A DISSERTATION ON

A MODEL FOR PREDICTION OF NIFTY INDEX USING TECHNICAL INDICATORS

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

MASTER OF ENGINEERING (DATA SCIENCE)

SUBMITTED BY

Pravin Bharat Jadhav **Exam No: 77000378J**

Under The Guidance of

Prof. K. C. Waghmare

DEPARTMENT OF COMPUTER ENGINEERING
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
No. 27, Near Trimurti Chowk, Dhankawadi
Pune-411043

(2021-2022)

DEPARTMENT OF COMPUTER ENGINEERING
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
No. 27, Near Trimurti Chowk, Dhankawadi
Pune-411043
2021-2022

CERTIFICATE

This is to certify that the dissertation report entitled

A MODEL FOR PREDICTION OF NIFTY INDEX USING TECHNICAL INDICATORS

Submitted By

Pravin Bharat Jadhav **Exam No:77000378J**

is a work carried out by him under the supervision of **Prof. K. C. Waghmare** and it is submitted towards the partial fulfillment of the requirement of Savitribai Phule Pune University, Pune for the award of the degree of **Master of Engineering (Data Science)** in academic year 2021-22.

(Prof. K. C. Waghmare)	**(Dr. Geetanjali V. Kale)**	**(Dr. R. Sreemathy)**
Guide	Head	Principal
Department of Computer Engineering	Department of Computer Engineering	Pune Institute of Computer Technology

Place: Pune.
Date:

Acknowledgment

It gives me great pleasure and satisfaction in presenting this final report on "**A Model For Prediction of Nifty Index Using Technical Indicators**".

I thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of [Computer Dept] which helped me in successfully completing my project work. Also, I would like to extend my sincere esteems to all staff in laboratory for their timely support.

I have furthermore to thank Computer Department HOD **Dr. Geetanjali V. Kale** and Guide **Prof. K. C.Waghmare** to encourage me to go ahead and for continuous guidance.I am grateful to **Dr. R. Sreemathy**, Principal, Pune Institute of Computer Technology,for her constant encouragement and for providing valuable resources.

I would like to thank all those, who have directly or indirectly helped me for the completion of the work during this project.

Pravin Bharat Jadhav

M.E(DS)

List of Figures

List of Tables

Abstract

Time series data refers to an ordered sequence or a set of data points that a variable takes at equal time intervals. The stock market is considered to be one of the most highly complex financial systems, consisting of various components or stocks, the price of which fluctuates greatly with respect to time. Stock market forecasting involves uncovering the market trends with respect to time. All stock market investors aim to maximise the returns on their investments and minimise the risks associated with them. With stock markets being highly sensitive and susceptible to quick changes, the main aim of stock-trend prediction is to develop new innovative approaches to foresee the stocks that result in high profits. Because stock markets are highly sensitive and prone to quick fluctuations, the core purpose of stock-trend prediction is to develop new, unique approaches to foresee the stocks that will result in large profits. The goal of this research is to look at the time series data of the Indian stock market and develop a statistical model that can accurately predict future stock values. As a result, we are using computational power in this study to forecast the future price of stock market data. We are using Nifty 50 data to make predictions, and we're using the Auto Regressive Integrated Moving Average (ARIMA) model because it's the best at forecasting time series data.

Keywords: Time series analysis,Auto Regressive Integrated Moving Average (ARIMA), Auto Correlation Function(ACF), Partial Auto Correlation Function(PACF), Simple Moving Average(SMA), Exponential Moving Average(EMA),Prediction,Machine Learning,Deep Learning .

Contents

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

A time series is a collection of data points that have been indexed (or listed or graphed) in chronological sequence. A time series is a collection of images taken at evenly spaced intervals over a period of time. As a result, it's a series of discrete-time data points. stock prices over a set period of time, hotel reservations, e-commerce sales, weather cycle reports, and so on. Time Series Analysis: Methods for studying time series data in order to extract useful statistics and other aspects of the data are referred to as time series analysis. The employment of a model to predict future values based on previously observed values is known as time series forecasting.Stock prices, sales demand, website traffic, daily temperatures, quarterly sales. These are examples of time series data.

