College of Excellence, nirf 2022 - 6 th Rank

Autonomous and Affiliated to Bharathiar University

Reaccredited with A++ grade by NAAC, An ISO 9001:2015 Certified Institution
Peelamedu, Coimbatore – 641004

WEB TRAFFIC PREDICTION USING TIME SERIES

Internship work submitted to PSGR Krishnammal College for women in partial fulfilment of the requirements for the award of the degree of

Master of Science in Data Analytics

Bharathiar University, Coimbatore – 641046

Done By

KOUSHIKA RG

(21MDA021)

Guided by

Dr.ALBINAA.T.A M.Sc,M.Phil.,Ph.D.,

Assistant Professor, Department of Data Analytics (PG)

DEPARTMENT OF DATA ANALYTICS (PG)

PSGR KRISHNAMMAL COLLEGE FOR WOMEN

Peelamedu, Coimbatore – 641 004.

www.psgrkcw.ac.in

SEPTEMBER 2022

CERTIFICATE

This is to certify that this internship work entitled **"WEB TRAFFIC PREDICTION USING TIME SERIES"**submitted to PSGR Krishnammal College For Women,Coimbatore in partial fulfillment of the requirements for the award of Master of Science in Data Analytics is a record of original work done by **KOUSHIKA RG (21MDA021)** during her period of study in the Department of Data Analytics(PG), PSGR Krishnammal College for Women, Coimbatore under my Supervision and guidance and the internship work has not formed the basis for the award of any other Degree / Diploma / Associateship / Fellowship or any similar title to any candidate of any University.

Forwarded by

Dr. ALBINAA.T.A, M.Sc.,M.Phil.,Ph.D.,	**Dr. N.RADHA, M.Sc., M.Phil., PhD.,**
Faculty Guide	**Head of the Department**

DECLARATION

I hereby declare that this internship work entitled **"WEB TRAFFIC PREDICTION USING TIME SERIES"** submitted to PSGR Krishnammal College For Women, Coimbatore for the award of theDegree of Master of Science in Data Analytics, is a record of original work done by **KOUSHIKA RG(21MDA021)**under the supervision and guidance of **Dr.ALBINAA.T.A,M.Sc.,M.Phil.,Ph.D.,** Assistant Professor, Department of Data Analytics(PG), PSGR. Krishnammal College for Women, Coimbatore and that this internship work has not formedthe basis for the award of any other Degree / Diploma / Associateship / Fellowship or anysimilar title to any candidate of any University.

KOUSHIKA RG

(21MDA021)

Endorsed by

Place: Coimbatore
Date:

Dr.ALBINAA.T.A,M.Sc,M.Phil.,Ph.D.,
Faculty Guide

CONTENTS

ACKNOWLEDGEMENT

"Success is to be measured not so much by the position that one has reached in life but asby the obstacle which he had overcome while trying to succeed".

I extend my thanks to **Dr.R.Nandhini,** Chairperson, PSGR Krishnammal College for Women,Coimbatore for her full support and for all the resources provided.

I express my whole hearted thanks to **Dr.N.Yesodha Devi M.Com, M.Phil,Ph.D.,** Secretary PSGR Krishnammal College for Women, Coimbatore for having given me the opportunity to undertake this internship work.

I extend my thanks to **Dr.P.Meena,M.Sc.,M.Phil.,Ph.D.,** Principal, PSGR Krishnammal College for Women, Coimbatore for her full support and for all the resources provided.

I am extremely grateful **to Dr.N.Radha M.Sc.,M.Phil.,Ph.D.,** Head, Department of Data Analytics (PG), PSGR Krishnammal College for Women, Coimbatore for her sustainedinterest and advice that have contributed to a great extent to the completion of the internship work.

I wish my indebtedness to my guide **Dr.T.A.ALBINAA, M.Sc.,M.Phil.,Ph.D.,** AssistantProfessor, Department of Data Analytics (PG), PSGR Krishnammal College for Women, Coimbatore for her appropriate guidance, suggestions and support in the completion of this internship work.

I extend my sincere thanks to **GAGAN SINGH** Project Manager, CYBERICSOFT Technologies Noida, for his kind support and contribution to the successful completion of the internship work.

My sincere thanks to all my staff of Department of Data Analytics (PG) for their timely supportand encouragement.

Finally, I place on record my deep sense of gratitude to my beloved parents and to my friends for their timely support in completing this internship work.

SYNOPSIS

This internship entitled “WEB TRAFFIC PREDICTION”. Now days, web traffic anticipating is a significant issue as this can make misfortunes the activities of major sites. Time arrangement topicshas been an interesting issue for research. Anticipating future time arrangement esteems is one of the most troublesome issues in the business. The time arrangement field includes various issues, running from induction and examination to gauging and grouping. Estimating the organization traffic and updates continuously would be the most productive approach to pass on the data These days, we are excessively reliant on Google worker however in the event that we need to have a worker for huge clients we mighthave anticipated the quantity of clients from earlier years to stay away from worker breakdown. Thesedays, web traffic anticipating is a significant issue as this can make misfortunes the activities of majorsites. Time-arrangement gauging has been an interesting issue for research. Anticipating future time arrangement esteems is one of the most troublesome issues in the business. Estimating the organization traffic and updates continuously would be the most productive approach to pass on the data. In this project, we apply a forecasting model for the purpose of predicting web traffic. Predicting web traffic can help web site owners in many ways including: (a) determining an effective strategy for load balancing of web pages residing in the cloud, (b) forecasting future trends based on historical data and (c) understanding the user behavior.

1.1 INTRODUCTION

It is fundamental for information researchers and business investigator to acquire time arrangement insightful abilities. Time arrangement data set had been the quickest developing class of data sets in the previous two years, and both customary ventures and arising innovation businesses had been creating additional time arrangement information. A few instances of time arrangement information bases are the monetary market data set, climate estimating data set, smart home monitoring database, and supply chain monitoring database. Itis essential for data analysts and business agent to obtain time course of action keen capacities. Time plan informational collection had been the speediest creating class of informational indexes in the past two years, and both standard endeavors and emerging development organizations had been making extra time course of action data. A couple of examples of time course of action data bases are the money related market informational index, environment assessing informational collection, sharp home checking informational collection, and stock organization noticing informational index.

Recently, more and more people are getting access to the internet all over the world, the rise in traffic for almost all websites are inevitable. The increase in traffic for the websites could cause a lot of problems and the company which manages to cope with the traffic changes in the most efficient way is going to succeed. As mostof the people may have encountered a crashed site or very slow loading time for a website when there is a lot of people using it, like when various shopping websites may crash just before festivals as more people try to log into the website than it was originally capable of which causes a lot of inconveniences for the users and as aresult of that it could decrease the user's ratings of the site and instead use another site, therefore, reducing their business. Therefore, a traffic management technique or plan should be put in place to reduce the risk of such mishaps which could be detrimental to the existence of the company. there wasn't an essential for such tools as most servers could handle the traffic inscription but the smartphone age has enlarged the ultimatum to such a high level for some websites that companies could not have reacted immediately enough to continue their orderly customer service level. Evaluating web traffic on a web server is highly essential for web service providers since, without a conventional dictate forecast, customers could have lengthy bide one's time and spontaneity that website. Nevertheless, this is a backbreaker task since it essential to make dependable predictions based on the arbitrary nature of human behavior. We bring out an architecture that gathered source data and in a supervised way executes the forecasting of the time series of the website

1.1 ORGANIZATION PROFILE

Cybricsoft Technologies Pvt. Ltd. Located in Noida, Uttar Pradesh. The company specializes in software development, Data Science, Artificial Intelligence, and Machine Learning across a full range of technologies, from front-end prototyping to a complete set of back-end services. Our software developers have the extensive understanding and experience necessary to create full-fledged applications for your business. Cybricsoft is the partner of choice for many of the world's leading enterprises, SMEs and technology challengers. they help businesses elevate their value through custom software development, product design, QA and consultancy services. Cybricsoft looks into future to be one stop solution and service provider for all IT needs. They provide Business Intelligence & Analytics services help enterprises understand current trends, predict the future accurately and analyze & combat risks, well in advance.

