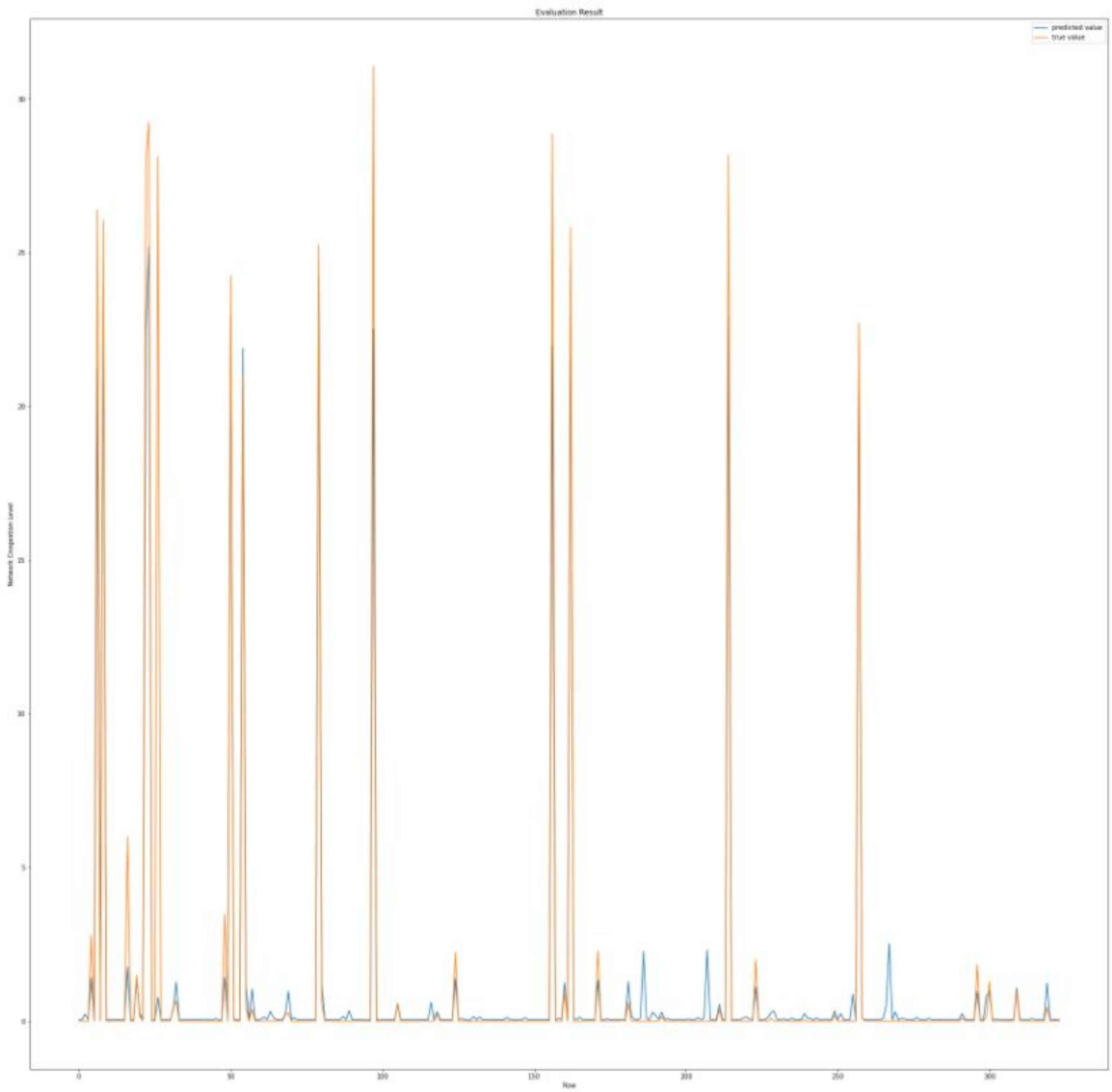

Figure 4. 12 Test results compared with the outputs of the dataset for site3 using Model1

The ability of a machine learning model Model1 to predict the level of network congestion for site3 was shown in Figure 4.12. The figure uses orange color to show what the real traces of the output data look like for the combinations of input features. The line which is observed to track the patterns of the real network output data is a blue line which is showing how well the model is following the expected output pattern.

