

NETWORK DATA ANALYTICS IN 5G NETWORKS USING MACHINE LEARNING TECHNIQUES

Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering degree in Computer Science and
Engineering by

RANGISETTI DINESH SIVA SAI SANDEEP (Reg. No. 37110622)

RAVIKANTI GOPICHAND (Reg. No. 37110630)



**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC JEPPIAAR NAGAR, RAJIV
GANDHI SALAI, CHENNAI – 600119**

MARCH - 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide **RANGISETTI DINESH SIVA SAI SANDEEP (Reg. No. 37110622)** and **RAVIKANTI GOPICHAND (Reg. No. 37110630)** who carried out the project entitled "**NETWORK DATA ANALYSIS IN 5G NETWORKS USING MACHINE LEARNING TECHNIQUES**" under my supervision from **August 2020** to **March 2021**.

Internal Guide

Dr. L. SUJHELEN M.E., Ph.D.,

Head of the Department

Dr. S. VIGNESHWARI M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **RAVIKANTI GOPICHAND** hereby declare that the Project Report entitled **“NETWORK DATA ANALYSIS IN 5G NETWORKS USING MACHINE LEARNING TECHNIQUES”** is done by me under the guidance of **Dr. L.SUJIHELEN M.E., Ph.D.**, Department of Computer Science and Engineering at **Sathyabama Institute of Science and Technology** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

RAVIKANTI GOPICHAND

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA M.E., Ph.D., Dean, School Of Computing** and **Dr. S. VIGNESHWARI M.E., Ph.D.,** Head of the Department, Dept. of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. L.SUJIHELEN M.E., Ph.D.,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of **Computer Science and Engineering** who were helpful in many ways for the completion of the project.

.

ABSTRACT

5G is being developed to accommodate today's digital society's massive increase in data and connectivity, as well as the internet of things with billions of connected devices and future technologies. The network data analytics feature (NWDAF) allows network operators to use their own machine learning (ML)-based data analytics methodologies or incorporate third-party solutions into their networks. 5G is designed to be a key enabler and a leading infrastructure provider in the information and communication technology sector, enabling a range of upcoming networks with varying requirements. It is driven by the need to meet today's rising mobile traffic. Machine learning (ML) is expected to be critical in making the 5G vision a reality, given the network's emerging complexity and the emergence of novel use cases such as autonomous vehicles, industrial automation, virtual reality, e-health, and a variety of intelligent applications. 5G will be financially advantageous because consumers will gradually upgrade their devices or buy a smartphone that is 5G compliant, resulting in a high financial turnover for telecom companies. 5G is now being embraced as the telecommunications industry's future, offering network stability and higher download rates, as well as enormous benefits for both customers and companies. 5G networks are currently being developed by service providers all over the world. Furthermore, 5G is a game-changing technology that allows smart devices, including Android phones, to use artificial intelligence software applications. Other emerging technologies, such as autonomous driving, which rely heavily on network communications, would benefit as well.

TABLE OF CONTENTS

	ABSTRACT	v
	LIST OF FIGURES	viii
CHAPTER NO.	TITLE	PAGE NO.
1.	INTRODUCTION	1
	1.1 INTRODUCTION FOR 5G	1
	1.2 NWDAF	2
	1.3 SOFTWARE ENVIRONMENT	4
2.	LITERATURE SURVEY	10
3.	METHODOLOGY	13
	3.1 AIM OF PROJECT	13
	3.1.1 OBJECTIVES	13
	3.2 SCOPE	13
	3.3 SYSTEM DESIGN	14
	3.3.1 PROPOSED SYSTEM	14
	3.3.2 ADVANTAGES OF PROPOSED SYSTEM	14
	3.4 EXISTING SYSTEMS	14
	3.4.1 DISADVANTAGES OF EXISTING SYSTEM	15
	3.5 REQUIREMENT SPECIFICATION	15
	3.6 SOFTWARE REQUIREMENTS	15
	3.7 FEATURES OF 5G	15
	3.8 MODULES	16
	3.8.1 ML IN CELLULAR NETWORKS	16
	3.8.2 DATA PREPROCESSING	16
	3.8.3 DATA ANALYTICS	16
	3.8.4 A CONTEXT INFORMATION PREPROCESSING	16
	3.8.5 PROJECT IMPLEMENTATION	17
	3.9 MACHINE LEARNING	17
	3.9.1 ML ALGORITHMS USED	17
	3.9.1.1 RANDOM FOREST	17

	3.9.1.2 SUPPORT VECTOR MACHINE	18
	3.9.1.3 LINEAR REGRESSION	18
4	RESULTS AND DISCUSSION	19
	4.1 RESEARCH AND SIGNIFICANCE	19
5	CONCLUSION AND FUTURE WORK	23
	APPENDIX	
	A. PLAGIARISM REPORT	25
	B. JOURNAL PAPER	26
	C. SOURCE CODE	40

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
1.2.1	NWDAF PROCESS	3
1.2.2	NWDAF WITH 5G ARCHITECTURE	4
4.1	DATASETS	19
4.2	QUANTITATIVE DATA	20
4.3	FEATURES OF DATASETS	20
4.4	ROC CURVE FOR CHECKING MODEL	20
4.5	CHECKING ACCELERATION LENGTH OF MODEL	21
4.6	EDA OF CALLS WITH PLAN	21
4.7	EDA OF CALLS WITH DIFFERENT DATASETS	22
4.8	EDA OF DATA WITH DIFFERENT DATASETS	22

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION FOR 5G :

The “G” in 5G stands for “generation” and 5 represents a numerical progression. The first wireless phone technology was 1G, which was upgraded to 2G in the early 1990s when companies allowed people to send text messages between two cellular devices, which captivated the world. The world gradually moved on to 3G, which gave people the ability to make phone calls, send text messages, and access the internet at lightning speed. Many of the features that were only available with the third generation of wireless technology were improved with 4G.

People could search the web at lightning speed, send text messages, make phone calls, and even download and upload large video files without encountering any problems or waiting long periods of time. Then, in order to improve 4G connectivity, businesses added LTE, which stands for "long-term evolution." LTE is the fastest and most reliable form of 4G, and it began competing in the market with innovations such as WiMax. While both innovations produced similar results, it was critical to establish a standard that could be used by all. LTE accomplished this by speeding up 4G technology, laying the groundwork for 5G. People will be able to download and upload Ultra HD and 3D video more easily with 5G. As a consequence, we may assume that the pace of life has increased. Imagining upgrading your data link from a backyard hose to a flames hose will be interesting. The distinction will be obvious and worth noting.

The ALLIANCES FOR NEXT-GENERATION MOBILE NETWORKS identify the following criteria for 5G networks:

- o Increased Data Transfer Speeds
- o 1 GB per second to a large number of employees on the same office floor at the same time
- o SPECTRAL performance is improved over 4G;
- o coverage speed is increased;

- o signalling efficiency is improved;
- o and legacy is reduced dramatically as compared to LTE.

Massive groups of major global telecommunications firms are now partnering to establish global 5G values. Despite the fact that most of those specifications are yet to be finalised, analysts expect it to be more compliant (with 4G and 3G) and have some global interoperability. With the exponential increase in consumer demand, 4G can now be easily replaced with 5G using a new advanced access technology known as Beam Division Multiple Access (BDMA) and or Filter Bank Multi Carrier (FBMC) multiple access. Consider the case of a base station interacting with mobile stations to grasp the idea behind BDMA techniques.

Every mobile station has an orthogonal beam, which can be split according to the positions of the mobile stations using the BDMA technique for openhanded multiple accesses to the mobile stations, which also improves the system's competency and is the key mechanism of this communication.

The decision to switch to 5G is focused on current trends; it is generally believed that 5G cellular networks will overcome six challenges that 4G cannot, namely:

1. higher bandwidth,
2. higher data rate,
3. lower end-to-end latency,
4. access to large devices, and
5. lower cost.
6. Quality that is Consistent.

1.2 NWDAF

In 5G cellular networks, the NWDAF is a newly defined data analytics feature that provides network analysis in response to requests from other NFs. NWDAF can use any other NF as a data base. As a result, NWDAF and NFs have a two-way relationship. Note that Nnwdaf denotes NWDAF's service-based interface, while Nnf denotes any NF's service-based interface (for example, Npcf denotes PCF's service-based interface). By using the Nnwdaf interface, NWDAF can either provide network

analysis data to other NFs (i.e., analytics information) or NFs can request data distribution from NWDAF (i.e., events subscription). NWDAF also uses the Nnf gui to get data from other NFs.

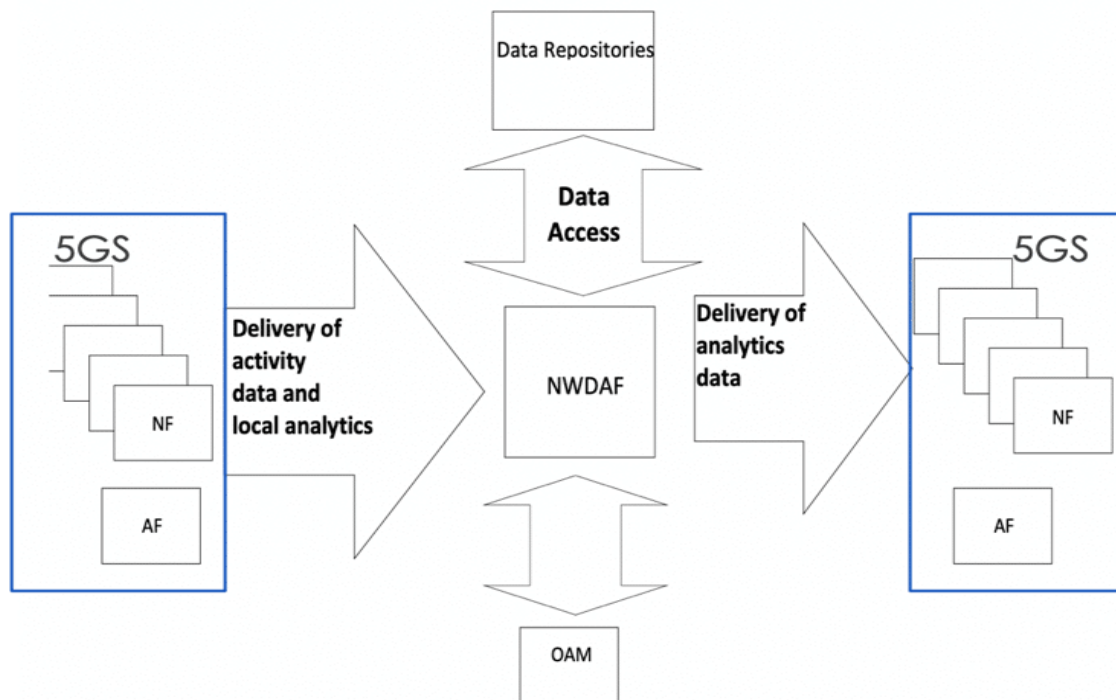


Fig 1.2.1 NWDAF PROCESS

Nnwdafe offers two services: event subscription and analytics information.