Cybricsoft follows complete Software Development Lifecycle (SDLC) using Agile Methodology Brainstorming Feasibility Analysis – Design – Programming – Integration – Quality Assurance-Release is the integral part of ever project. Word Wide. They offer simple solutions to complex problems and addressing client key business and technology challenges

1.2 PROBLEM STATEMENT

Web traffic is the amount of data sent and received by visitors to a website and it has been the largest portion of website traffic. Traffic flow prediction heavily depends on historical and real-time traffic data collected from various internet flow monitoring sources. Traffic management and control driven by big data is becoming a new trend. This inspires us to reconsider the traffic flow prediction model based on deep architecture models with such rich amount of traffic data.

The problem of forecasting the future values of time series has always been one of the most challenging problems in the field. These data visualizations offer a combination of historic data and real-time information that is useful for identifying emerging trends and monitoring efficiency. Real time dashboards usually contain data that is time-sensitive.

The main objective is to predict the web traffic website data used for this project is from Analytics blog. That data contains hour index and sessions The data is returned in csv format. The web traffic is basically the number of sessions in a given time frame, and it varies a lot with respect to time of the day, week and so on, and how much web traffic of platform can withstand depends on the size of the servers that are supporting the platform. Basically, forecasting the web traffic or a number of sessions based on the historical data.

1.3 TOOL DESCRIPTION

PYTHON

Python is a highly interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990.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 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.

FEATURES

- 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-maintaining.
- A broad standard library − Python's bulk of the library is very portable and cross platform compatible on UNIX, Windows, and Macintosh.
- Interactive Mode − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- Portable − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- Extendable − Can also add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- Databases − Python provides interfaces to all major commercial databases.
- 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.

APPLICATIONS OF PYTHON

GUI-Based Desktop Applications

Python has simple syntax, modular architecture, rich text processing tools and the ability to work on multiple operating systems which make it a desirable choice for developing desktop-based applications. There are various GUI toolkits like python, PyQt or PyGtk available which help developers create highly functional Graphical User Interface (GUI).

Image Processing and Graphic Design Applications

Python has been used to make 2D imaging software such as Inkscape, GIMP, Paint Shop Pro and Scribus. Further, 3D animation packages, like Blender, 3ds Max, Cinema 4D, Houdini, Lightwave and Maya, also use Python in variable proportions.

Scientific and Computational Applications

The higher speeds, productivity and availability of tools, such as Scientific Python and Numeric Python, have resulted in Python becoming an integral part of applications involved in computation and processing of scientific data. 3D modeling software, such as Free CAD, and finite element method software, such as Abaqus, are coded in Python.

Games

Python has various modules, libraries and platforms that support development of games. For example, PySoy is a 3D game engine supporting Python 3, and PyGame provides functionality and a library for game development. There have been numerous games built using Python including Civilization-IV, Disney's Toon town Online, and Vega Strike.

Enterprise and Business Applications

With features that include special libraries, extensibility, scalability and easily readable syntax, Python is a suitable coding language for customizing larger applications. Reddit, which was originally written in Common Lips, was rewritten in Python in 2005. Python also contributed in a large part to functionality in YouTube.

Operating Systems

Python is often an integral part of Linux distributions. For instance, Ubuntu's Ubiquity Installer, and Fedora's and Red Hat Enterprise Linux's Anaconda Installer are written in Python. Gentoo Linux makes use of Python for Portage, its package management system.

Language Development

Python's design and module architecture has influenced development of numerous languages. Boo language uses an object model, syntax and indentation, similar to Python. Further, syntax of languages like Apple's Swift, Coffee Script, Cobra, and OCaml all share similarity with Python

PACKAGES

NumPy

NumPy is a very popular python library for large multi-dimensional array and matrix processing, with the help of a large collection of high-level mathematical functions. It is very useful for fundamental scientific computations in Machine Learning. In this project, as regression involves some mathematical calculations using NumPy.

Pandas

Pandas is a popular Python library for data analysis. It is not directly related to Machine Learning. As the dataset must be prepared before training. In this case, Pandas comes handy as it was developed specifically for data extraction and preparation. It provides high-level data structures and wide variety tools for data analysis. Here, data selection and data pre-processing are done using pandas.

Matplotlib

Matplotlib is a very popular Python library for data visualization. It is a 2D plotting library used for creating 2D graphs and plots. A module named pyplot makes it easy for programmers for plotting as it provides features to control line styles, font properties, formatting axes, etc. It provides various kinds of graphs and plots for data visualization, viz., histogram, error charts, bar chats, etc. In this project Matplotlib is used for creating visualizations (bar charts).

Scikit-learn

Skikit-learn is one of the most popular ML libraries for classical ML algorithms. Scikit-learn supports most of the supervised and unsupervised learning algorithms. Scikit-learn can also be used for data-mining and data-analysis, which makes it a great tool who is starting out with ML. Here sklearn is used for generating confusion matrix. As mentioned above, the project contains supervised learning algorithms, which have used sklearn.

Seaborn

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Apart from bar chart, the project includes heatmap for visualization.

Stats models

stats models are a Python package that provides a complement to scipy for statistical computations including descriptive statistics and estimation and inference for statistical models.

TensorFlow

TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks

JUPYTER NOTEBOOK

Notebook is an open-source web application that allows creating and sharing codes and documents. It provides an environment, where the code can be documented, run it, look at the outcome, visualize data and see the results without leaving the environment. This makes it a handy tool for performing end to end data science workflows – data cleaning, statistical modelling, building and training machine learning models, visualizing data, and many, many other uses. Jupyter Notebooks really shine even in the prototyping phase. This is because the code is written in independent cells, which are executed individually. This allows the user to test a specific block of code in a project without having to execute the code from the start of the script. Many other IDE environments (like RStudio) also do these in several ways, but it is found that Jupiter's individual cells structure to be the best of the lot. These Notebooks are incredibly flexible, interactive and powerful tools in the hands of a data scientist. They even allow run other languages besides Python, like R, SQL, etc. Since they are more interactive than an IDE platform, they are widely used to display codes in a more pedagogical manner.

GOOGLE COLAB

Colaboratory or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

FEATURES OF GOOGLE COLAB NOTEBOOK

- Google Colab provides tons of exciting features that any modern IDE
- Interactive tutorials to learn machine learning and neural networks.
- Write and execute Python 3 code without having a local setup.
- Execute terminal commands from the Notebook.
- Import datasets from external sources such as Kaggle.
- Save your Notebooks to Google Drive.
- Import Notebooks from Google Drive.
- Free cloud service, GPUs and TPUs.
- Integrate with PyTorch, Tensor Flow, Open CV.
- Import or publish directly from / to GitHub.

2.DOMAIN – WEB TRAFFIC ANALYTICS

Web analytics is the technology and method for the collection, measurement, analysis and reporting of websites and web applications usage data Web analytics has been growing ever since the development of the World Wide Web. It has grown from a simple function of HTTP (Hypertext Transfer Protocol) traffic logging to a more comprehensive suite of usage data tracking, analysis, and reporting.