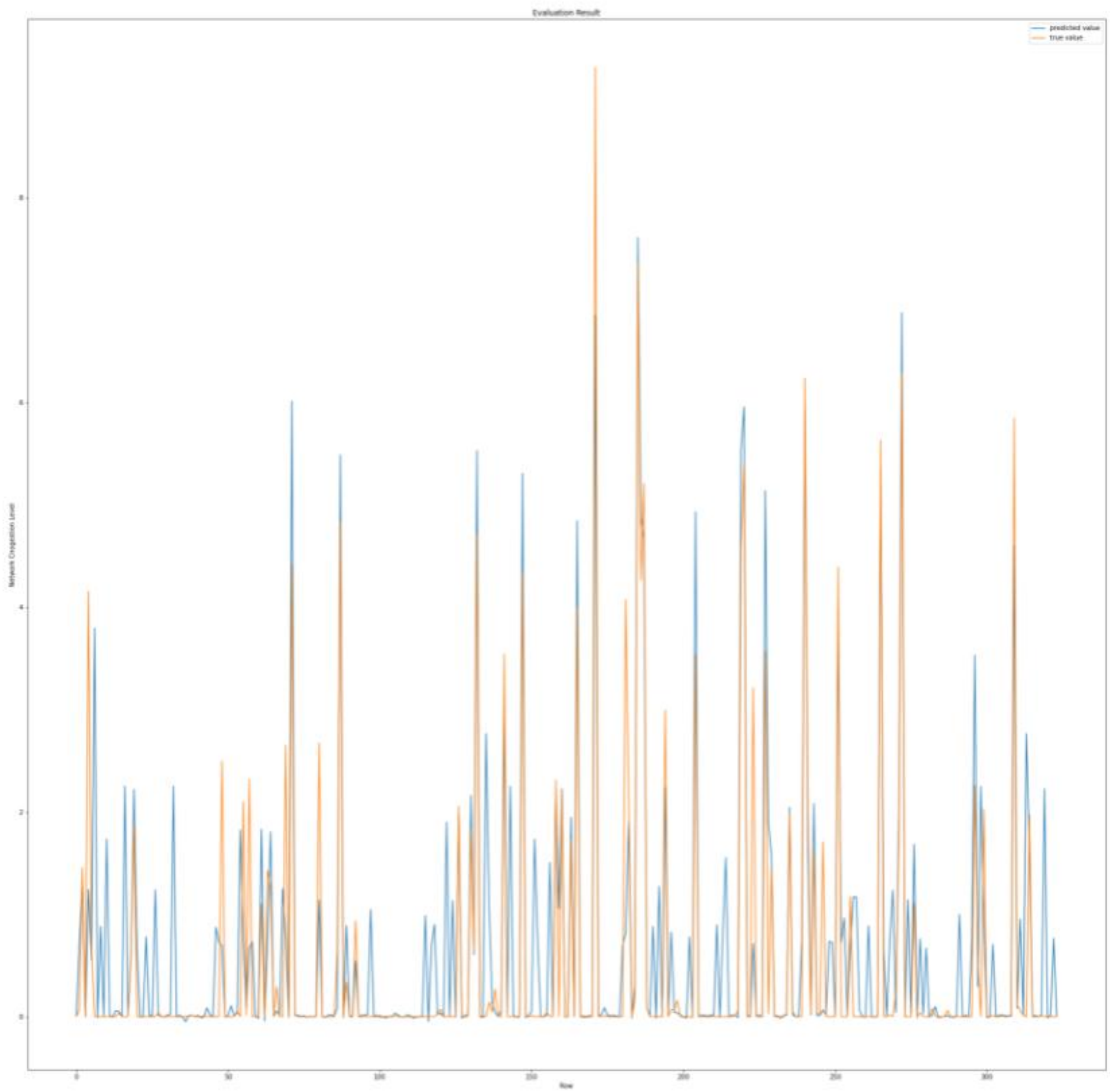

Figure 4. 13 Test results compared with the outputs of the dataset for site1 using Model2

Figure 4.13 shows how well the model Model2 is trying to mimic the expected network congestion pattern observed in site 1. Since our model is required to have hands on prediction on a new data the figure above shows the response of the model to the combination of input features in blue. While the orange line is the base or the bench mark of measurement the blue line is expected and seen to trace the orange line.