The Nnwdafe events subscription programme helps NF users to subscribe to and unsubscribe from various analytics events, as well as notify NF users that have a matching subscription regarding detected events. Nnwdafe analytics information, on the other hand, allows NF users to request and receive various types of analytic case information from NWDAF. As a result, the events payment service can be thought of as more of a billing service, while the analytics information service is the one that necessitates the use of AI/ML techniques. In light of these considerations, our primary emphasis is on the Nnwdafe analytics information provider.

The diagram illustrates the 5G Network Architecture, divided into three main sections:

- 5G RAN (Radio Access Network):** Located on the left, it includes a radio tower and a **User Equipment** (UE) icon. It connects to the **UPF** (User Plane Function) and an **Edge NWDAF** (Network Data Analytics Function).
- 5G Core:** The central cloud-like area containing various network functions:
 - Control Plane:** Includes the **Central NWDAF**, **NSSF** (Network Slice Selection Function), **AUSF** (Authentication Server Function), and **UDM** (Unified Data Management).
 - User Plane:** Includes the **UPF**, **AMF** (Access and Mobility Management Function), **SMF** (Session Management Function), **PCF** (Policy Control Function), and **AF** (Application Function).
 - Edge NWDAFs:** There are two **Edge NWDAF** blocks, one near the 5G RAN and another near the AF.
- ISP (Data Network):** Represented by a cloud icon at the bottom left, it connects to the **UPF** via the **User Plane**.

Connections and Interfaces:

- Control Plane:** Indicated by green lines, it shows connections between the **Central NWDAF** and **NSSF** (labeled **N34**), **AMF**, and **UDM**. It also shows connections between **AMF** and **SMF**, **SMF** and **PCF**, and **PCF** and **AF**. The **Edge NWDAF** near the 5G RAN connects to the **UPF** and the **Central NWDAF**. The **Edge NWDAF** near the AF connects to the **AF** and the **Central NWDAF**.
- User Plane:** Indicated by a blue line, it shows the path from the **ISP (Data Network)** through the **UPF** to the **5G RAN**.
- APIs (HTTP/JSON):** A green line connects the **AUSF** and **UDM**.
- N23:** A green line connects the **SMF** and the **PCF**.

Fig 1.2.2 NWDAF WITH 5G ARCHITECTURE

The following events can be analysed using NnwdaF analytics information:

- Detail on abnormal behaviour in a group of UEs or for a single UE.
- Detail on expected behaviour for a group of UEs or for a single UE.
- The success of a network in a specific region of concern.
- Network slice instance's load standard.
- Detail on NF load analytics for a particular NF.
- For a given UE, a communication pattern.
- Information about consumer data that has been gathered in a single location.

1.3 SOFTWARE ENVIRONMENT

- **Python:**
- Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.
- **History of Python**
 - Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.
 - Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.
 - Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).
 - Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python Features

Python's features include –

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.

- o **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- o **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- o **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- o **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- o **Databases** – Python provides interfaces to all major commercial databases.
- o **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- o **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

1. It supports functional and structured programming methods as well as OOP.
2. It can be used as a scripting language or can be compiled to byte-code for building large applications.
3. It provides very high-level dynamic data types and supports dynamic type checking.
4. It supports automatic garbage collection.
5. It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.
6. Python is available on a wide variety of platforms including Linux and Mac OS.

Let's understand how to set up our Python environment.

Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python <https://www.python.org>.

Windows Installation

Here are the steps to install Python on Windows machine.

Open a Web browser and go to <https://www.python.org/downloads/>.

Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.

To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.

Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

First Python Program

Let us execute programs in different modes of programming.

Interactive Mode Programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt –

```
$ python Python2.4.3(#1,Nov112010,13:34:43)
[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>>
```

Type the following text at the Python prompt and press the Enter –

```
>>>print"Hello, Python!"
```

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!")**; However in Python version 2.4.3, this produces the following result –

```
Hello, Python!
```

Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the following source code in a test.py file –

```
print"Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run this program as follows –