Log files have been used to keep track of web requests since World Wide Web emerged and the first widely used browser Mosaic was released in 1993. One of the pioneers of web log analysis was Web Trends, a Portland, Oregon based company, which conducted website analytics using data collected from web server logs. In the same year, Web Trends created the first commercial website analytics software. In 1995, Dr. Stephen Turner created Analog, the first free log file analysis software. In 1996, Resistor offered hit counter as a service for websites that would display a banner. Web server logs have some limits in types of data collected. For example, they could not provide information about visitors' screen sizes, user interactions with page elements, mouse events such as clicking and hovering, etc. The new technique of page tagging is able to overcome the limitation and gets more popular recently. The fundamental basis of web analytics is collection and analysis of website usage data. Today, web analytics is used in many industries for different purposes, including traffic monitoring, e-commerce optimization, marketing/advertising, web development, information architecture, website performance improvement, web-based campaigns/programs, etc. Some of the major web analytics usages are:

1. Improving website/application design and user experience. This includes optimizing website information architecture, navigation, content presentation/layout, and user interaction. It also helps to identify user interest/attention areas and improve web application features. A particular example is a heat map that highlights areas of a webpage with higher than average click rate and helps determine if intended link/content is in the right place.
2. Optimizing E-Commerce and improving e-CRM on customer orientation, acquisition and retention. More and more companies analyze website usage data in order to understand customers' needs to increase traffic and ultimately increase their revenue. Different sites can have Manuscript only – published in Encyclopedia of Information Science and Technology, Third Edition, IGI Global different goals like selling more products and attracting more users to generate more income through advertisements. Websites want to keep visitors longer (reducing bounce rate) to encourage users to return and to make every visit end with completion of targeted action (conversion).
3. Tracking and measuring success of actions and programs such as commercial campaigns. To bring value, web analytics must differentiate between a wide variety of traffic sources, marketing

channels, and visitor types. A common question is: “where did visitors learn that information?” For example, parameters used in tracking direct traffic from email, social media, or mobile devices allow correlation of traffic sources with marketing campaign cost, which helps to evaluate return on investments.

4. Identifying problems and improving performance of web applications. The study performed by Tag Man shows a significant correlation between page-load time and the likelihood of a user to convert Web analytics helps to address this issue. Page loading metrics such as average page load time by browser and geographic location are used to measure performance. Both real-time and historical performance analysis allow proactive detection, investigation, and diagnosis of performance issues. Improvements may range from simple image optimization to modification of the expiration date in the HTTP headers to force browsers to use cached website content. A heat map might help to reveal website errors, such as that users click on buttons or images without links. The same techniques can be used by developers of web-based applications and games to add/modify software features. And analyses and reports Meaningful and measurable metrics must be defined in order to analyze web traffic and relate it to business goals. The most common traditional metrics used in web analytics are
    - Visit count: page view, visit, unique visitor.
    - Visit duration: time on page, time on site.
    - Bounce rate and exit rate. The most basic analysis is the dimensional analysis involving measures and dimensions.

The basic metrics mentioned above and other derived metrics are aggregated by dimensions at different levels. For example, we can use dimensional analysis to answer the question: “what are the total visits by month (or day of the week) and by website sections (or page)?" Dimensional analysis is the fundamental piece of other analyses and reports. Most common types of analyses include: Trend analysis looks at data along the time dimension and shows the chronological changes of selected metrics. For example, data can show how the percentage of mobile client access has changed for the past two years. Distribution analysis is about metric value breakdown. Values are usually calculated as percentages of the total by one or more dimensions. It is often used to analyze visitor and client profiles. For example, the percentages of browser types for the past month give information about client diversity. Other commonly used dimensions in this type of analysis are traffic source (e.g. referral source analysis reveals the campaign effectiveness), location, technical data that includes information about browser, OS, device, screen resolution and color depth, client technology support,

etc. User activity or behavior analysis analyzes how users interact with websites. Typical examples are engagement analysis, clickstream analysis, and in-page analysis. Engagement analysis is one of the

most frequently used analyses in the industry. It measures the following factors:

- How many pages were visited per session?
- What is the duration of a visit?
- How often new visitors become returning visitors?
- How often visitors return to the site

There were several attempts to create engagement calculators that will distinguish between user visits. For example, one user came from Google search, visited two pages in five minutes and downloaded necessary document. Another user came from the main site, visited twenty pages in 40 minutes, downloaded five documents Clickstream analysis, also known as click paths, analyzes the navigation path a visitor browsed through a website. A clickstream is a list of all the pages viewed by a visitor presented in the viewing order, also defined as the "succession of mouse clicks" that each visitor makes Clickstream analysis helps to improve the navigation and information architecture of websites. Visitor interest/attention analysis analyzes users' attentions on a web page. It uses client script to track user mouse movements and clicks, and shows results in a heat map. It can also show how far down visitors scroll the page. Analysis of link popularity and areas of attention helps to develop content placement strategies. For example, it helps determine what navigational items should be placed on the top of the page or find the best places for advertisements. Conversion analysis is one of the key analyses in e-commerce and other sectors. Conversion rate is calculated by dividing the number of completed targeted actions by the number of unique users visited the site. All web analytics providers strive to improve conversion tracking. For example, Google Analytics provides Multi-Channel Funnels conversion reports that show what campaigns, sources, or channels have contributed to a visitor's multi-visit conversion. Performance analysis helps reveal website performance issues or linking errors. For example, after a website redesign, indirect traffic volume needs to be watched. If there is less indirect traffic, then some links from other sites and/or bookmarks were potentially broken after the redesign.

Web analytics is a field of web traffic data collection and analysis. It had gained wide adoption and become one of the important tools to help web application management and business analysis. With the recent Web 2.0 and cloud service advancements, it has quickly evolved from simple system level data logging to more comprehensive information collection and analysis. With the continuing expansion of data sources, Web/digital analytics will play an even more important role in the future.

3.DATA MODELLING

The first step is Data gathering. This step is very important because the quality and quantity of the data gathered will directly affect the level of the prediction model. So, this project taken data It is a six-month series data set

3.1 DATASET DESCRIPTION

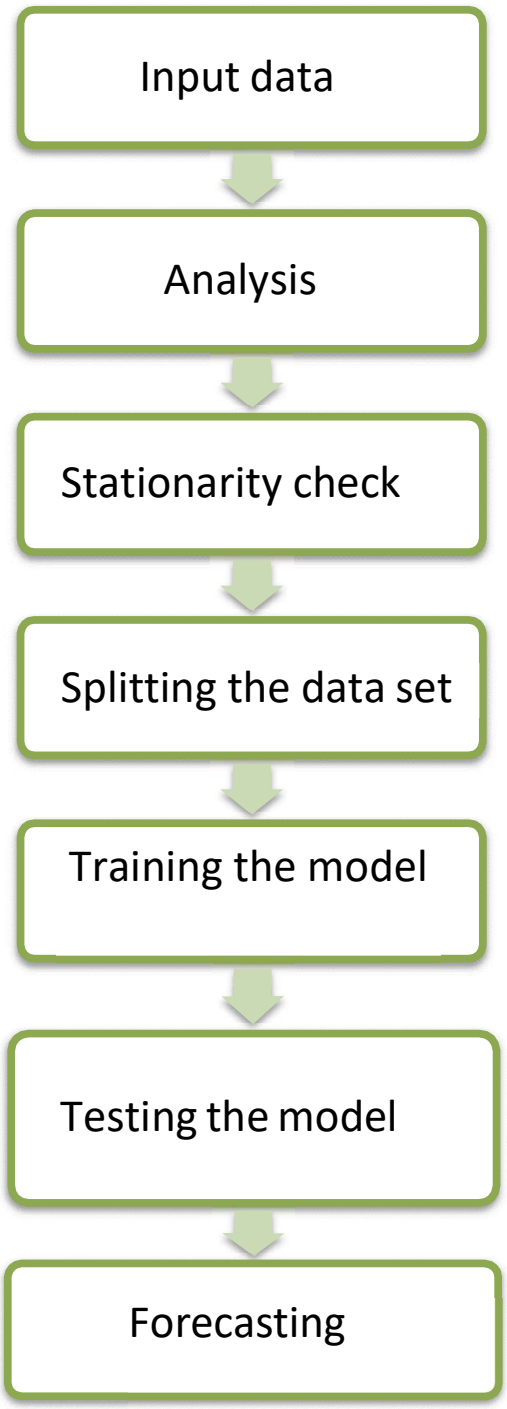
Hour index: The first column is the hours as in this is the first hours, this is the second hour and so on.
Sessions: Session is the volume of traffic at an hourly level
Shape of data: 4896, 2