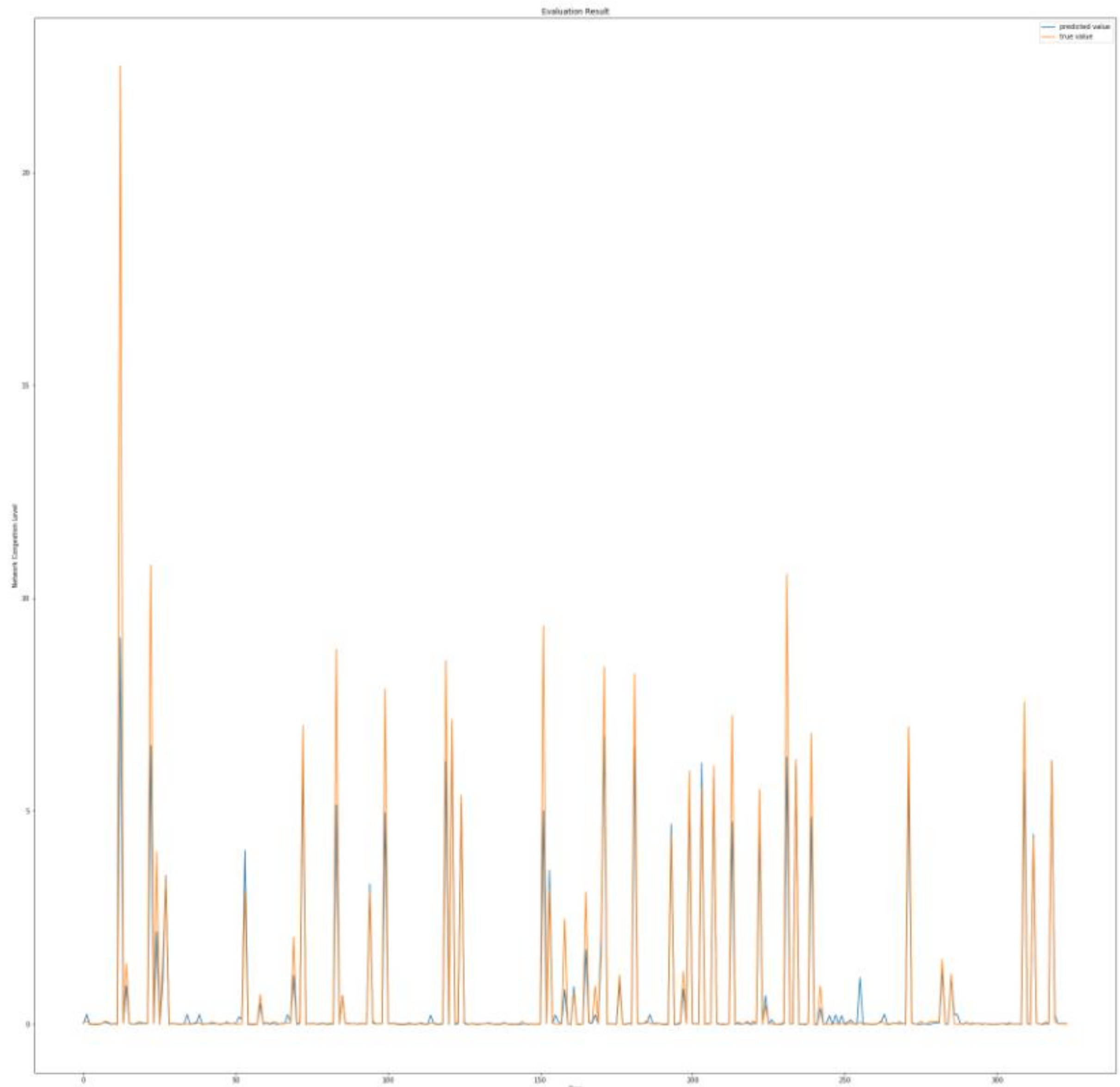

Figure 4. 14 Test results compared with the outputs of the dataset for site2 using Model2

The prediction capacity of the multilayer neural network model Model2 on a real test data from site 2 was shown in Figure 4.14. The blue line denotes the prediction results and the orange line shows real network congestion scenarios. **It has been shown how effectively the blue line was observed to follow the footsteps of the orange line.**