```
$ python test.py
```

This produces the following result –

```
Hello, Python!
```

Flask Framework:

Flask is a web application framework written in Python. Armin Ronacher, who

leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects. Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http methods –

Sr.No	Methods & Description
1	GET Sends data in unencrypted form to the server. Most common method.
2	HEAD Same as GET, but without response body
3	POST Used to send HTML form data to server. Data received by POST method is not cached by server.
4	PUT Replaces all current representations of the target resource with the uploaded content.
5	DELETE Removes all current representations of the target resource given by a URL

CHAPTER 2

LITERATURE

SURVEY

1. Optical access network technology for 5G wireless front/backend network

Trends in 5G mobile networks and RAN innovations are discussed in this article. Cloud RAN, which uses switches to allow versatile MFH networking, is addressed, as well as two successful MFH candidates in Cloud RAN.

2. Optimal Slice Allocation in 5G Core Networks

5G network slicing is critical for delivering versatile, scalable, and on-demand solutions for 5G networks' diverse set of applications. Role isolation (intra-slice) and ensuring end-to-end delay for a slice are two major challenges in 5G network slicing. By addressing these two problems, we answer the issue of optimum slice allocation in 5G core networks in this letter. We build on the work to construct a model that meets end-to-end delay constraints as well as separation between slice components for reliability.

3. Virtualising and orchestrating a 5G evolved packet core network

The design, development, and testing of a fully functional virtualised mobile core network are described in this paper. There are lessons learned and ideas for future improvements. To virtualise a commercial software evolved packet core solution deployed on standard off-the-shelf hardware, the researchers used open-source software for virtual network feature infrastructure management (OpenStack), network flow programming (OpenDaylight), and

network orchestration (OpenBaton). The results presented in this paper validate the

principle of function virtualisation in mobile networks and pave the way for future mobile core network function flexibility, which will be expected in 5G networks. The paper gives analysts and network providers firsthand insight of building similar virtual mobile network infrastructures, as well as highlighting the problems and concerns that must be addressed in order to fully leverage the potential of virtualisation in 5G networks..

4. An investigation of energy efficiency in 5G wireless networks

In next generation wireless networks along with the overwhelming demand of high data rate and network capacity, the user demands ubiquitous connectivity with the network. In order to fulfill the demand of anywhere at any time data services, the network operators have to install more and more base stations that eventually leads towards high power consumption. For this, the potential solution is derived from 5G network that proposes a heterogeneous environment of wireless access networks. More particularly, deployment of Femto and Pico cell under the umbrella of Macro cell base stations (BS). Such networking strategy will result high network capacity and energy efficiency along with better network coverage. In this article, an analysis of energy efficiency has been carried out by using two- tier and three tier network configurations. The simulation results demonstrate that rational deployment of small cells improves the energy efficiency of wireless network.

5. 5G network challenges and realization insights

5G technologies are a new paradigm strongly supported by the European Commission (EC) to overcome the challenges of next generation networks and

aiming to tackle the novel and manifold business requirements of vertical sectors such as Industry 4.0, Smart Grids, Smart Cities, and eHealth. Extraordinarily high speeds and capacity, multi-tenancy, fixed and wireless access network convergence, self-X, unconventional resource virtualisation, on-demand service- oriented resource allocation and automated management of resources are only few examples of the complex demands that 5G aims to undertake. The accommodation of such features requires multipronged efforts in different network technology domains at various levels. In this paper, we investigate how the relative 5G network features challenge the current generation of access networks, both fixed and wireless, and data centres, from an end-to-end network service point of view. Furthermore, we provide some insights into how emerging technologies such as Network Function Virtualisation (NFV) and Software Defined Networks (SDN) can be utilized for the accommodation of the above mentioned features and realization of 5G networks.

CHAPTER 3

METHODOLOGY

3.1 AIM OF PROJECT

The primary goal of this proposed system, “NWDAF” (NETWORK DATA ANALYSIS FUNCTION), is to allow operators to use their ml-based data analytics methodologies or integrate third-party solutions into their networks in such a way that they can obtain all of the necessary and detailed information about the data that can be analysed in a specific area or society, which will benefit both service providers and consumers. The proposed framework also offers data analysis status information ranging from current analysis methods in 4g and 3g to next generation methods such as 5g, which is an evolving and upgraded network with ml methods.

3.1.1 OBJECTIVES

The proposed model “NWDAF” (NETWORK DATA ANALYSIS FUNCTION) is based on cutting-edge technology that streamlines the entire process. It will be very necessary and profitable for the entire network providers, especially for data analysis. This system can provide more details, such as data analysis with plans, subscriptions, and call rates, as well as the required data across the network, which will be very beneficial for service providers to process data analysis, taking less time to complete the process and saving time and data.

3.2 SCOPE

This model would assist in analysing data using various methods from beginning to end without having to do so manually for each data across the network using various ml methods, saving time and energy. NWDAF will certainly assist in reducing the time taken to analyse data, thus saving time and effort, and will gradually support network operators and consumers in the future.

3.3 SYSTEM DESIGN

3.3.1 PROPOSED SYSTEM

In today's technologically advanced and interconnected world, both success and longevity are dependent on the ability to respond to changes over time. The ever-increasing need for high-speed networking and quicker data transfer protocols is one of the most important developments in the telecommunications industry. The exponentially growing demand for high-speed broadband has pushed 4G to its limits, making it inefficient in enabling the next telecommunications era. 5G network infrastructure is user-centric, while 4G network architecture is application-centric. 5G delivers the high-speed networking needed to enhance data access. By allowing the interconnection between sensors and computers, the ultra-low latency relation provided by 5G is expected to accelerate Internet adoption. To reach the highest degree of accessibility, Internet applications like asset and data monitoring, pattern analysis, and industrial automation need ultra-low latency.

3.3.2 ADVANTAGES OF PROPOSED SYSTEM

- High data rates per area and massive a large number of connected devices.
Thus, the interference among transmitters should be minimized.
- Congestion control and prioritization of data packets
- Reliability and cross-layering between network and transport layers
- Secure localization
- Extremely high data rates per device.

3.4 EXISTING SYSTEM

Increased bandwidth, improved data rate, lower latency, and better service efficiency are some of the primary priorities or demands that must be met in 4G. To meet these needs, wireless network architecture must be dramatically changed. This paper discusses the conclusions of a systematic survey of 5G cellular network infrastructure, as well as some of the key new innovations that can help develop the system and satisfy consumer demands.

3.4.1 DISADVANTAGES OF EXISTING SYSTEM

- Connectivity can be impaired by obstructions. Since frequency waves can only fly a small distance, 5G connectivity has a restricted range.
- The initial costs of implementation are substantial.
- Accessibility issues in remote areas.
- Devices' batteries are draining.
- The upload and update rates are not comparable.
- Taking away from the aesthetics.

3.5 REQUIREMENT SPECIFICATION

This proposed software runs effectively on a computing system that has the minimum requirements. Undertaking all the equipment necessities are not satisfied but rather exist in their systems administration between the customers machines already. So the main need is to introduce appropriate equipment for the product.

3.6 Software Requirements

- 1–10Gbps links to field endpoints (i.e. not theoretical maximum)
- End-to-end round trip latency is 1 millisecond.
- 1000x bandwidth per square metre
- ten to one hundred times the number of mobile devices
- 99.999 percent performance perception
- Perception of complete coverage
- The amount of electricity consumed in the network has been decreased by 90%.
- Low-power, machine-type computers have a battery life of up to ten years.

3.7 FEATURES OF 5G

Bandwidth: Supports 1Gbps or higher

Frequency bands: Sub-1 GHz, 1 to 6 GHz, > 6 GHz in mm bands (28 GHz, 40GHz).

Peak data rate: Approx. 1 to 10 Gbps.

Cell Edge Data rate: 100 Mbps and End to End delay: 1 to 5 ms.

3.8 MODULES

3.8.1 ML in cellular networks

The usage of ML and Artificial Intelligence (AI) techniques to perform autonomous operations in cellular networks has been widely used with the use cases that range from optimization of video flows to energy-efficient networks and resource allocation. The application of ML techniques to cellular networks is a topic that has gained a lot of attention recently, thanks to the revived importance of ML and AI throughout all facets of the industry. The surveys in present some recent results on how it is possible to apply regression techniques to mobile and cellular scenarios in order to optimize the network performance.

3.8.2 Data Preprocessing

The performance analysis presented in this Section is based on the Sample dataset. We sampled the number of users in each base station with a time step $T_s = 5$ minutes, and divided the dataset into a training set (which will be used for k-fold cross validation) and a test set. In Machine Learning (ML) processes, data preprocessing is critical to encode the dataset in a form that could be interpreted and parsed by the algorithm.

3.8.3 Data analytics

Data analytics has been widely accepted as a key enabler for 5G cellular networks. They can be applied in order to improve the performance of several mechanisms ranging from network management and traffic engineering to radio access network selection and traffic steering.

3.8.4 A Context Information Preprocessing

One of the most critical aspects of network context-based systems is the vast volumes of data. that need to be aggregated and transmitted towards a specific network analytics function deployed in the core network. The CIP (common

industrial protocol)

mechanism performs context data aggregation, filtering and preprocessing techniques. The module comprises aggregating and compressing mobile network-related context information, as well as techniques related to identifying and discarding user profile- redundant or unnecessary context data, before any transmission to the network data analytics function.

3.8.5 PROJECT IMPLEMENTATION

Intelligent cellular networks based on the state-of-the-art AI/ML techniques. In next-generation wireless technologies, including 5G, would require the support of extremely high data rates; thus, decision making in the new radio systems can benefit from ML techniques. They propose different ML techniques for different tasks in 5G networks including supervised learning-based methods for MIMO(MULTIPLE INPUT MULTIPLE OUTPUT) channel controlling, unsupervised learning (Decision tree, K- means clustering) for anomaly detection, and reinforcement learning for decision making (Random Forest) under unknown network conditions.

3.9 MACHINE LEARNING

Machine learning (ML) is the study of computer algorithms that develop themselves over time as a result of experience and data. It is considered to be a component of artificial intelligence. Machine learning algorithms create a model based on sample data, referred to as "training data," in order to make predictions or decisions without being specifically programmed. Machine learning algorithms are used in a wide range of applications, including email filtering and computer vision, where designing traditional algorithms to perform the necessary tasks is difficult or impossible.

3.9.1 ML ALGORITHMS USED

3.9.1.1 RANDOM FOREST

Random forest is a machine learning algorithm for supervised classification that employs the ensemble method. Simply put, a random forest is a set of decision trees that helps to solve the problem of decision tree overfitting. These decision trees are generated at random by choosing random features from a dataset. Based

on the

maximum number of votes earned from the decision trees, Random Forest makes a decision or makes a prediction. The random forest considers the final outcome as the one that has been reached the most amount of times across the various decision trees.

3.9.1.2 Support vector machines

SVM is supervised learning model with associated learning algorithms that analyze data used for classification and regression analysis. A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. The outcome which is arrived at, for a maximum number of times through the numerous decision trees is considered as the final outcome by the random forest.

3.9.1.3 LINEAR REGRESSION

Linear regression may be defined as the statistical model that analyzes the linear relationship between a dependent variable with given set of independent variables. Linear relationship between variables means that when the value of one or more independent variables will change (increase or decrease), the value of dependent variable will also change accordingly (increase or decrease).

Mathematically the relationship can be represented with the help of following equation –

$$Y = mX + b$$

Here, Y is the dependent variable we are trying to predict

X is the dependent variable we are using to make predictions.

m is the slop of the regression line which represents the effect X has on Y

b is a constant, known as the Y-intercept. If $X = 0$, Y would be equal to b .

3.10 ARTIFICIAL INTELLIGENCE

Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 RESEARCH AND SIGNIFICANCE

Data analytics usage to help the service of 5G cellular networks has recently become a hot research subject. Data analytics is used in a variety of network activities, including network management and monitoring, as well as policy implementation. Data is obtained from a variety of network components and can include a variety of information fields such as wireless channel quality, network load, accounting information, configuration and fault indicators, subscriber profile, and so on. This information is saved and updated on a regular basis. It is preprocessed after collection, with any necessary transformations, discretization, normalisation, outlier detection, and dimensionality reduction applied. The output of this procedure is then fed into a data analysis procedure, which creates a blueprint for extracting information from the processed data. The detection of circumstances where the occurrence of such particular events is likely to occur, for example, may be one of the outcomes of this procedure (e.g., a significant increase of the number of high mobility users). Causes a particular outcome (e.g., increase of the handover blocking probability). Any strategies for particular circumstances may be included in the information model (e.g., force the network components to place high moving users to macro cells). The results of the information discovery can then be passed on to strategy, management, or control modules. These modules can use this knowledge to boost the network's efficiency and optimise its operation. We present a systematic framework for using data analytics to help in the selection of Radio Access Technology (RAT) and traffic steering.

	State	Acc_len	Area_code	Int_plan	Vmail_plan	Vmail_message	Day_min	Day_calls	Day_charge	Eve_mins	Eve_calls	Eve_charge	Night_mins
0	0	128	415	0	1	25	265.1	110	45.07	197.4	99	16.78	244.7
1	0	107	415	0	1	26	161.6	123	27.47	195.5	103	16.62	254.4
2	0	137	415	0	0	0	243.4	114	41.38	121.2	110	10.30	162.6
3	0	84	408	1	0	0	299.4	71	50.90	61.9	88	5.26	196.9
4	0	75	415	1	0	0	166.7	113	28.34	148.3	122	12.61	186.9

FIG 4.1 DATASETS

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4617 entries, 0 to 4616
Data columns (total 20 columns):
State                4617 non-null int64
Acc_len              4617 non-null float64
Area_code            4617 non-null int64
Int_plan             4617 non-null int64
Vmail_plan           4617 non-null int64
Vmail_message        4617 non-null float64
Day_min              4617 non-null float64
Day_calls            4617 non-null float64
Day_charge           4617 non-null float64
Eve_mins             4617 non-null float64
Eve_calls            4617 non-null float64
Eve_charge           4617 non-null float64
Night_mins           4617 non-null float64
Night_calls          4617 non-null float64
Night_charge         4617 non-null float64
Int_mins             4617 non-null float64
Int_calls            4617 non-null float64
Int_charge           4617 non-null float64
Custserv_calls       4617 non-null float64
Churn                4617 non-null int64
dtypes: float64(15), int64(5)
memory usage: 721.5 KB

```

FIG 4.2 QUANTITATIVE DATA

	State	Acc_len	Area_code	Int_plan	Vmail_plan	Vmail_message	Day_min	Day_calls	Day_charge	Eve_mins	Eve_calls
count	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000	4617.000000
mean	0.014945	100.645224	437.046350	0.096600	0.267706	7.849903	180.447152	100.054364	30.676576	200.429088	100.179770
std	0.121345	39.597194	42.288212	0.295444	0.442812	13.592333	53.983540	19.883027	9.177145	50.557001	19.821314
min	0.000000	1.000000	408.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	74.000000	408.000000	0.000000	0.000000	0.000000	143.700000	87.000000	24.430000	165.900000	87.000000
50%	0.000000	100.000000	415.000000	0.000000	0.000000	0.000000	180.000000	100.000000	30.600000	200.800000	101.000000
75%	0.000000	127.000000	510.000000	0.000000	1.000000	17.000000	216.800000	113.000000	36.860000	234.000000	114.000000
max	1.000000	243.000000	510.000000	1.000000	1.000000	51.000000	351.500000	165.000000	59.760000	363.700000	170.000000