1 weak data: index [168]

variate of dataset: Univariate data since Use only one variable, cannot use external data, Based only on relationships between past and present

Hour Index	Sessions
0	1418159421
1	1113769116
2	919158921
3	822352824
4	735526737
...	...
163	1732529736
164	1797399801
165	1712569716
166	1721551725
167	1650693654

3.2 SYSTEM FLOW DIAGRAM



3.3 PROCESS FLOW

STATIONARITY

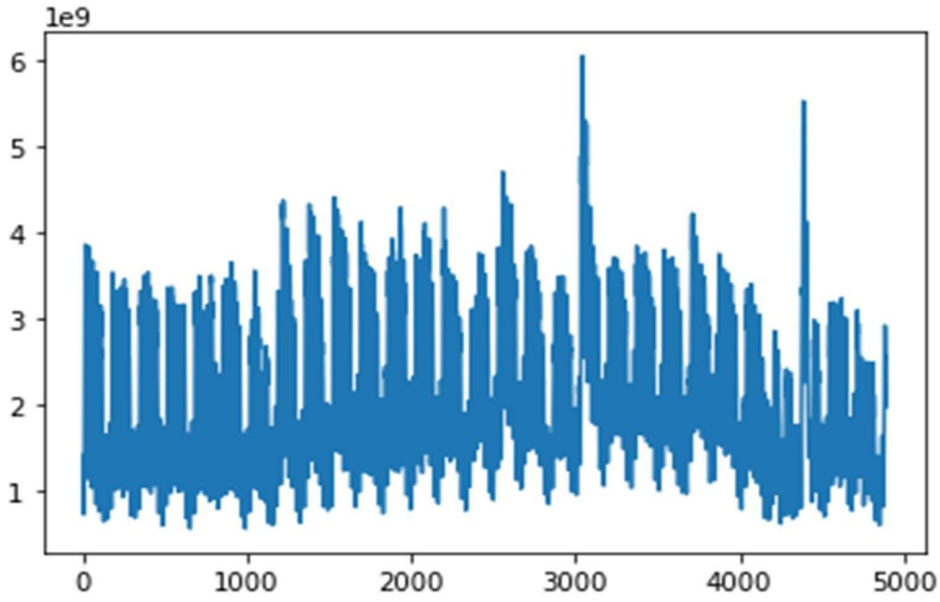
Statistical properties Such as mean, variance remain constant over time. Most of the Timeseries models work on the assumption that the Timeseries is stationary. Timeseries has a particular behaviour over time, there is a very high probability that it will follow the same in the future. Also, the theories related to stationary series are more mature and easier to implement as compared to non-stationary series. Stationarity is defined using very strict criterion. However, for practical purposes we can assume the series to be stationary if it has constant statistical properties over time, i.e. the following:

1. constant mean
2. constant variance
3. an autocovariance that does not depend on time

	Stationarity	**Not Stationarity**
Constant mean	yes	no
Constant variance	yes	no
seasonality	no	yes

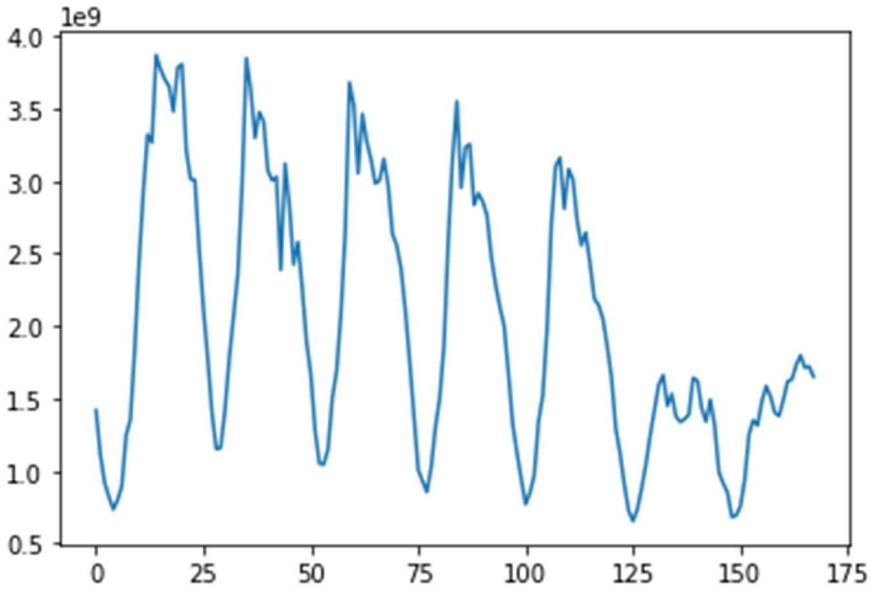
DATA EXPLORATION

Fig 3.3.1



Each point of this curve is an early session count and you can see there are some repeating patterns throughoutthe time series. The traffic volume comes down, after almost equal intervals of time. Apart from that, there are a couple of spikes as well in the traffic, In this plot. Let's explore this data, at a more granular level, we can use the below code and replace the entire time series, with a subset of it.

Fig:3.3.2



Here we are plotting the first week's data only, now the repeating pattern can be seen more clearly, and these dips in the plot in web traffic are may be occurring once every 24 hours. So clearly there are two instances of time in a day, when we have a huge traffic volume, like during a few times and when we have a modest level oftraffic on the website. As in here, I will help you to explore this data as much as possible, before getting started with model building Since by exploring the above plot, we can infer the mean over the plot is same and we have constant variance so we can conclude that it is Stationarity

To check the stationary by statistical methods:

A p-value below a threshold (such as 5% or 1%) suggests we reject the null hypothesis (stationary), otherwise ap-value above the threshold suggests we fail to reject the null hypothesis (non-stationary)

- **p-value > 0.05**: Fail to reject the null hypothesis (H0), the data has a unit root and is non-stationary.
- **p-value <= 0.05**: Reject the null hypothesis (H0), the data does not have a unit root and is stationary

check the stationary or not

In [6]:
```
from statsmodels.tsa.stattools import adfuller
def ad_test(df):
    dftest = adfuller(df, autolag = 'AIC')
    print("2. P-Value : ", dftest[1])
    for key, val in dftest[4].items():
        print("\t",key, ": ", val)
ad_test(df['Sessions'])
```

```
2. P-Value :  3.6325870678651973e-16
         1% :  -3.431695415409747
         5% :  -2.8621345244394583
         10% :  -2.5670864667133415
```