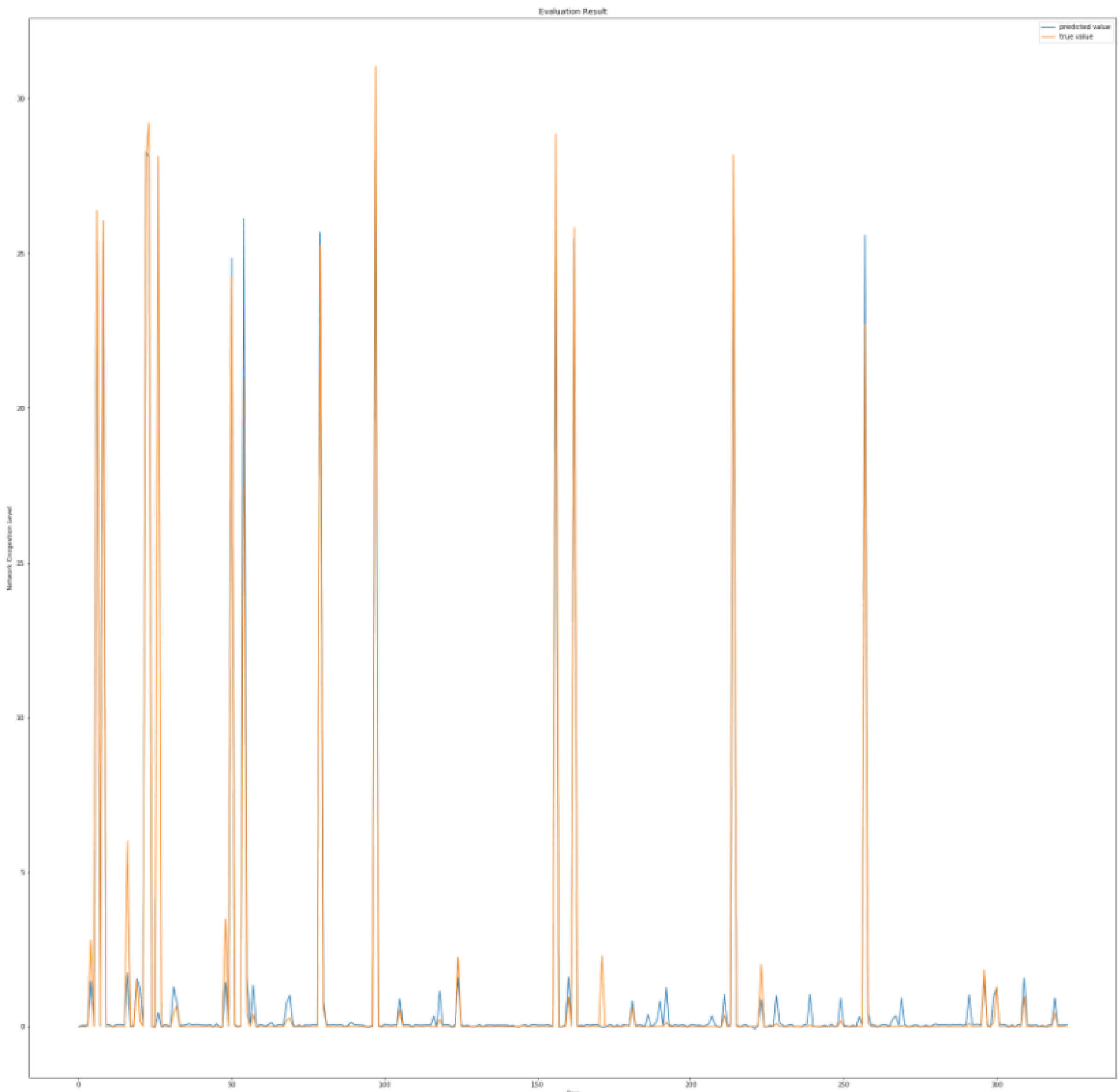

Figure 4. 15 Test results compared with the outputs of the dataset for site3 using Model2

As can be seen from the figure 4.15 above the model Model2 was able to produce prediction results shown in blue line. Whereas the orange line represents what a real network data would look like thus observing the blue neighborhood between the two lines show that how effectively the machine learning model was able to follow mimic the real traces of mobile network congestion data.

As we can see from the above figures the two models were able to follow the test dataset thus we can conclude that the results found were prominent that can respond well to real mobile network parameters from the KPI.

4.5. Comparative Analysis of Results

The predictions of a mobile network congestion using multilayer perceptron have been implemented in this project. The project tried to use to models in order to match the required prediction satisfactory prediction levels. The first model was an MLP having ten hidden layers each consisting of two hundred neurons and a fifteen neuron input layer to accommodate the available network parameters of the network data gathered for three selected sites. The output layer comprised of one neuron to show the level of the network congestion level. The second model consisted of fifteen hidden layers each having two hundred neurons and accommodate fifteen input neurons at the input layer. The output layer consisted of a single neuron to predict the congestion level in advance.