FIG 4.3 FEATURES OF DATASETS

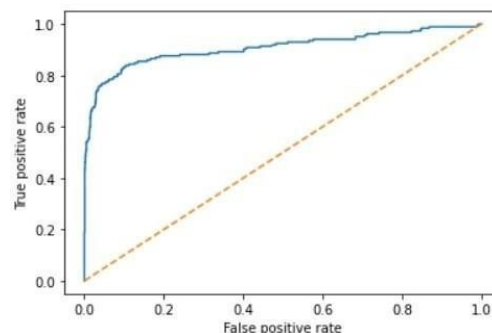


FIG 4.4 ROC CURVE FOR CHECKING MODEL

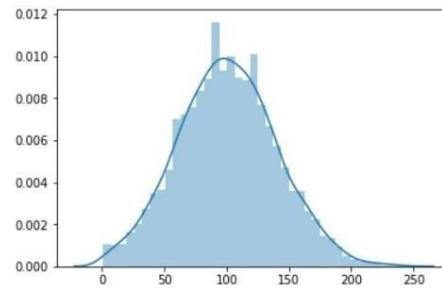
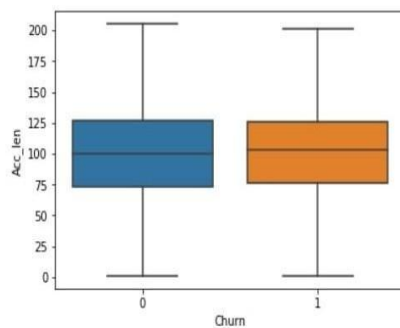


FIG 4.5 CHECKING ACCELERATION LENGTH OF MODEL

```
In [67]: #comparing the results there is not much difference
sns.boxplot(x='Churn',y='Acc_len',data=df,sym='')
```

```
Out[67]: <matplotlib.axes._subplots.AxesSubplot at 0x1a24ea79b0>
```

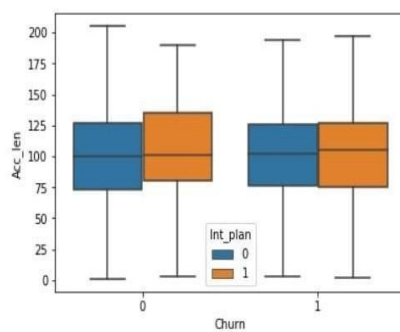


Activate
Go to Settin

```
In [ ]: #adding 3rd variable Intenation plan for the above analysis and compare the churn results
```

```
In [68]: sns.boxplot(x='Churn',y='Acc_len',data=df,sym='',hue='Int_plan')
```

```
Out[68]: <matplotlib.axes._subplots.AxesSubplot at 0x1a26322a90>
```



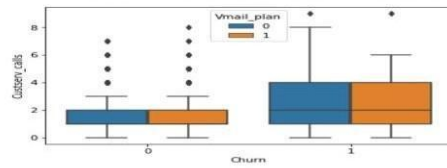
Comapring churn with customer service calls and also with international plan

Activate

FIG 4.6 EDA OF CALLS WITH PLAN

Comparing churn with customer service calls and voice mail plans

```
sns.boxplot(x='Churn',y='Custserv_calls',data=df,hue='Vmail_plan')
<matplotlib.axes._subplots.AxesSubplot at 0x1a21575c18>
```



Comparing churn with customer service calls and area code

```
sns.boxplot(x='Churn',y='Custserv_calls',data=df,hue='Area_code',sym='')
<matplotlib.axes._subplots.AxesSubplot at 0x1a263412e8>
```

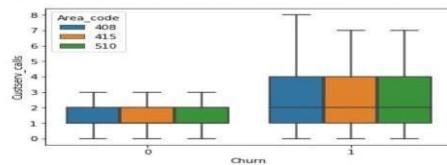
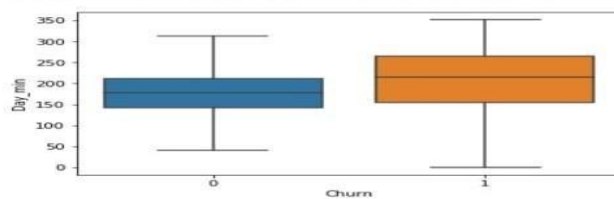


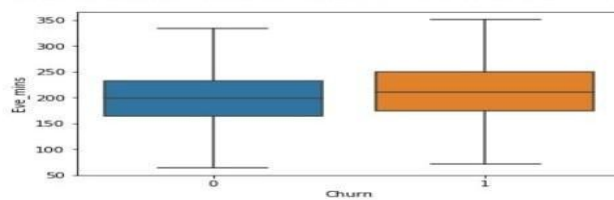
FIG 4.7 EDA OF CALLS WITH DIFFERENT DATASETS

comparing churn with minutes

```
sns.boxplot(x='Churn',y='Day_min',data=df,sym='')
<matplotlib.axes._subplots.AxesSubplot at 0x1a3a96fac8>
```



```
sns.boxplot(x='Churn',y='Eve_mins',data=df,sym='')
<matplotlib.axes._subplots.AxesSubplot at 0x1a289adef0>
```



```
sns.boxplot(x='Churn',y='Night_mins',data=df,sym='')
<matplotlib.axes._subplots.AxesSubplot at 0x1a3a2804a8>
```

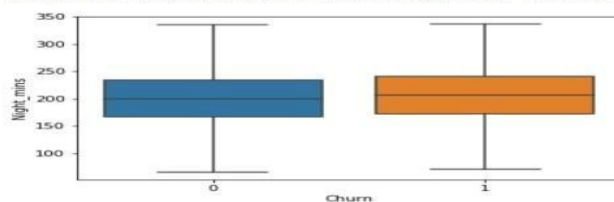


FIG 4.8 EDA OF DATA WITH DIFFERENT DATASETS

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

The next generation of cellular networks will include machine learning, software-defined networks, and edge cloud. We analysed how these three components can be used together in the system design for 5G networks in this paper, presenting insights and findings based on a dataset collected for more than a month from hundreds of base stations in a prototype cellular network in two different cities. We defined a clustering solution that minimises the need for inter-controller synchronisation and reduces control plane latency by restricting interactions among different controllers, and we evaluated the performance of the proposed approach using data from a real network. Then, using one month of data collected from the San Francisco base stations, we outlined a second possible application allowed by our design, presenting a comprehensive collection of results related to the accuracy of predicting the number of users in base stations. We demonstrated how the cluster-based architecture proposed in this paper would help to minimise prediction error. In comparison to a solution in which each base station attempts to perform the regression using only its own data, as is the case for a fully distributed architecture (e.g., LTE), the controller-based design enables data from several neighbouring base stations to be aggregated and to forecast a vector containing the number of users in the controller's nodes. This captures the spatial correlation created by user versatility, and decreases the RMSE by up to 53% when the temporal horizon of the prediction is increased. Finally, we discussed some prediction-enabled use cases, such as controlling the network or providing creative predictive services to network users, such as suggesting different driving routes to enhance the network user experience. We used a real-world example in the San Francisco area to demonstrate how the quickest path does not always result in the highest throughput or the least amount of downtime and that the best route based on these metrics (which we derive from the number of users in each base station)

can vary depending on departure time, making a prediction-based approach useful. This paper, we claim, is the first to discuss

many concerns related to the realistic application of machine learning techniques in 5G cellular networks, with findings and assumptions centred on a real-world dataset.

5.2 FUTURE WORK

We will test various prediction algorithms (e.g., neural networks) in the future to see whether it is possible to increase prediction accuracy even further, and we will expand the regression to other related network metrics (e.g., the number of handovers, utilisation) to check the limits of what can be predicted in a cellular network.

APPENDIX
A.PLAGARISM
REPORT



Plagiarism Checker X Originality Report

Similarity Found: 10%

Date: Sunday, March 07, 2021

Statistics: 432 words Plagiarized / 4471 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

NETWORK DATA ANALYTICS IN 5G NETWORKS USING MACHINE LEARNING
TECHNIQUES RANGISETTI DINESH SIVA SAI SANDEEP, RAVIKANTI GOPICHAND, L.
SUJIHELEN DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI

NETWORK DATA ANALYTICS IN 5G NETWORKS USING MACHINE LEARNING TECHNIQUES

RANGISETTI DINESH SIVA SAI

SANDEEP, RAVIKANTI GOPICHAND, L.

SUJIHELEN

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY,
CHENNAI

ABSTRACT: 5G is being developed to accommodate today's digital society's massive increase in data and connectivity, as well as the internet of things' billions of connected devices and future technologies. The network data analytics feature (NWDAF) allows network operators to use their own machine learning (ML)-based data analytics methodologies or incorporate third-party solutions into their networks. 5G is designed to be a key enabler and a leading infrastructure provider in the information and communication technology sector, enabling a range of upcoming networks with varying requirements. It is driven by the need to meet today's rising mobile traffic. Machine learning (ML) is expected to be critical in making the 5G vision a reality, given the network's emerging complexity and the emergence of novel use cases such as autonomous vehicles, industrial automation, virtual reality, e-health, and a

variety of intelligent applications.

KEYWORDS: Wireless connectivity, 5G, machine learning, and B5Gble are some of the terms used in this paper.

1. INTRODUCTION

The NWDAF, as specified by 3GPP TS 29.520, implements standard interfaces from the service-based architecture to collect data via subscription or request model from other network functions and procedures. This is for delivering analytics features in the network for automation or reporting, as well as resolving significant custom design or format issues. The NWDAF is supposed to provide a distributed

architecture that allows for
real-time analytics at the edge
and a central

function for analytics that require central aggregation (e.g., service experience). The NWDAF uses a request or subscription model to collect data and provide analytics services.

3GPP has specified an exclusive Analytics Function called "Network Data Analytics Function (NWDAF)" in the 5G Rel 15 and 16 Specifications. Since the NWDAF is a part of 5G Core, interworking nuances for Analytics service providers are streamlined. 3GPP also specifies Standard use cases that can assist any operator in managing subscriber QoE in a dynamic, growing network using Network Automation with the NWDAF as the brain of the architecture. Via analyzed Data Monetization and Targeted Marketing, Content Analytics also opens up new ways to produce additional revenue.