4. MODEL FITTING

I. **STATISTICAL ALGORITHUM:** AUTOREGRESSIVE INTEGRATED MOVING AVERAGE(ARIMA)

II. **MACHINE LEARNING ALGORITHUM:** RANDOM FOREST REGRESSION

III. **DEEP LEARNING ALGORITHUM:** CONVOLUTION NEURAL NETWORK(CNN)

4.1 STATISTICAL ALGORITHUM: AUTOREGRESSIVE INTEGRATED MOVINGAVERAGE (ARIMA)

In an Existing system, ARIMA model have been used. It was introduced by Box and Jenkins (1976). ARIMA stands for Autoregressive Integrated Moving Average models. It is a forecasting technique that predicts the future values of a series based entirely on its own inertia. The essence of the model is that the non-stationary time series sequence by differential transforms method. There are few problems in this system. Some of the problems with traditional time series model are,

- Time interval between data has to be same throughout the data
- Day with NA is not allowed
- Seasonality with multiple periods (Week and Year) is hard to handle
- Parameter tuning by expert is necessary

ARIMA represents auto-backward incorporated moving normal. It is perhaps the most widely recognized and solid models utilized in Time Arrangement expectations. It contains Autoregression (AR), Coordinated (I) and Moving-normal (Mama). Auto regression is "A model that utilizes the reliant connection between a perception and some number of slacked perceptions." Incorporated is "a model that utilizes the differencing of crude perceptions (for examplededucting a perception from the past time step). Differencing in measurements is a change applied to time-arrangement information to make it fixed. “This permits the properties don't rely upon the hour of perception, dispensing with pattern and irregularity and settling the mean of the time arrangement. “Moving-normal is "a model that utilizes the reliance between a perception and a lingering mistake from a moving normal model applied to slacked perceptions. In spite of the AR model, the limited MA model is consistently fixed." At the point when we utilize these three parts in the ARIMA

model, it concocts three boundaries:

1. p (slack request): number of slack perceptions remembered for the model

2. d (level of differencing): number of times that the crude perceptions are differenced

3. q (request of moving normal): size of the moving normal window"

fig:4.1.1

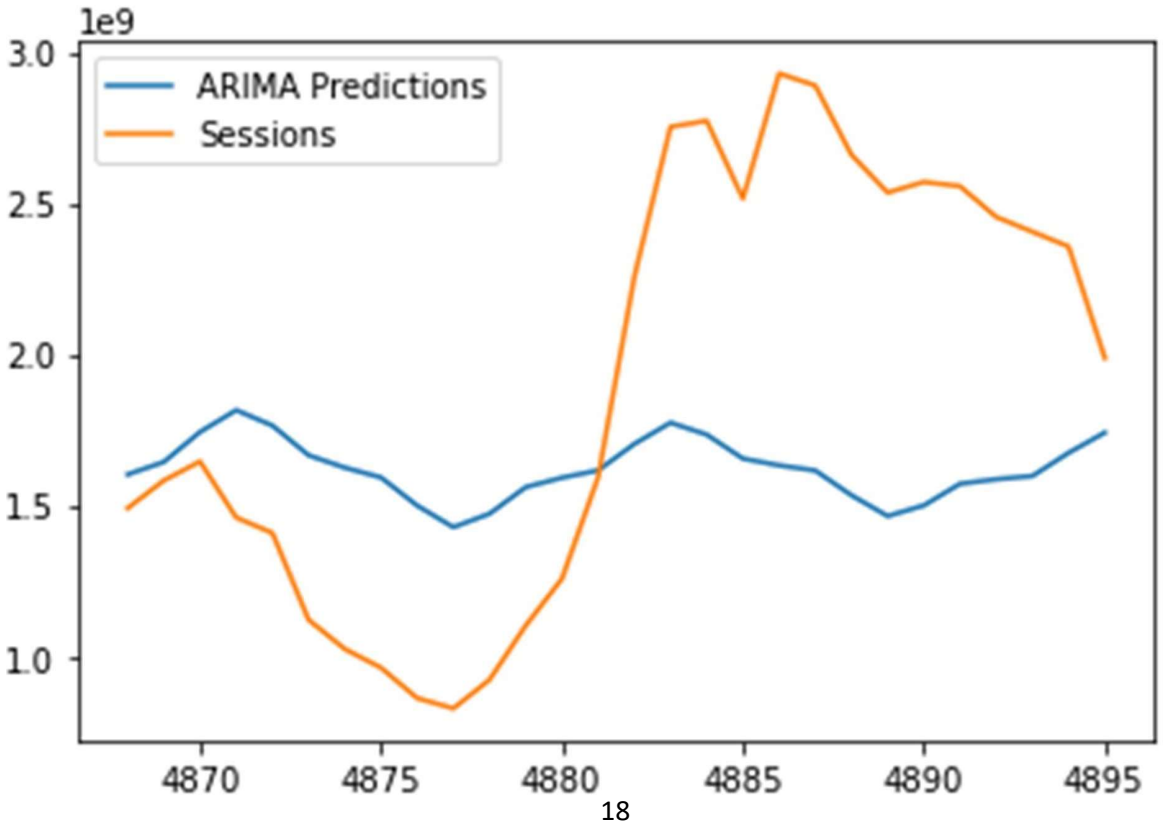
```
In [13]: from pmdarima import auto_arima
         stepwise_fit = auto_arima(df['Sessions'], trace=True,
         suppress_warnings=True)

         Performing stepwise search to minimize aic
          ARIMA(2,1,2)(0,0,0)[0] intercept   : AIC=201746.479, Time=5.14 sec
          ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=203342.692, Time=0.24 sec
          ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=201817.088, Time=0.72 sec
          ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=202091.445, Time=0.68 sec
          ARIMA(0,1,0)(0,0,0)[0]             : AIC=203340.694, Time=0.19 sec
          ARIMA(1,1,2)(0,0,0)[0] intercept   : AIC=201779.582, Time=2.47 sec
          ARIMA(2,1,1)(0,0,0)[0] intercept   : AIC=201809.586, Time=5.19 sec
          ARIMA(3,1,2)(0,0,0)[0] intercept   : AIC=inf, Time=16.75 sec
          ARIMA(2,1,3)(0,0,0)[0] intercept   : AIC=201161.981, Time=4.44 sec
          ARIMA(1,1,3)(0,0,0)[0] intercept   : AIC=201505.268, Time=3.21 sec
          ARIMA(3,1,3)(0,0,0)[0] intercept   : AIC=201163.045, Time=7.05 sec
          ARIMA(2,1,4)(0,0,0)[0] intercept   : AIC=201161.086, Time=6.63 sec
          ARIMA(1,1,4)(0,0,0)[0] intercept   : AIC=201300.013, Time=4.33 sec
          ARIMA(3,1,4)(0,0,0)[0] intercept   : AIC=201159.809, Time=16.77 sec
          ARIMA(4,1,4)(0,0,0)[0] intercept   : AIC=200533.493, Time=24.61 sec
          ARIMA(4,1,3)(0,0,0)[0] intercept   : AIC=200872.012, Time=13.88 sec
          ARIMA(5,1,4)(0,0,0)[0] intercept   : AIC=200772.814, Time=11.03 sec
          ARIMA(4,1,5)(0,0,0)[0] intercept   : AIC=200365.679, Time=28.62 sec
          ARIMA(3,1,5)(0,0,0)[0] intercept   : AIC=200777.262, Time=15.69 sec
          ARIMA(5,1,5)(0,0,0)[0] intercept   : AIC=200442.594, Time=30.64 sec
          ARIMA(4,1,5)(0,0,0)[0]             : AIC=200355.963, Time=25.91 sec
          ARIMA(3,1,5)(0,0,0)[0]             : AIC=inf, Time=18.80 sec
          ARIMA(4,1,4)(0,0,0)[0]             : AIC=200534.140, Time=22.73 sec
          ARIMA(5,1,5)(0,0,0)[0]             : AIC=200458.844, Time=27.85 sec
          ARIMA(3,1,4)(0,0,0)[0]             : AIC=201157.813, Time=11.50 sec
          ARIMA(5,1,4)(0,0,0)[0]             : AIC=200770.537, Time=10.65 sec

         Best model:  ARIMA(4,1,5)(0,0,0)[0]
         Total fit time: 315.763 seconds
```