The mean squared prediction error of Model1 was 0.345 on the test data and the mean squared prediction error of Model2 was 0.272 and the graphical comparison of each model prediction with the congestion from the test data was also confirming that the performance of the models were satisfactory. In each of the results it is observed how effectively the machine learning model was able to follow mimic the real traces of mobile network congestion data. The results of this research showed that performance analysis of MLPNN models is a crucial process in model implementation of MLPNN for mobile network congestion prediction and a multilayer perceptron having 15 layers can give a comparable prediction of the real mobile network congestion situation.

Table 4.1 Comparative Analysis of Results

Author	**Objective**	**Algorithm(s) Used**	**Model Used**	**Metrics& Results**
Siddiqui and Choudhary [20]	The study is under taken predict voice traffic congestion in busy hours. To provide maximum utilization of voice traffic, they observe QoS report at constant time interval.	multilayered feed forward NN with back propagation	feed-forward neural network	The simulated results to measure Network quality and reduce loss of congestion. Show that once the L77influential variables that affect congestion are decided, the neural network Mainly focus on prediction voice traffic with above mention QoS parameters using neural network in daily basis and work on data from the real word for quality test.But this thesis use similar algorithm and model but it measure the Network quality well.
Simone Manganite, Michael Schapiro [7]	MORC is presented a novel rate-control protocol for mobile networks.	MORC's online learning algorithm	PCC framework	Bandwidth Utilization shows the higher throughput achieved by the Server with MORC enables. But MORC is still far from realizing the full potential of next-generation mobile Networks.
Sneha	The problem of	Support Vector	Mobility Model	The results which measure

Kumar Kasera, Ramachandran Ramjee, Sandra R. Thuel [15]	congestion control in the IP RAN of a CDMA wireless access network and examined three control techniques,	Regression (vSVR) algorithm.		user mobility, Call generation, call termination, and soft-handoffs were very promising. Both policies were shown to be able to adapt to significant congestion without either increasing the frame error rate or blocking. Issues related to data traffic, downlink congestion, and wireless RANs.But this thesis use the number of hidden layer neurons increasing continually, we get the satisfying prediction result.
Min Liu et al., [19]	Proposed and implemented an algorithm of prediction for optical networks congestion degree by constructing BP-ANN.	Multiple layer perception Neural Network	deep neural networks including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU)	The results measure Congestion degree and bandwidth or holding time of services in where 60 samples are selected from the best prediction illustrate the goodness-of-fit of the BP-ANN models. The trained BP-ANN model can accurately predict the output with an average and maximum relative error 0.75% and 3.5%, respectively. The prediction accuracy is related with the number and quality of example

				data and the accuracy of network model. By improving them, we will get predictions that are more accurate. But this thesis use MLP_NN and get Model1 which consisted of 10 layers was having an average loss value of 1.192 and mean squared error of 0.345 it measure the Network congestion well.
Guiomar Corral et al, [16]	They propose a system to reduce short-term congestion in ATM networks	UPC algorithm	Node, Process model,OPNET	The best results to analysis buffer utilization are obtained in situations of heavy load. If there is not much traffic in the network, a congestion But, The prediction of the future buffer utilization is the main goal of the new algorithm
Raheem and Okeene [9]	They propose a GSM congestion prediction model based on multilayer Perceptron neural networks (MLP-NNs) with sigmoid activation function and Levenberg Marquardt Algorithms (LMA)	Levenberg-Marquardt Algorithm	A Multi-Layer Perceptron Feed-forward neural network	The performance measure Network quality and reduce loss of congestion of the developed neural network model, a linear regression between the network outputs and the corresponding targets was carried out. Three parameters were returned. With availability of relevant historical traffic data, artificial neural networks can model the

	using twelve month real traffic data.			behavior of mobile network to predict the occurrence of network irregularities. But this thesis use similar model and Two parameters used the first train loss and the second is validation loss and measure Network Quality well.
Sophia & Olatokun [14]	Findings from the study showed that apart from the carrying capacity of the MTN network	multilayered feed forward NN with back propagation	feed-forward neural network	The extent of available capacity of network revealed that on the average, the capacity of calls the network can handle nationwide within a time slot is 450,000 calls, since MTN has about 2,500 BTS and one BTS can handle 180 calls at a Time. could be conducted to determine the level of GSM for social-economic activities and how it impacts on congestion
Yeshinegus [31]	The derived model can be integrated with the existing system to detect frauds in telecommunication companies, specifically in ethio telecom	PART and J48	CRISP-DM (Cross industry Standard process for data mining	The experiment result showed Detect frauds that the model from the PART algorithm Exhibited 100% accuracy level followed by J48 algorithm with 99.98%.For this study prepaid sample voice (call detail record) CDR data has been used along with SMS, GPRS and