Established Telco Analytics vendors will need to make improvements in the Southbound (Data Source adoptors) to retrieve data from other Network functions using 3GPP specified services, as before 4G, all Network vendors supported based or probe vendor based custom data that is file-based or telemetry (Kafka) real-time streaming of data with Data lake. In addition to the current common REST API method for closed-loop automation, NWDAF needs Northbound updates to help along subscribe and event requests for

3GPP Analytics services. Unlike 4G- EPC, which uses various protocols such as S1AP, GTP, and Diameter-based Architecture, the 5G Core 3GPP uses NFV (Network Function Virtualization). Instead of using Services communicating uniformly via HTTP/2 as outlined in IETF RFC 7540 with JSON as the application layer serialisation protocol, 3GPP has used NFV (Network functions virtualization) to identify Nodes as Functions in 5G Core. The need for analytics in telecom networks can be traced all the way back to the days of the public switched telecommunications network (PSTN) and Signaling system no 7. (SS7). Simple network-node main performance indicators (KPIs) were used for network management and network performance monitoring, based on stats obtained from T1/E1 connection probes.

Analytics based on advanced AI/machine learning (ML) algorithms for predictive analytics, anomaly detection, pattern analysis, and clustering have been a top option for use cases like customer relationship management, targeted marketing, and data monetization as part of the 4G

evolution. In addition to network management, sophisticated AI/machine learning (ML) algorithms for predictive analytics, anomaly detection, pattern analysis, and

clustering have emerged as a top option for use cases such as customer service management, targeted ads, and data monetization as part of the 4G evolution. With the addition of 5G and its complexity, advanced prescriptive analytics would be much more important in driving closed-loop automation and self-healing networks. With the factors, a common pressure point for communications service providers (CSPs) is how to incorporate analytics into the network; currently, analytics are difficult due to the large number of nodes.

With the considerations, a common pain point for communications service providers (CSPs) is how to incorporate analytics into the network; currently, analytics are complicated due to the many non-standardized interfaces and inconsistent data collection approaches used by network vendors.

2. RELATED WORK

Since, NWDAF is still new in 5G cellular networks, the literature on the subject isn't as extensive as it should be, so this paper introduces a new work that calls NWDAF compliant deployment. The developers to add NWDAF's mobility and network slicing capabilities. The capabilities of NWDAF relevant to network slicing are defined. Unfortunately, there is no extensive collection of data sets to

use for the NWDAF scenarios studied in this paper in terms of traffic data for 5G. A ray-tracing simulation is used to generate data for a 5G multiple-input multiple-output

(MIMO) analysis.

Researchers in build a data set by tracking one eNodeB and one piece of consumer equipment. Packet Data Convergence Protocol (PDCP), Radio Resource Control (RRC), and Mobile Access Control (MAC) data are also included in the data collection. This data collection is therefore unsuitable for machine learning because of the monitoring's coverage and irregularity. To our knowledge, no user traffic data gathered from gNodeB based on 5G SA implementation and appropriate for the NWDAF scenarios of this paper has been released.

In the last decade, however, intelligent cellular networks built on cutting-edge AI/ML techniques have been extensively studied. The writers discuss the importance and necessity of using AI for next-

generation cellular networks (i.e., 5G and 6G), as well as some of the difficulties and roadblocks.

According to Jiang et al., next-generation wireless technology, such as 5G, would necessitate the funding of incredibly high data speeds, so decision-making in modern radio networks will benefit from machine learning techniques. They suggest supervised learning-based approaches for MIMO channel management, unsupervised learning for anomaly detection, and reinforcement learning for decision-making under uncertain network environments, among other ML strategies for various tasks in 5G networks. They may not, though, look at how machine learning might be used for network analytics in 5G. Without discussing NWDAF, Casellas et al. highlight the importance of allowing AI/ML strategies to monitor, handle, and orchestrate components of 5G networks.

In, machine learning is proposed as a method for managing self-organizing 5G networks. If 5G technology moves toward millimetre wave (mmWave) radio models, which will necessitate major network densification, self-organization of cellular networks becomes more prominent. To this end, the authors present a complex network management scheme that combines various machine learning techniques. In and, the authors look at recent research on cellular networks that use ANN/ML techniques. Because of security challenges, including authentication

problems, Fang et al. propose using machine learning methods for 5G and beyond cellular networks. The authors argue in that, as a result of unprecedented consumer demands on mobile and wire-less networking, real-time extraction of fine-grained analytics and agile network resource management are becoming increasingly important.

The investigators look at deep learning research in the literature and they see it as a way to satisfy previously unmet consumer demands. Various experiments focused on 5G wireless networks have already been conducted for more complex activities. Investigate the applications of machine learning in 5G vehicular networks. Additionally, investigate machine learning in mmWave massive MIMO to monitor the radio, including beamforming. In brief, numerous publications in the literature investigate machine learning (ML), artificial intelligence (AI), and deep learning (DL) principles for 5G cellular networks, both with and without consideration of NWDAF. These films, however, are only in their infancy. Furthermore, owing to the novelty of NWDAF, certain parts of the 3GPP specifications for NWDAF are

currently left void. Furthermore, the current data sets do not follow the criteria of this paper's NWDAF scenario. As a result, we produce our synthetic

data set based on 3GPP specifications.

3. PROPOSED SYSTEM

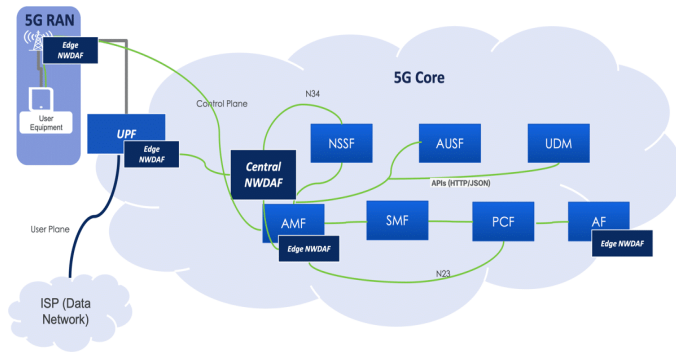
I. NWDAF

In 5G cellular networks, the NWDAF is a newly defined data analytics feature that provides network analysis in response to requests from other NFs. NWDAF can use any other NF as a data base. As a result, NWDAF and NFs have a two-way relationship. Note that Nnwdaf denotes NWDAF's service-based interface, while Nnf denotes any NF's service-based interface (for example, Npcf denotes PCF's service-based interface). By using the Nnwdaf interface, NWDAF can either provide network analysis data to other NFs (i.e., analytics information) or NFs can request data distribution from NWDAF (i.e., events subscription). NWDAF also uses the Nnf gui to get data from other NFs.

Nnwdaf offers two services: event subscription and analytics information. The Nnwdaf events subscription programme helps NF users to subscribe to and unsubscribe from various analytics events, as well as notify NF users that have a matching subscription regarding detected events. Nnwdaf analytics information, on the other hand, allows NF users to request and receive various types of analytic case information from NWDAF. As a result, the events payment service can be thought of as more of a billing service, while the analytics information service is the one that necessitates the use of AI/ML techniques. In light of these considerations, our primary emphasis is on the Nnwdaf analytics information provider.



5G Architecture with Distributed NWDAF



The following events can be analysed using Nnwdaf analytics information:

- Detail on abnormal behaviour in a group of UEs or for a single UE
- Detail on expected behaviour for a group of UEs or for a single UE
- The success of a network in a specific region of concern.
- Network slice instance's load standard,
- Detail on NF load analytics for a particular NF,
- For a given UE, a communication pattern
- Information about consumer data that has been gathered in a single location

II. TOOLS FOR MACHINE LEARNING

With recent technological advances and, as a result, the massive volume of data available and needed to be processed, machine learning algorithms have become very popular and are seen as a possible solution for a variety of problems. ML models are used to generate certain necessary information from historical records. There are different types of machine learning algorithms, each of

which is programmed to solve a specific problem. ML models are divided into three categories, and the ones we include in this paper are classified as supervised learning algorithms. Since the 5G data set produced is properly labelled, we will take advantage of the merits of supervised learning algorithms. We're focusing on two related issues in the report. We use separate solution-specific ML models for each of these problems and assess the efficiency of these models.

Prediction of Network Load Efficiency

As Known, one of the key objectives of this paper is to estimate the load efficiency of the network. We define this problem as a time series issue, and then we use three separate ML models, namely LR, RNN, and LSTM, where LSTM is a slightly tweaked version of RNN. When training these ML models, both the branded data collection (i.e. the data rate of each time the mark is slotted) and the derived features are used.

Linear Regression

LR matches the linear relationship between the particular

characteristics of
certain observations and
marks. Since LR is one of the
most widely used ML

models for forecasting and forecasting issues, we use this model as a base model for comparison. Recursive Neural Networks (RNNs) are a kind of neural network.

Many questions can be solved using neural networks and deep learning. Because of the design of neural networks, the algorithm is able to learn complex relationships among features and produce extremely accurate results. RNN is a specialised variant of the neural network algorithm that allows data from previous neurons to be carried over and improves prediction and forecasting precision. For the purpose of network load estimation, we use RNN. One plain RNN layer, four hidden layers, and one output layer make up the model. Mean absolute error (MAE) is the RNN's loss function, and it's also one of the efficiency metrics we consider.

Long-Term Memory vs. Short-Term Memory

LSTM is a tweaked clone of RNN with different neuron structures than RNN. Unlike simple RNNs, which hold information from previous neurons, the LSTM's dynamic structure allows it to keep information from very old records. In this way, LSTM will make correct predictions even though there is a time delay between observed

patterns. We use one LSTM layer, four hidden layers, and one output layer in the LSTM model. The loss function of the LSTM is MAE, much like the RNN model.

Detecting Anomalies

As previously said, this paper has two contributions from the viewpoint of the NWDAF, one of which we will look at in this subsection: obtaining insights into behaviour knowledge for a category of UE. During the development of the network traffic data collection, network load abnormalities are applied, as well as the marking of and time slot as usual or abnormal. Two separate machine learning methods, logistic regression and extreme gradient boosting, are used to classify this behaviour data.