Fig:4.1.2

```
In [9]:
import statsmodels.api as sm

import warnings
from statsmodels.tools.sm_exceptions import ConvergenceWarning
warnings.simplefilter('ignore', ConvergenceWarning)

In [10]: train = df.iloc[:-28]
test = df.iloc[-28:]
print(test.shape)
print(train.shape)

(28, 2)
(4868, 2)

In [11]: model = sm.tsa.arima.ARIMA(train['Sessions'], order=(4,1,5))
result = model.fit()

In [21]: start=len(train)
end=len(train)+len(test)-1
pred=result.predict(start=start,end=end,typ='levels').rename('ARIMA Predictions')
pred.plot(legend=True)
test['Sessions'].plot(legend=True)

Out[21]: <AxesSubplot:>
```

fig:4.1.1: where the value for p, d, q has found using pmdarima package

Fig:4.1.2: model have been fitted



4.2 MACHINE LEARNING ALGORITHUM: RANDOM FOREST REGRESSION

Random Forest Regression Model:

Random forests were introduced in 2001 by Bierman and are since then one of the most popular algorithms in machine learning the popularity comes from the wide range of applications in which they areknown to perform well on even high dimensional, are fast to compute and easy to tune. Successful applications can be cited: chemo-informatics ecology 3D object recognition and time series predict

Random forests can be related to two main sources, regression trees and bagging Regression trees are constructed by a recursive partitioning of the input space based on some criterion to estimate the regression function f. At each step of the tree construction, a split is selected (a variable and a location on the variable)based on the evaluation of the criterion among all the admissible splits based on all the variables. The cell is cutin two on the selected split and the previous step is reiterated on the new cells. A tree is then a piecewise constant decomposition of the input space. A binary tree can be associated to the input space partitioning. Each node corresponds to a test matching how the input space was cut. An illustration is given in fig. 1 of apartitioning in the two-dimensional space and its associated binary tree. The principle of bagging (short form of bootstrap aggregating) is to create M randomly generated training sets by randomly sampling αn observationswith or without replacement from the set D and to construct on each set a predictor. Once the predictors are constructed, the bagging prediction for a new observation x is an aggregation, generally the empirical mean, of the predictions given by the M predictors for the point x. This procedure aims to improve stability and accuracyof the base predictor. In the context of random forests, the predictors are regression trees. In order to explain the random forest procedure, we then have to explicit the construction of one tree. Random forests taking into account the temporal dependency of the observations and showed that wecan improve significantly the per-formance on forecasting tasks when choosing the right block length. A variantof the variable importance based on the block bootstrap mechanism is also introduced. The non-overlapping variant seems to be mistaken regarding the importance of the variables, forgetting some variables fundamental to the forecasting problem as the hour variable in our first application, and thus we do not advise to use this variant for this purpose. However, both moving and circular variants seem to perform much better than the standard random forests when the block length is well-chosen, and we showed that a good

heuristic for theblock length choice is correlated to a multiple of the smallest seasonality. We will use the sklearn module for training our random forest regression model, specifically the Random Forest Regressor function. The Random Forest Regressor documentation shows many different parameterswe can select for our model. Some of the important parameters are highlighted below

- **n_estimators** — the number of decision trees you will be running in the model
- **criterion** — this variable allows you to select the criterion (loss function) used to determine model outcomes. We can select from loss functions such as mean squared error (MSE) and mean absoluteerror (MAE). The default value is MSE.
- **max_depth** — this sets the maximum possible depth of each tree
- **max_features** — the maximum number of features the model will consider when determining a split
- **bootstrap** — the default value for this is True, meaning the model follows bootstrapping principles(defined earlier)
- **max_samples** — This parameter assumes bootstrapping is set to True, if not, this parameter doesn't apply. In the case of True, this value sets the largest size of each sample for each tree.

4.2.1 DATA SET SPLITTING FOR RANDOM FOREST REGESSOR

```
In [37]: df['A']=df['Sessions'].shift(+1)
         df['B']=df['Sessions'].shift(+2)
         df['c']=df['Sessions'].shift(+3)
         df
```

Out[37]:

	Hour Index	Sessions	A	B	c
0	0	1418159421	NaN	NaN	NaN
1	1	1113769116	1.418159e+09	NaN	NaN
2	2	919158921	1.113769e+09	1.418159e+09	NaN
3	3	822352824	9.191589e+08	1.113769e+09	1.418159e+09
4	4	735526737	8.223528e+08	9.191589e+08	1.113769e+09
...	...	...	...	...	...
4891	4891	2555880561	2.569853e+09	2.534923e+09	2.659673e+09
4892	4892	2454084459	2.555881e+09	2.569853e+09	2.534923e+09
4893	4893	2405182410	2.454084e+09	2.555881e+09	2.569853e+09
4894	4894	2356280361	2.405182e+09	2.454084e+09	2.555881e+09
4895	4895	1987019991	2.356280e+09	2.405182e+09	2.454084e+09

4896 rows × 5 columns

4.2.2 DROP THE NULL VALUES

```
In [38]: df=df.dropna()
         df
```

Out[38]:

	Hour Index	Sessions	A	B	c
3	3	822352824	9.191589e+08	1.113769e+09	1.418159e+09
4	4	735526737	8.223528e+08	9.191589e+08	1.113769e+09
5	5	798400800	7.355267e+08	8.223528e+08	9.191589e+08
6	6	895206897	7.984008e+08	7.355267e+08	8.223528e+08
7	7	1246503249	8.952069e+08	7.984008e+08	7.355267e+08
...	...	...	...	...	...
4891	4891	2555880561	2.569853e+09	2.534923e+09	2.659673e+09
4892	4892	2454084459	2.555881e+09	2.569853e+09	2.534923e+09
4893	4893	2405182410	2.454084e+09	2.555881e+09	2.569853e+09
4894	4894	2356280361	2.405182e+09	2.454084e+09	2.555881e+09
4895	4895	1987019991	2.356280e+09	2.405182e+09	2.454084e+09

4893 rows × 5 columns

4.2.3 MODEL FITTING

```
In [39]: from sklearn.linear_model import LinearRegression
         lin_model=LinearRegression()
```

```
In [40]: from sklearn.ensemble import RandomForestRegressor
         model=RandomForestRegressor(n_estimators=100,max_features=3, random_state=1)
```