				other data such as pre-paid wallet recharge log from OCS and CCB data warehouse in Ethio telecom
Dereje [4]	They shows the knowledge which was discovered during analysis of each cluster and the relationship between attribute against CSSR	K-means algorithm	WEKA tool and CRISP data mining process model.	The study result reveals Performance of the network which attribute should Enhance to improve the call setup success rate. Enhancing CSSR leas giving QoS to Customers and it implies customer satisfaction and increases company revenues ET to apply data mining technique using cluster analysis on GSMMobile network data to analyze the data, evaluate the performance of the network, toAssess the quality of the service and to make better decision.
Asemelash[32]	The research work was conducted base on Nonlinear Auto regressive(NAR), Neural Network time series prediction method using Addis Ababa 3G mobile sites in	Levenberg-Marquardt algorithm	A Multi-Layer Perceptron Feed-forward neural network	As aresult to measure fault occurrence time the best model is selected with minimum value of mean square error of prediction.Also, the model is tested with actual fault occurrence time which was not used in thetraining and achieved 90.71% in prediction. They used Levenberg-

	a case study.			Marquardt algorithm to train the neural Network and an iterative approach of hidden layer neural number selection is applied. But this thesis use Each dataset was divided into two sets that are the train and test data which the train data is 70% of the total dataset and is used to train the model and the test data were 30% of the total dataset is used to test the performance of the trained model on the data and get an average loss value of 1.2781 and a mean squared error of 0.272 during the testing phase.
This Work		Adam-Optimization	A Multi-Layer Perceptron Feed-forward neural network	This work use two model, to measure network congestion. The first model was an MLP having ten hidden layers each consisting of two hundred neurons and a fifteen neuron input layer to accommodate the available network parameters of the network data gathered for three selected sites. The output layer comprised of one neuron to show the level of the network congestion level. The second

				model consisted of fifteen hidden layers each having two hundred neurons and accommodate fifteen input neurons at the input layer. The output layer consisted of a single neuron to predict the congestion level in advance. The mean squared prediction error of Model1 was 0.345 on the test data and the mean squared prediction error of Model2 was 0.272 and the graphical comparison of each model prediction with the congestion from the test data was also confirming that the performance of the models were satisfactory. In each of the results it is observed how effectively the machine learning model was able to follow mimic the real traces of mobile network congestion data. Therfore the algorithm and the model used in this thesis perform well for Network congestion Prediction compare than others.

We can conclude that Model2 can best fit with the presented data that Model1 do. Also, we can see the trained network was simulated with new input data and our results show that the behaviors of the developed models were close to a real network situation. With the availability of relevant historical traffic data, multilayer neural networks can model the behavior of mobile network to predict the occurrence of network irregularities. With the number of hidden layer neurons increasing continually, we get the satisfying prediction result. Hidden layer neurons from 5 to 10, as a result of the simple neural network structure, the prediction accuracy is not ideal.Therfore the algorithm and the model used in this thesis perform well for Network congestion Prediction compare than others.

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORKS

The previous chapter discussed about worth issues in the design and implementation of the MLP_NNs model. In this chapter, the researcher concludes the overall work of the study and provides recommendation for other problems to be investigated

5.1 Conclusions

Mobile traffic congestion pattern of three sites in Addis Ababa was studied. The Multilayer perceptron feed forward network was used to model the network. The multilayer perceptron models were trained using real time data collected from the Key Performance Indicator table obtained from Ethio Telecom. The data was first tested for the presence of a reasonable correlation between each of the KPI element parameters and the target TCH congestion variable which shows the number of free traffic channels available for setting up new call or continuing with the present connection.

Two multilayer neural networks namely Model1 and Model2 were built on Jupiter notebook environment using the python programming language. Model1 which consisted of 10 layers was having an average loss value of 1.192 and mean squared error of 0.345 and Model2 which was made up of 15 hidden layers was observed to have an average loss value of 1.2781 and a mean squared error of 0.272 during the testing phase. We can conclude that Model2 can best fit with the presented data that Model1 do. Also, we can see the trained network was simulated with new input data and our results show that the behaviors of the developed models were close to a real network situation. With the availability of relevant historical traffic data, multilayer neural networks can model the behavior of mobile network to predict the occurrence of network irregularities.