Logistic Regression is a technique for predicting the outcome of multiple classification problems, logistic regression is a widely used model. In order to approximate the labels of the given results, logistic regression employs a logistic function. For the anomaly detection query, we use logistic regression as the base model. We set class weight parameters to solve the

problems that would be created by the imbalanced number of mark forms because the generated data has unequal numbers of anomaly and natural states in the simulation

(i.e., anomaly and normal). Help

Vector Machinery

SVM is a supervised learning model with related learning algorithms that analyses the data used for classification and regression analysis. A Support Vector Machine (SVM) is a discriminatory classifier formally described by a separating hyperplane.

The result that has been reached for a cumulative number of times across a number of decision-making trees is assumed to be the final result of the random forest.

III. DATA COLLECTION

The data collected from UE is passed to the 5G SBA, where NWDAF and other NFs are located. NWDAF is linked to other NFs through the Service-Based Interface (SBI), and NWDAF and other NFs make data transfer between them. NWDAF then collects input from various NFs and adapts multiple ML models to both forecast network load capacity and identify a network load anomaly. In view of the set topology, our method uses classified data to train; thus, it selects the finest ML model according to the characteristics of the topology. We use a fixed cell topology consisting of a fixed number of RRU cells, a fixed set of subscription categories and a fixed set of personal

equipment.

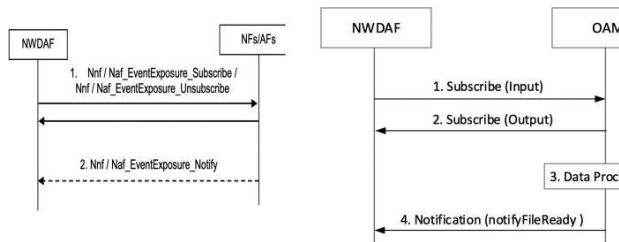
We use a fixed cellular topology, which includes a fixed range of RRU cells, subscriber groups, and personal equipment (i.e., device) types. Despite the fact that our device model can accommodate topologies with a wide number of RRU cells, subscriber categories, and personal equipment types, we only consider a network topology with five RRU cells in our simulations for the sake of simplicity. There are three subscriber divisions in each of these cells, which reflect platinum, gold, and silver subscriptions, respectively. Since mobile services are more practical, the rationale behind these three subscriptions is to make the provided data more realistic.

IV. DATA PREPROCESSING

We use a fixed cellular topology, which includes a fixed range of RRU cells, subscriber groups, and personal equipment (i.e., device) types. Despite the fact that our device model can accommodate topologies with a wide number of RRU cells, subscriber categories, and personal equipment types, we only consider a network topology with five RRU cells in our simulations for the sake

of simplicity. There are three
subscriber divisions in each of

these cells, which reflect platinum, gold, and silver subscriptions, respectively. Since mobile services are more practical, the rationale behind these three subscriptions is to make the provided data more realistic.



The performance analysis provided in this section is dependent on the dataset of the sample. We sampled the number of users at each base station with a time stage $T_s = 5$ minutes, and divided the dataset into a training set (which will be used for k-fold cross validation) and a test set.

In Machine Learning (ML) operations, data preprocessing is important for encoding a dataset in a form that could be interpreted and parsed by an algorithm.

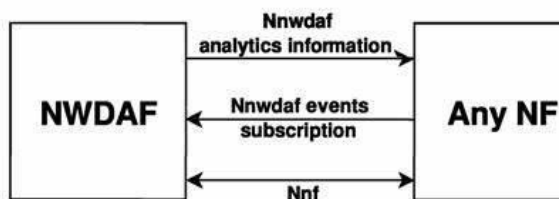
Since mobile service providers offer these types of subscriptions in the real world, the reasoning behind these three subscriptions is to make the produced data more practical. In addition, there are five distinct categories of personal equipment (i.e., consumer equipment) under each subscriber category: IoT unit,

car, mobile phone, smartwatch, and tablet computer. According to the proposed model, each subscriber segment and RRU cell's personal equipment has a predetermined amount of traffic load at the start of the simulation. As a result, from the beginning to the end of the simulation, network traffic is saturated. After that, for each simulation time level (t),

As a result, from the beginning to the end of the simulation, network traffic is saturated. Then, for each simulation time stage (t), a portion of the load is transferred from one cell (source) to another cell (target) adjacent to the source cell. Predetermined percentages are also used in the handover process. We adjust the mean handover ratios according to the time of day to make the traffic more practical. . Also, since IoT devices are not supposed to travel as frequently as other personal appliances we deem due to their construction, no significant change in mean handover ratios is expected for the time of day. We hope to achieve real-like consumer traffic using these carefully calibrated mathematical parameters and canonical approximations. The specifics of the distribution used to calculate the handover ratio.

V. DATA ANALYTICS

Data analysis has been commonly recognised as a primary enabler for 5G wireless networks. They can be used to enhance the efficiency of a number of mechanisms, ranging from network control and traffic engineering to radio access network selection and traffic management.



Throughout the evolution of computer science, there are various kinds of AI/ML models. Among these several AI/ML models, the algorithms behind these models are working differently. In general, it can be seen that ML algorithms can be divided into three distinct sections, namely supervised, unsupervised, and enhanced learning. When we consider supervised ML algorithms in this article, labelled data becomes key in this sense. As a consequence, the role of data generation is becoming a significant factor. Considering all of these, we create a branded data collection for 5G cellular networks. When selecting the fields of the proposed data set, we are driven by the 5G requirements released by the 3GPP consortium.

The below are the fields that have

been chosen:

- Data rate: The amount of data transferred in bytes during a given time span.
- Cell information for a group of UEs that are linked to the network.
- Subscription categories: Subscribing is the strategy of a network of UE.
- Personal equipment ID (PEID): A UE's device type details (e.g., cell phone, smart watch).

We come up with the following predefined parameters as input fields when creating the labelled data set.

- The number of RRU cells in the topology is represented by the RRU cell number.
- The names and IDs of the categories.

- Personal appliance IDs and unit

type names.

- Initial load settings, which show the initial load for each piece of personal equipment.
- Cell arrangements that are adjacent to each other.
- The amount of personal equipment that is handed over.
- For the handover operation, mean and variance ratios.
- Simulation time step, which reflects the length of time it takes to get data from NFs.
- Simulator time, which is the overall duration of the simulation.

State	Acc_len	Area_code	Int_plan	Vmail_plan	Vmail_message	Day_min	Day_calls	Day_charge	Eve_mins	Eve_calls	Eve_charge	Ni
0	0	128	415	0	1	25	285.1	110	45.07	197.4	99	16.78
1	0	107	415	0	1	26	161.6	123	27.47	195.5	103	16.62
2	0	137	415	0	0	0	243.4	114	41.38	121.2	110	10.30
3	0	84	408	1	0	0	299.4	71	50.90	61.9	88	5.26
4	0	75	415	1	0	0	166.7	113	28.34	148.3	122	12.61

Initial load configurations for each RRU cell for the beginning of the results, based on subscriber group and personal equipment type. As can be observed, various loads are allocated to each type of personal equipment and subscriber segment. The reasoning behind these load prices is that a customer is most likely to subscribe to the platinum category with a mobile phone subscription rather than the gold or silver categories. For a tablet computer, motorcycle, or IoT system,

though, it is less likely to fall into a higher range. In order to maintain network equilibrium, the mean handover ratios for RRU cells with four neighbouring RRU cells are also doubled.

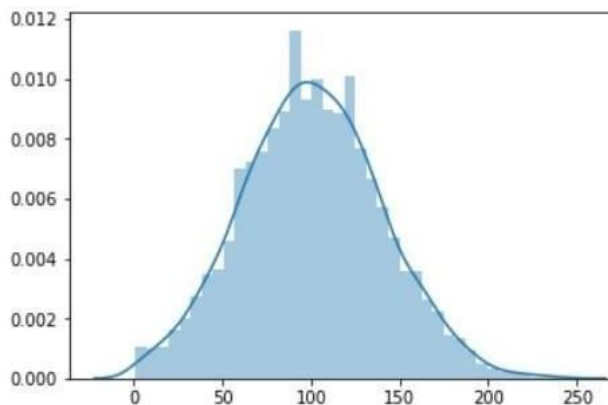
4. RESULTS AND DISCUSSION

The results are taken into account in two folds. In the first, we analyse performance network loading using LR, LSTM, and RNN templates. In the above case, we analyse deviations in the network throughout the simulation using logistic regression.

I. Prediction of network load efficiency

Using the three AI/ML models described above, we conduct a simulation for each user device in each subscriber group and RRU cell. Then we calculate the average percentage error (MAPE) and mean

absolute error (MAE) for both of these examples.



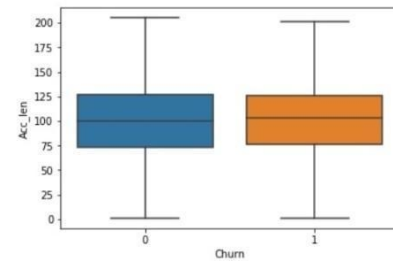
II. Detecting Anomalies

We use logistic regression and svm models to find irregularities in the 5G network traffic results. We use AUC-ROC as a performance metric to assess the success of these two models. We also use receiver activity characteristics (ROC) curves to visualise the performance. True positive rate (i.e., chance of identification -Pd) and false positive rate are the two axes of the ROC curve (i.e., false alarm rate -Pf). The ratio of true positives to real true values is known as Pd. It is a measure of the model's precision in detecting irregularities

(i.e., percentage of the successful predictions among the network states having anomaly).

Checking Acc_len for churners and non churners

```
In [67]: #comparing the results there is not much difference
sns.boxplot(x='Churn',y='Acc_len',data=df,sym="")
Out[67]: <matplotlib.axes._subplots.AxesSubplot at 0x1a24ea79b0>
```

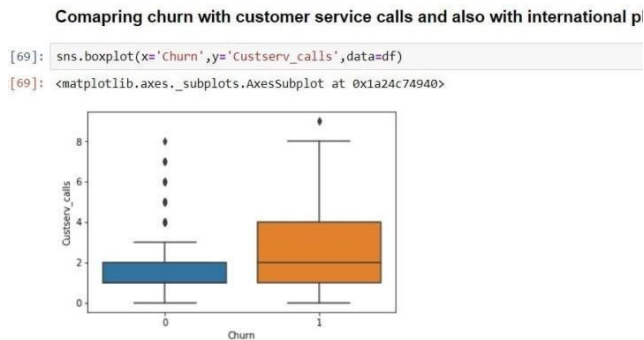


Pf is the ratio of false positives to true negative values on the other axis. When the real network condition is not showing anomaly, it reflects the likelihood of making a mistake (i.e., percentage of the false predictions for the non-anomaly states).