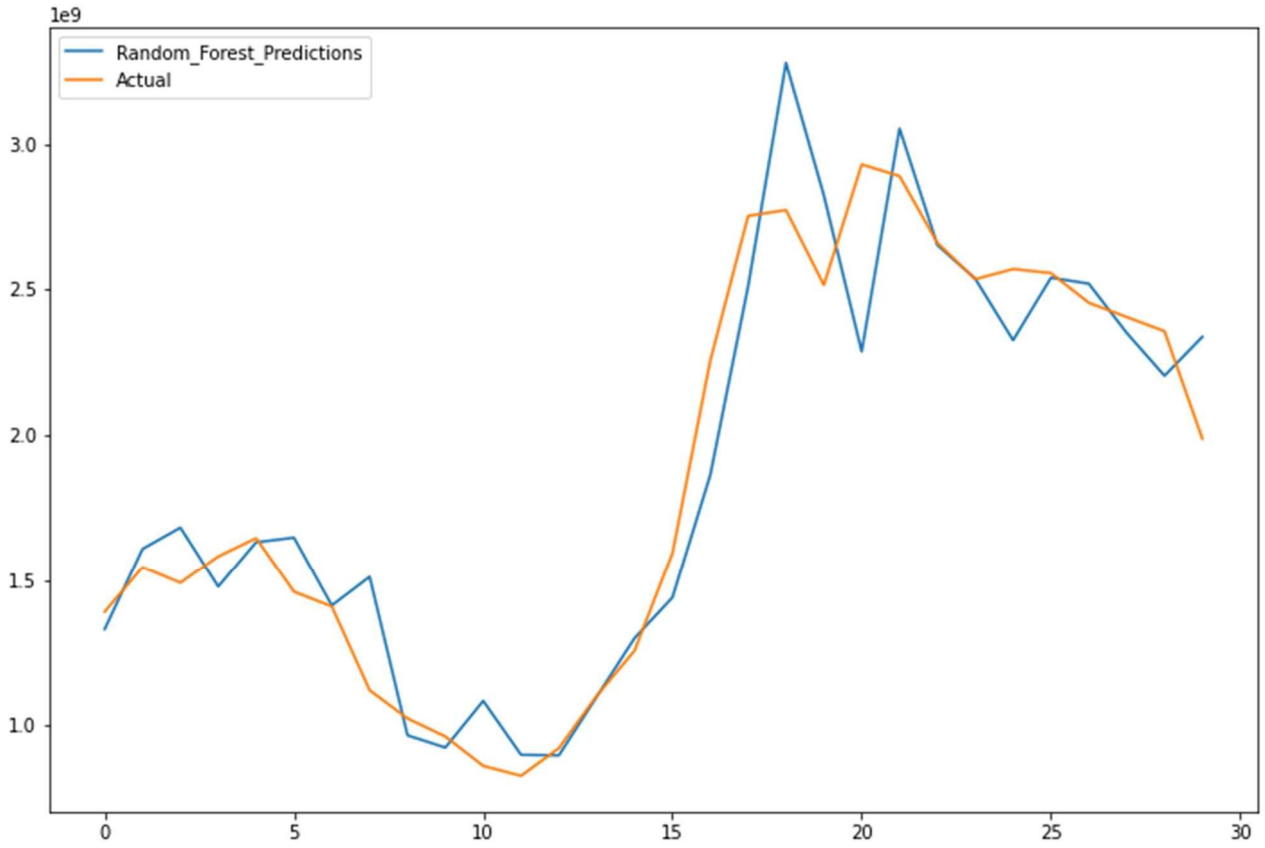
```
In [41]: import numpy as np
         x1,x2,x3,y=df['A'],df['B'],df['c'],df['Sessions']
         x1,x2,x3,y=np.array(x1),np.array(x2),np.array(x3),np.array(y)
         x1,x2,x3,y=x1.reshape(-1,1),x2.reshape(-1,1),x3.reshape(-1,1),y.reshape(-1,1)
         final_x=np.concatenate((x1,x2,x3),axis=1)
         print(final_x)
```

```
[[9.19158921e+08 1.11376912e+09 1.41815942e+09]
 [8.22352824e+08 9.19158921e+08 1.11376912e+09]
 [7.35526737e+08 8.22352824e+08 9.19158921e+08]
 ...
 [2.45408446e+09 2.55588056e+09 2.56985258e+09]
 [2.40518241e+09 2.45408446e+09 2.55588056e+09]
 [2.35628036e+09 2.40518241e+09 2.45408446e+09]]
```

```
In [42]: X_train,X_test,y_train,y_test=final_x[:-30],final_x[-30:],y[:-30],y[-30:]
```

```
In [43]: model.fit(X_train,y_train)
         lin_model.fit(X_train,y_train)
```

4.2.4 RESULT OF RANDOM FOREST REGESSOR



4.3 DEEP LEARNING ALGORITHUM: CONVOLUTION NEURAL NETWORK(CNN)

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are:

- Convolutional layer
- Pooling layer
- Fully-connected (FC) layer

The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer. With each layer, the CNN increases in its complexity, identifying greater portions of the image. Earlier layers focus on simple features, such as colours and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object.

Convolution Layer

The convolution layer is the core building block of the CNN. It carries the main portion of the network's computational load. This layer performs a dot product between two matrices, where one matrix is the set of learnable parameters otherwise known as a kernel, and the other matrix is the restricted portion of the receptive field.

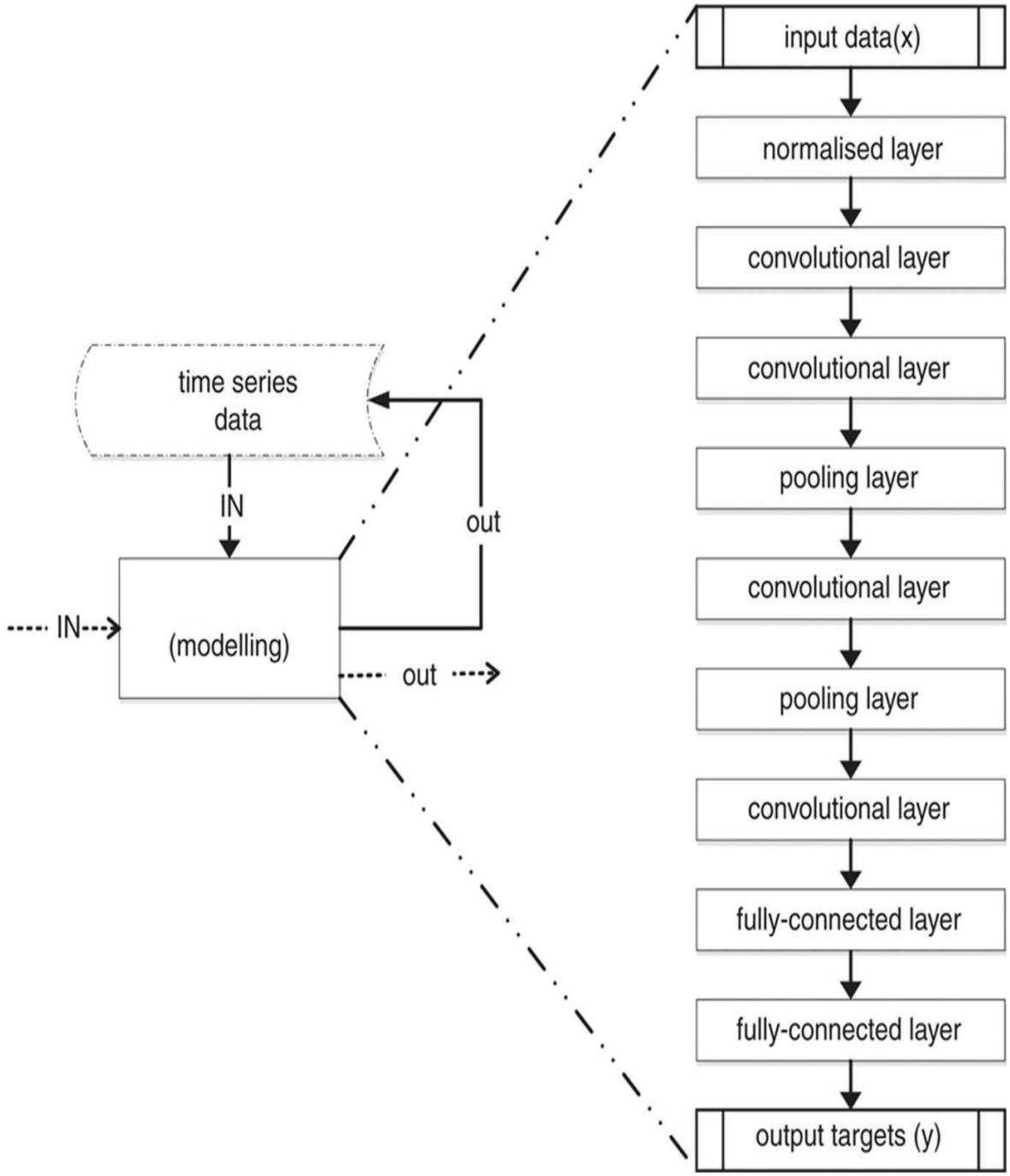
Pooling Layer

The pooling layer replaces the output of the network at certain locations by deriving a summary statistic of the nearby outputs. This helps in reducing the spatial size of the representation, which decreases the required amount of computation and weights. The pooling operation is processed on every slice of the representation individually.