5.2 Future Works

Despite the limited availability of data and the differing data characteristics, the result of the machine learning models in this study gives a clear indication that machine learning models could be part of future prediction models in mobile network congestion handling and avoidance algorithms. Indeed, machine learning models are software based and does not need hardware engagement to function. In favor for machine learning models, hardware related engagements are often complicated and expensive due to implementation times and expensive equipment. The step from the current stage to the situation where the model is used by Ethio telecom is not large at all. The only missing component is to code a script that can continuously read data from the database, obtain the relevant variables, and perform the mobile traffic congestion prediction in real time to assist current congestion handling and avoidance algorithms. Due to the relative simplicity of implementing machine learning models, the telecommunication industry should implement the framework presented in this research to enable the machine learning algorithms as an enhancing add-on to existing mobile network congestion handling and avoidance algorithms.

REFERENCES

[1] W. Bogale, "A Background paper on telecom and telecom statistics in Ethiopia," 2005.

[2] **www.ethiotelecom.et.(2020).Brief historical review of telecom sector in Ethiopia.**

[3] **T. Estifanos, "Telecommunication In Ethiopia,'' 2010.**

[4] D. Gebremariam, "Discovery of hidden Knowledge From Ethio Telecom mobile network data,"Addis Ababa University, 2015.

[5] **http://www.ethiotelecom.et/2011-efy-first-half/.(2018). Ethio telecom 2011 First Half Business performance summary report.**

[6] **https://www.ethiotelecom.et/.(2020). ethio-telecom-2012-efy-2019-20-annualbusiness-performance-summary-report.**

[7] S. Mangiante, M. Schapira, "congestion control for future mobile network," India, 2018.

[8] M. U. Nayarah Shabir Khan, Anand K, Shabia S.K, " Early Prediction of Congestion in GSM base on Area location using Neural Network," 2016.

[9] **M. A. Raheem, O. U. Okereke**, " A Neural Network Approach to GSM Traffic Congestion Prediction," 2014.

[10] L. DEYU, "Date Mining Approach to Analyze Mobile Telecommunication Network Quality of Service: the Case of Ethio Telecom," Addis Ababa university, 2014.

[11] G.Gebremeskel, "Data mining application in supporting fraud detection: on Ethio mobile service," 2006.

[12] G.Jemebr, "Data mining application in supporting fraud detection on mobile communication: the case of Ethio mobile, " 2005.

[13] W. Staling, "Wireless Communication and Network".

[14] T. A. Mughele.E. Sophia & Wole Olatokun," Congestion Control Mechanisms and Patterns of Call Distribution in GSM Telecommunication Networks: The Case of MTN Nigeria," vol. 5, 2012.

[15] S. Kasera, R. Ramjee, S. Thuel," **Congestion Control Policies for IP-Based CDMA Radio Access Networks,"** 2005.

[16] G. Corral, A. Zaballos, J. Camps, J. Garrell, "Congestion Control in ATM Networks using Artificial Intelligence Techniques,".

[17] Pragyan Verma a, Preeti Sharma b,Sattyam Kishore Mishra c, "Dropping of Call Due to Congestion in Mobile Network," 2012.

[18] O. U. O. Aliyu Ozovehe, Anene E.C., and Abraham U.Usman, "Traffic Congestion Analysis in Mobile Macro cells," 2016.

[19] M. Liu, M. Zhang, W. Zhao, C. Song, D. Wang, Q. Li, Z. Wang," Prediction of Congestion Degree for Optical Networks Based on BP Artificial Neural Network, "2017.

[20] A. K. C. Khadim Moin Siddiqui, "Telecom Voice Traffic prediction for GSM Using Feed Forward Neural Network," vol. 5, 2013.

[21] E. Markus, J. Agee, O. U. Okereke," Predicting Telephone Traffic Congestion Using Multi-Layer Feed forward Neural Networks," 2011.

[22] J. Alan Bivens et al., "Embrocates developed Network Congestion Abbreviation and Source Problem Prediction Using Neural Network," 2002.

[23] Aliyu Ozovehe, studied Mobile Soft Switch Traffic Prediction using Polynomial Neural Networks.

[24] Q. Do, T. Hang Doan, T. Nguyen, N. Duong and V. Linh, "Prediction of Data Traffic in Telecom Networks based on Deep Neural Networks," vietnam, 2020.