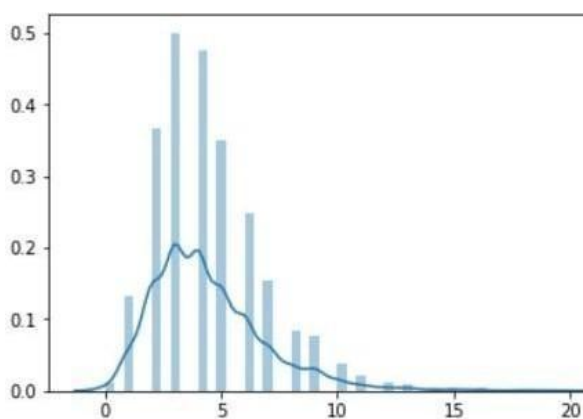
The AUC-ROC, consistency, and precision metrics for SVM and logistic regression models.

For each subscriber group and RRU cell, the AUC-ROC score of the SVM model is substantially higher than that of the logistic regression model. As a result, as shown by the average AUC-ROC, The SVM model models deviations even better than the logistic regression model in terms of consistency and precision points. Although precision does not seem to have improved dramatically,

accuracy has improved significantly.



In general, RRU cells are for a fixed false alarm rate, logistic regression is unable to raise Pd as effectively as XGBoost. If we consider the network topology in Section IV.B, where RRU cell three has more neighbours than RRU cell four, we can see that RRU cell three is more volatile than the other RRU cells since it has a higher handover ratio. Since logistic regression is a less complicated ML model than XGBoost, it performs worse in both cells with predictions and is unable to increase the success score sufficiently.



The ROC curves of the models do not differ much across subscriber groups.

We recognize that subscriber groups have no substantial impact on model results. In the other hand, when two ROC curves (i.e., logarithmic curves) are compared, it can be shown that We accept that the types of subscribers do not greatly influence the success of the model.

5. CONCLUSION

This paper proposes a novel system for intelligent network analysis for 5G cellular networks. To this end, we first explain NWDAF in the service-based 5G cellular network architecture, and then use multiple ML strategies to overcome two big problems. In the first query, network load is predicted using time series analysis, primarily using linear regression, LSTM, and RNN models. In the second problem, network anomalies are classified using state-of-the-art tree-based gradient boosting techniques, logistic regression models. In addition, we are introducing a comprehensive generation of a cell-based data collection to test network data analytics.

In addition, we are implementing a systematic generation of a cell-based data collection to test network data and alytics for 5G wireless networks using the fields specified in the 5G standard paper. In the tests, we see that the new ral network models outperform the linear regression model to correctly

estimate the

network load. Similarly, the tree-based XGBoost conducts logistic regression when classifying deviations in the network. In conclusion, we demonstrate the very practical use of NWDAF by using widespread and common ML models.

Due to the uniqueness of NWDAF, there are still several open-ended issues in the literature. One such problem is the multi-label naming of anomalies. That is because an anomaly occurrence will take place at various stages, and the classification of this event at its subsequent level is the ultimate target for network operators. Another problem is the enrichment of created data with the appropriate fields by drawing on the 3GPP requirements in order to observe and collect analytics at the level of the network. In addition, based on the created data collection, our future work involves the identification of contact patterns using exploratory data analytics.

Last but not least, our study can be expanded by using various AI/ML models while concentrating on other NWDAF capabilities.

V. REFERENCE

[1]A. Gupta and R. K. Jha, "5G Network Survey: Architecture and New Technology," IEEE Access, vol. 3, pp. 1206–1232, July 2015.

[2]M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Network-Works: Systematic Survey," IEEE Commun. Surveys & Tuts., vol. 18, no. 3, pp. 1617–1655, 3rd quarter 2016.

[3] "Mobile Visual Networking Index (VNI)," accessed 2019-12-05. [Online], please. Available: <https://www.cisco.com/c/m/en/us/service-provider/visual-networking-index.html>

[4]H. Shariatmadari, R. Ratasuk, S. Iraji, A. Laya, T. Taleb, R. Jäntti, and R.A. Ghosh, "Machine-type communications: current state and possible per-spectives for 5G networks," IEEE Commun. Mag., vol. 53, no. 9, pp. 10–17, Sept. 2015.

[5]J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. Soong, and J. C. Zhang, "What's going to be 5G?" IEEE J. Selected Areas Commun., vol. 32, no. 6, pp. 1065–1082, June 2014.

[6] "Non-standalone and standalone: two standard routes to 5G," accessed: 2019-12-05. [Online], please. Available: <https://www.ericsson.com/en/B>

log/20

19/7/standalone-and-non-stand
alone- 5g-two-5g-tracks

[7]3GPP, "Architecture changes
for 5G System (5GS) to support
network data analytics
services," 3rd Generation
Partnership Project

(3GPP), Technical Specification (TS 23.288), Sept. 2019, version

3.[Online]. 16.1.0.

Available:

<https://portal.3gpp.org/desktopmodules/specificationDetails.aspx?specificationId=3579>

[8] 3GPP, System Architecture for the 5G System (5GS), 3rd Generation Partnership Project (3GPP), Technical Specification (TS 23.501), Sept. 2019, version 16.2.0.

[On-line]. Available:

<https://portal.3gpp.org/desktopmodules/specificationDetails.aspx?specificationId=3144>

[9] 3GPP, "5G System; Network Data Management Services; Stage 3," 3rd Generation Partnership Project (3GPP), Technical Specification (TS 29.520), September 2019, version

16.1.0. [On line]. Available:

<https://portal.3gpp.org/desktopmodules/specificationDetails.aspx?specificationId=3355>

[10] 3GPP, "5G System; Unified Data Processing Services; Stage 3," 3rd Generation Partnership Project (3GPP), Technical Specification (TS 29.503), Sept. 2019, version 16.1.0. [On line].

Available:

<https://portal.3gpp.org/desktopmodules/specificationDetails.aspx?specificationId=3355>

/specificationDetails.aspx?specific
ation Id=3342

C. Source Code

```
import pandas as pd
import matplotlib.pyplot
as plt import seaborn as
sns
%matplotlib inline

#df=
pd.read_csv('df_after_enc.csv')
ff=pd.read_csv('cleaned_dataset.c
sv')

ff.head()

#to scale import standardizer
from sklearn.preprocessing import StandardScaler

#scaling on the quantitative data
list=['Acc_len', 'Vmail_message', 'Day_min', 'Day_calls', 'Day_charge',
'Eve_mins', 'Eve_calls', 'Eve_charge', 'Night_mins', 'Night_calls',
'Night_charge', 'Int_mins', 'Int_calls', 'Int_charge', 'Custserv_calls']

#list=['State', 'Acc_len', 'Area_code', 'Int_plan', 'Vmail_plan', 'Vmail_message', 'Day_min',
'Day_calls', 'Day_charge', 'Eve_mins',
# 'Eve_calls', 'Eve_charge', 'Night_mins', 'Night_calls',
'Night_charge', '#Int_mins', 'Int_calls', 'Int_charge',
'Custserv_calls']

ff[list] =

StandardScaler().fit_transform(ff[list])

ff.info()

ff.Churn=ff.Churn.astype('int64')

ff.describe()

#FINDING THE CORRELATION bewtween the variables to drop the
redundant columns
data_cor= ff.corr()

data_cor.Churn.sort_values()

#saving file after only scaling
```

```
df_after_scale = ff.to_csv('df_after_scale.csv',index=False)

df =

ff.drop(['Day_charge','Eve_charge','Night_charge','Int_charge'],axis=1)

df.head()
```

```
df.info()
```

```
df=df.to_csv('df_scale_corr.csv',index=False)
```

```
# saved file after scaling and dropping highly correlated variables.
```

```
import pandas
```

```
as pd import
```

```
numpy as np
```

```
from sklearn.linear_model import
```

```
LogisticRegression from sklearn.ensemble import
```

```
RandomForestClassifier from
```

```
imblearn.over_sampling import SMOTE
```

```
from collections import Counter
```

```
from sklearn.model_selection import
```

```
GridSearchCV,RandomizedSearchCV from sklearn.metrics import
```

```
classification_report,confusion_matrix,accuracy_score,recall_score,precision_score,roc  
_curve,roc_auc_score
```

```
from sklearn.model_selection import
```

```
train_test_split import matplotlib.pyplot as plt
```

```
df=
```

```
pd.read_csv('cleaned_dataset.csv')
```

```
df.head()
```

```
df.columns
```

```
df =
```

```
df.astype('float')
```

```
df.info()
```

```
X =
```

```
df.drop('Churn',axis=1)
```

```
y=df.Churn
```

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state=10)
```

```
print(X_train.shape)
```

```
print(X_test.shape)
```

```
print(Counter(y_train))  
print(Counter(y_test))
```

```
sm = SMOTE()  
X_train_sm,y_train_sm = sm.fit_sample(X_train,y_train)
```

```

# now the chuners and non chuners ratio is same in the output
print(Counter(y_train_sm))
print(Counter(y_test))

model =
RandomForestClassifier(n_estimators=9,
max_features='sqrt',
max_depth=9
,
criterion='gini
',
bootstrap=False)

model.fit(X_train_sm,y_train_sm)

y_pred_rf= model.predict(X_test)

print(classification_report(y_test,y_pred_

rf))

# got a recall of 82% and precison of 80% at a particular trial

accuracy_score(y_test,y_pred_rf)

confusion_matrix(y_test,y_pred_rf)

params=

{'n_estimators':np.arange(1,10),

'criterion': ['entropy', 'gini'],
'max_features':['auto','sqrt','log2'],
'bootstrap':[True,False]
}

model_cv=RandomizedSearchCV(RandomForestClassifier(),params)
#model_cv =GridSearchCV(RandomForestClassifier(),params)

model_cv.fit(X_train_sm,y_train_sm)

model_cv.best_params_

model_cv.best_estimator_

#checking for the ROC - CHECK FOR EFFICIENCY OF THE MODEL

```

```
model.predict_proba(X_test)
```

```
#EXTRACTING ONLY the 2nd column values as it the predict proba values for the  
churners
```