Fully Connected Layer

Neurons in this layer have full connectivity with all neurons in the preceding and succeeding layer as seen in regular FCNN. This is why it can be computed as usual by a matrix multiplication followed by a bias effect. The FC layer helps to map the representation between the input and the output.

4.3.1 WORK FLOW OF CONVOLUTION NEURAL NETWORK



4.3.2 TRAINING THE CNN MODEL

```
[ ] from tensorflow.keras.models import Sequential
    from tensorflow.keras.layers import *
    from tensorflow.keras.callbacks import *
```

```
[ ] model= Sequential()
```

```
[ ] model.add(Conv1D(64, 3, padding='same', activation='relu',input_shape=(num,1)))
```

```
[ ] model.add(Conv1D(32, 5, padding='same', activation='relu',input_shape=(num,1)))
```

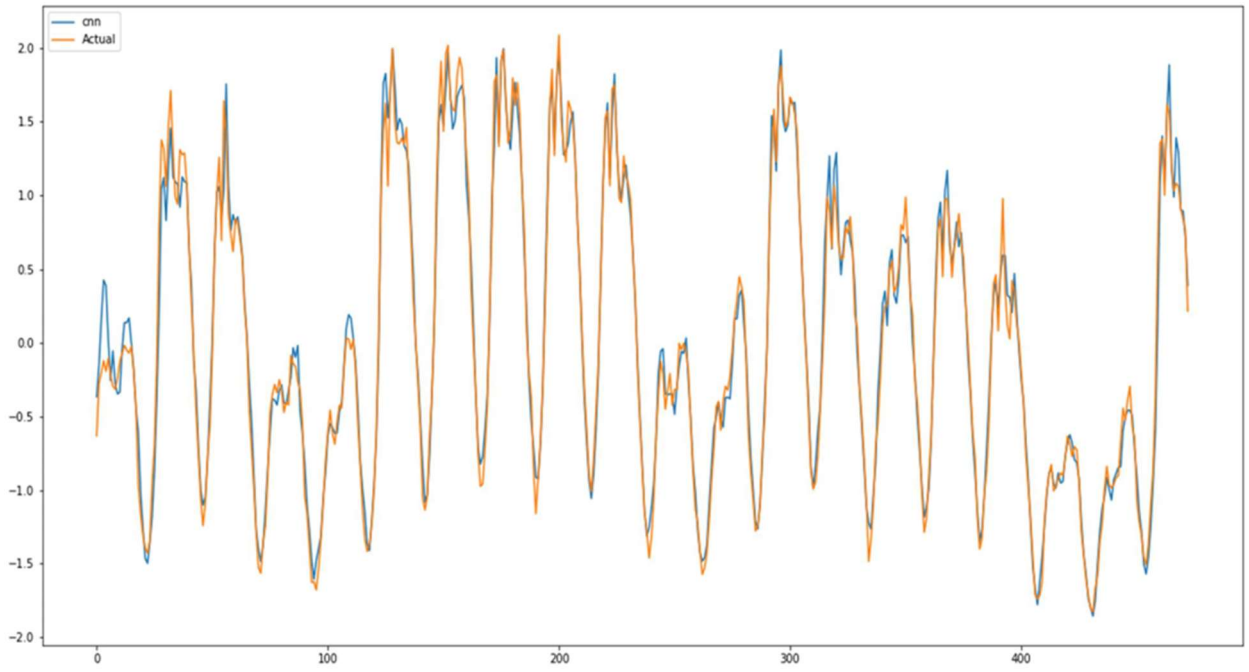
```
[ ] model.add(Flatten())
```

```
[ ] model.add(Dense(64,activation='relu'))
    model.add(Dense(1,activation='linear'))
```

```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d (Conv1D)              (None, 168, 64)           256

conv1d_1 (Conv1D)            (None, 168, 32)           10272

flatten (Flatten)            (None, 5376)              0

dense (Dense)                (None, 64)                344128

dense_1 (Dense)              (None, 1)                 65
```

4.3.3 RESULT OF CNN



5.MODEL EVALUATION

MEAN ABSOLUTE PERCENTAGE ERROR:

The mean absolute percentage error (MAPE) also called the mean absolute percentage deviation (MAPD) measures accuracy of a forecast system. It measures this accuracy as a percentage, and can be calculated as the average absolute percent error for each time period minus actual values divided by actual values.

The mean absolute percentage error (MAPE) is the most common measure used to forecast error, probably because the variable's units are scaled to percentage units, which makes it easier to understand

$$\mathrm{M} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

Where:

- n is the number of fitted points,
- At is the actual value,
- Ft is the forecast value.
- Σ is summation notation (the absolute value is summed for every forecasted point in time).

1. MAPE OF ARIMAMODEL: 0.36095577434567166

2. MAPE OF RANDOM FOREST: 0.0868972023805521

3. MAPE OF CNN: 0.005467832646

6.CONCLUSION

Web traffic Time series prediction can be carried out using Autoregressive integrated moving average Random Forest regressor and convolution neural network effectively However, the CNN-based model still has the advantage of speed. Our system can be used across all websites for improving their web traffic load management and business analysis brings more efficiency to our system. Our system effectively captures seasonal patterns and long-term trends Including highs and lows. In future Prediction of the number of users will access the website is possible. The proposed will keep on improving as more user data is fed. Time Series Forecasting is one of the least explored areas and various models are evaluated to improve the accuracy of the forecast. The main focus of the proposal is to predict future web traffic to make decisions for better congestion control. Past Values are considered to predict future values. will also seek to explore multivariate time series and offer suggestions for simplifying the decision-making process in real-time.

5.BIBLIOGRAPHY

REFERENCES:

[1] Hong Suk Yi, Heian Jung, Sangho on Bae "Deep Neural Networks for Traffic Flow Prediction "in IEEE 978-1-5090-3015-6/17.

[2] Sandhya K et al. / International Research Journal of Multidisciplinary Tec novation /2019, 1(1), 56-63

[3] Jun Lv, Xing Li, Tran Quang Anh, Tong Li "A New Algorithm for Network Traffic

[4] Prediction" in Proceedings of the 11th IEEE Symposium on Computers and Communications, 0-7695-2588-1/06.

[5] Weitao Wang, Yuebin Bai, Chao Yu, Yuhao Gu, Peng Feng, Xiaojing Wang, and Rui Wang, "A Network Traffic Flow Prediction with Deep Learning Approach for Largescale Metropolitan AreaNetwork" in IEEE, 978-1-5386-3416-5/18.

[6] Hao Yin, Chuang Lin, Berton Sebastien, Network traffic prediction based on a new time seriesmodel in International Journal of communication systems, 2005

REFERENCES WEBSITES:

1. https://www.bigcommerce.com/
2. https://en.wikipedia.org/wiki/Web_traffic
3. https://www.semrush.com/analytics/traffic/
4. https://seranking.com/website-traffic-checker.html
5. https://study.com/academy/lesson/what-is-web-traffic