[25] P. Torres, H. Marques, P. Marques, J. Rodriguez, "Using Deep Neural Networks for Forcasting Cell Congestion on LTE Networks," 2018.

[26] N. Kumar, M. Raubal, "Applications of deep learning in congestion detection, prediction and alleviation.."

[27] S. Zhang , Y. Yao, J. Hu , Y. Zhao, " Deep Auto encoder Neural Networks for Short-Term Traffic Congestion Prediction of Transportation Networks," 2019.

[28] N. Ogwueleka, "Fraud Detection in Mobile Communications Networks Using User Profiling and Classification Techniques, " 2009.

[29] B. Kusaksizoglu, "Fraud Detection in Mobile Communication Networks Using Data Mining," 2006.

[30] Y. Getaneh," Predictive Modeling for Fraud Dection in Telecommunications : The Case of Ethio Telecom , Addis Ababa University, Addis Ababa, " 2013.

[31] A.Tesfay, "Neural Network Based 3G Mobile Sites Fault Prediction: A case Study in AddisAbaba, Ethiopia,2018.

[32] D. P. Pitambare, "Survey on Optimization of Number of Hidden Layers in Neural Networks, " 2005.

[33] P. Harrington, “Machine learning in action. Manning Publications,"2012.

[34] Ethio-telecom (November 2013). Press release. Retrieved from http://www. Data Mining Approach to Analyze Mobile Telecommunication Network QoS: The case of ethio telecom

[35] ETSI," Speech Processing, Transmission and Quality Aspects (STQ); QoS aspects for popular services in GSM and 3G networks,"2004.

[36] F. Carvalho & T. Magedanz "Telecommunication Systems and Technologies", Berline: Encyclopedia of Life Support System (ELSS) (2007.

[37] Ali, M. Shehzad, A., & Akram, M. "Radio Access Network Audit & Optimization in GSM (Radio Access Network Quality Improvement Techniques)", 2010.

[38] P. Kumar, Anuradha.B, Vivek and Naresh,"Improvement Of Key Performance Indicators and QoS Evaluation in Operational GSM Network,"2004.

[39] G. Gomez & R. Sanchez," End-to-End Quality of Service over Cellular Networks", 2005.

[40] B. Haider, M. Zafrullah, M.K. Islam," Radio Frequency Optimization & QoS Evaluation in Operational GSM Network."2009.

41] W. Hardy." QoS Measurement and evaluation of telecommunications quality of service", 2001.

[42] ETSI “Key Performance Indicators (KPI) for UMTS and GSM Technical Specification," (2010)

[43] R. Horak, “Telecommunications and data communications handbook,"2007.

[44] Krose, V. & Smagt, P, "An Introduction to neural network Approach to Analyze Mobile Telecommunication Network QoS," 1996. The University of Amsterdam

[45] S. Kyriazakos, D. Drakoulis, G. Karetsos."Practical radio resource management in wireless

systems,"2004.

[46] Y. Singh & A. Chauhan, “Neural Networks in Data Mining. Journal of Theoretical and Applied Information Technology,"2009.

[47] L. Sorokosz& W. Zieniutycz, “Artificial Neural Networks in Microwave Components and Circuits Modeling,"2012.

[48] H. Hippert, C. Pedreira, and R. Souza, “Neural Networks for short-term load forecasting,”2001.

[49] K. Dotche ,F. Sekyere, W. Banuenumah "UMTS Network planning, optimization, and inter-operation with GSM."2008.

[50] S. Kyriazakos, N. Papaoulakis, D. Nikitopoulos, E. Kechagias," A Comprehensive Study and Performance Evaluation of Operational GSM and GPRS Systems under Varying Traffic Conditions. IST Mobile and Wireless telecommunications Summit," 2002. Greece

[51] Qi. Zhang,K. Gupta, V. Devabhaktuni "Neural Network for RF and Microwave Design,"2000.

[52] S. Sumathi & S. Sivanandam, “Introduction to Data Mining and its Applications. Springer Tele Management Forum Telecom Operations,"2006.

[53] Qi. Zhang,K. Gupta, V. Devabhaktuni Neural Network for RF and Microwave Design,"2000.

[54] S. Dudul. and Ghatol, "Identification of Linear Dynamical Time- Invariant Systems using Feed forward Neural Network,"2008.

[55] L. Breiman, J. Friedman, R. Olshen, C. Stone, "Classification and Regression Trees", 2004.