```
y_pred_prob=model.predict_proba(X_test)[:,1]
```

```
print(y_pred_prob)
```

```

fpr, tpr , thresholds = roc_curve(y_test,y_pred_prob)

plt.plot(fpr,tpr)
plt.xlabel("False positive
rate") plt.ylabel("True
positive rate")
plt.plot([0,1],[0,1],"--")
plt.show()

## The default probabilty threshold is 50%

# checking the area under the curve - for a default random model AUC -
0.5 #So, we make sure we get better AUC score for a better performing
model

auc =

roc_auc_score(y_test,y_pred_prob)

print(auc)

importances=
model.feature_importances_
sort=np.argsort(importances)
label=X.columns[sort]

X.shape
ss=
range(X.shape[1])
print(ss)

plt.barh(ss,importances[sort],tick_label=label)

import joblib

joblib.dump(model,'Telecom_trained_model.m

l')

#to load the trained model
model=joblib.load('Telecom_trained_model.ml')

import pandas as pd
import matplotlib.pyplot
as plt import seaborn as
sns
%matplotlib inline

```

#since the output variable is Goal- churn or no churn TRIAL AND ERROR between
logistic regression ,SVM ,Randomforest

```
from sklearn.linear_model import  
LogisticRegression from sklearn.svm import SVC  
from sklearn.ensemble import  
RandomForestClassifier from  
sklearn.model_selection import train_test_split
```

```

from collections import Counter
from imblearn.over_sampling import
SMOTE from sklearn.metrics import
accuracy_score,classification_report,confusion_matrix,recall_score,precision_s

core df= pd.read_csv('df_after_scale.csv')

X =

df.drop('Churn',axis=1)

X.head()

y=df.Churn
X_train,X_test,y_train,y_test =
train_test_split(X,y,test_size=0.3,random_state=10) print(X_train.shape)
print(X_test.shape)

print(Counter(y_train))
print(Counter(y_test))

#### SMOTING REQUIRED -since the ratio of churners and non-churners is not

equal sm = SMOTE()

X_train_sm,y_train_sm = sm.fit_sample(X_train,y_train)

# now the chuners and non chuners ratio is same in the output
print(Counter(y_train_sm))
print(Counter(y_test))

clf=LogisticRegression()
clf.fit(X_train_sm,y_train_sm)

y_pred=clf.predict(X_test)

#checking accuracy score

accuracy_score(y_test,y_pr

ed)

recall_score(y_test,y_pred)

precision_score(y_test,y_pred)

```

```
confusion_matrix(y_test,y_pred)
```

```
pd.crosstab(y_test,y_pred)
```

```
print(classification_report(y_test,y_pred))
```

```
##For the LR though the recall is 78% the precision score is very less
```

```
#Support vector Machine
```

```
svc= SVC()
```

```
svc.fit(X_train_sm,y_train_sm)
```

```
y_pred_svc= svc.predict(X_test)
```

```
accuracy_score(y_test,y_pred_s
```

```
vc)
```

```
print(classification_report(y_test,y_pred_svc))
```

```
confusion_matrix(y_test,y_pred_svc)
```

```
pd.crosstab(y_test,y_pred)
```

```
#Random forest
```

```
model = RandomForestClassifier(random_state=5,n_estimators=100,max_depth=15)
```

```
model.fit(X_train_sm,y_train_sm)
```

```
model.feature_importances_
```

```
y_pred_rf=
```

```
model.predict(X_test)
```

```
print(classification_report(y_test,y_pred_rf))
```

```
accuracy_score(y_test,y_pred_rf)
```

```
confusion_matrix(y_test,y_pred_rf)
```

```
##Random forest gives best results for all the metrics as compared to LR and SVM
```

```
import pandas as pd
```

```

import matplotlib.pyplot
as plt import seaborn as
sns
%matplotlib inline

#importing dataset after cleaning and encoding the categorical
variables df=pd.read_csv('cleaned_dataset.csv')

df.head()

#analysing data

like

#GOAL : 2 classes Churners and non-Churners
#which features are the driving factors for
churn Do churners call customer service
more?

#taking a part of the dataframe and analysing
ff=df.loc[:,['Custserv_calls','Vmail_message','Int_plan','Ch
urn']] #using groupby()
#df.groupby(['Churn']).mea
n() ff.head()

ff.groupby(['Churn']).std()

ff.groupby(['Churn']).mean()
#the mean and std of customer services calls for the churners is high compared to
non- churners

#Churning in some area_codes is more compared to
others-- gg=pd.read_csv('df_after_enc.csv')
gg.groupby('Area_code')['Churn'].value_counts()

#EDA with
visualization
df.head()

#checking the distribution of variables-normal distrubition(-1.96 to
+1.96) from scipy import stats

list=['Acc_len', 'Area_code','Vmail_message', 'Day_min', 'Day_calls',
'Day_charge', 'Eve_mins',
'Eve_calls', 'Eve_charge', 'Night_mins', 'Night_calls', 'Night_charge',

```

```
    'Int_mins', 'Int_calls', 'Int_charge', 'Custserv_calls']  
for i in list:  
    print('skewness and Kurtosis for '+ str(i)+' : '+str(stats.skew(df[i]))+'  
and '+ str(stats.kurtosis(df[i])))
```

#conclusion: we can see the Normality for International calls is not

satisfied #checking the acclength distribution- it is normal

```
sns.distplot([df.Acc_len])
```

#whereas for intcalls it is

not

```
sns.distplot([df.Int_calls])
```

#Checking Acc_len for churners and non churners

#comparing the results there is not much
difference

```
sns.boxplot(x='Churn',y='Acc_len',data=df,sym  
="")
```

<matplotlib.axes._subplots.AxesSubplot at 0x1a24ea79b0>

```
sns.boxplot(x='Churn',y='Acc_len',data=df,sym="",hue='Int_plan')
```

#Comapring churn with customer service calls and also with international

```
plan sns.boxplot(x='Churn',y='Custserv_calls',data=df)
```

#we can see that churners call customer service more

#checking with international plan now

```
sns.boxplot(x='Churn',y='Custserv_calls',data=df,hue='Int_plan',sym="")
```

##conclusion : Customer who churn call customer service more unless they have
opted for international plan they do lesser number of calls

Comparing churn with customer service calls and voice mail plans

```
sns.boxplot(x='Churn',y='Custserv_calls',data=df,hue='Vmail_plan')
```

##Comparing churn with customer service calls and area code

```
sns.boxplot(x='Churn',y='Custserv_calls',data=df,hue='Area_code',sy
```

```
m="") df.columns
```

```
#comparing churn with minutes
sns.boxplot(x='Churn',y='Day_min',data=df,sym="")

sns.boxplot(x='Churn',y='Eve_mins',data=df,sym="")

sns.boxplot(x='Churn',y='Night_mins',data=df,sym="")
```

```
db_host = '18.136.56.185'
username = 'dm_team3'
user_pass =
'dm_team15119#'
db_name =
'project_telecom'

conn =
create_engine('mysql+pymysql://'+username+'_'+user_pass+'@'+db_host+'/'+db_name)
```

```
sql = 'select * from telecom_churn_data'
```

```
df =
```

```
pd.read_sql(sql,conn)
```

```
df.head()
```

```
#convereted all the columns names to valid names from the pdf provided
df.columns=['State', 'Acc_len', 'Area_code', 'Phone', 'Int_plan', 'Vmail_plan',
            'Vmail_message', 'Day_min', 'Day_calls', 'Day_charge',
            'Eve_mins', 'Eve_calls', 'Eve_charge', 'Night_mins',
            'Night_calls', 'Night_charge', 'Int_mins', 'Int_calls', 'Int_charge',
            'Custserv_calls', 'Churn']
```

```
df.info()
```

```
df.head()
print(df.sha
pe)
```

```
!pip install
pymysql import
pandas as pd
import seaborn as
sns
```

```
import matplotlib.pyplot as plt
%matplotlib inline
from sqlalchemy import create_engine
```

```
#Checking if any values in the variables are
null df.isnull().sum()
```

```
#convert the variable data types to valid data
```

```
types df.columns
```

```
#creating a list except the elements which are
```

```
catergorical[state,Int_plan,Phone,Vmail_plan,Churn]
```

```
list = ['Acc_len', 'Area_code','Vmail_message', 'Day_min', 'Day_calls',  
        'Day_charge', 'Eve_mins',  
        'Eve_calls', 'Eve_charge', 'Night_mins', 'Night_calls', 'Night_charge',  
        'Int_mins', 'Int_calls', 'Int_charge', 'Custserv_calls']
```

```
#ff['Acc_len']=
```

```
pd.to_numeric(ff.Acc_len,errors='coerce') df[list]=
```

```
df[list].apply(pd.to_numeric,errors='coerce')
```

```
df.info()
```

```
#dropping phone field -not required since it is
```

```
unique df.drop(columns='Phone',inplace=True)
```

```
print(df.shape)
```

```
#saving the file before proceeding further
```

```
df_bfre_enc=
```

```
df.to_csv('df_bfre_enc.csv',index=False)
```

```
#Encoding the columns values of international plan and voice mail and churn  
using label encoder
```

```
from sklearn.preprocessing import
```

```
LabelEncoder
```

```
list1=['Int_plan','Vmail_plan','Churn']
```

```
encoder = LabelEncoder()
```

```
for i in list1:
```

```
    df[i]=encoder.fit_transform(df[i])
```

```
df.info()
```

```
df.head()
```

```
df_after_enc=df.to_csv('df_after_enc.csv',index=False)
```

a column like state needs to be done one hot encoding since the values in it are huge and each value is unique -learn about one hot encoding

```
df.State.value_counts()
```

```
#plt.bar(df.State,height=0.8,width=0.6) #plt.show()
```

#to reduce the reduce complexity using the pandas method to convert categorical variable into dummy/indicator variables with the get_dummies function

```
df.State=pd.get_dummies(df.State,dtype='int64')  
df.State.value_counts()
```

```
#df.Area_code.value_counts()
```

```
cleaned =  
df.to_csv('cleaned_dataset.csv',index=False)
```