Trends and seasonality are the main components of time series. A trend in data is a pattern that demonstrates the movement of a series to progressively higher or lower values over time.A trend usually lasts for a short period of time before dissipating; it does not recur. For example, some new kaggle kernels become popular for a short time before dissipating. There's a good probability it'll become popular once more.When Time Series Analysis reveals a general rising pattern, it is called an uptrend.When Time Series Analysis reveals a downward pattern, it is called a downtrend. A horizontal or stationary trend occurs when there is no discernible pattern.

Seasonality is the property of a time series in which the data undergoes predictable and recurring changes across the calendar year. Seasonal refers to any predictable variation or pattern that recurs or repeats over a one-year period.Seasonality refers to predictable variations in a business or economy over a one-year period that are dependent on the seasons, such as calendar or commercial seasons.Seasonality is a tool that can be used to examine stock prices and economic trends.Seasonality can be used by businesses to assist them in making decisions about inventory and staffing. Retail sales are an example of a seasonal statistic, with higher expenditure often occurring in the fourth quarter of the calendar year.

ARIMA stands for "Auto Regressive Integrated Moving Average," and it's a forecasting technique based on the premise that past values of a time series can be used alone to predict future values.

ARIMA is a univariate time series data forecasting approach. It supports both an autoregressive and a moving average element, as its name suggests. The method's integral element is differencing, which allows it to support time series data with a trend.ARIMA has the drawback of not supporting seasonal data. That is a time series with a cycle that repeats.ARIMA expects data that is either not seasonal or has had the seasonal component removed, such as data that has been seasonally adjusted using seasonal differencing methods.

SARIMA (Seasonal Autoregressive Integrated Moving Average) is an extension of ARIMA that handles univariate time series data with a seasonal component.

It adds three new hyperparameters for the seasonal component of the series: autoregression (AR), difference (I), and moving average (MA), as well as an additional parameter for the seasonality period. Additional seasonal parameters are added to the ARIMA model to create a seasonal ARIMA model.The seasonal component of the model consists of words that are quite similar to the non-seasonal components, but they involve seasonal backshifts.

In this study, we are using the Nifty Index Price for Prediction and we have downloaded it from Yahoo Finance.

1.2 MOTIVATION AND NEED

Nowadays, the stock market may be the secondary income source for some people. Investors and traders take their trading decisions on the basis of their analysis. Whether it is fundamental or technical analysis they are performing to predict the future price of the stock, However, failures in analysis can result in significant losses for investors and traders. it is necessary to predict the future price of the stock by using some of the computational power. We are using machine learning and deep learning techniques to perform the prediction task.

CHAPTER 2

LITERATURE SURVEY

This study proposes several machine learning algorithms for the four stock market categories on the Tehran stock exchange, including diversified financials, petroleum, non-metallic minerals, and basic metals. Nine machine learning models are compared in this study: decision tree, Random Forest, Adaptive Boosting, eXtreme Gradient Boosting, Support Vector Classifier, Nave Bayes, K nearest Neighbours, Logistic Regression, and Artificial Neural Network, as well as two deep learning algorithms: Recurrent Neural Network and Long Short Term Memory. In comparison to the other machine learning models, recurrent neural networks (RNN) and long short-term memory (LSTM) do the best [1].

Various feature selection algorithms were used in this work to pick features. The parameters of the machine learning-based stock trend prediction models are determined using temporal sliding window cross validation using data from the Chinese stock market for the past eight years. Both feature selection and stock trend prediction are done using the random forest model. This research is used to pick stocks in the Chinese stock market. They looked at the annual returns of Chinese corporations from 2011 to 2018 [2].

The author of this research proposes combining the complete ensemble empirical mode decomposition with adaptive noise and intrinsic sample entropy to improve the predictability of financial time series. The SP 500 Index stock was employed in this analysis, and the time period was from January 2018 to April 2020 for each closing price of stock. Using empirical model decomposition and sample entropy, they devised a strategy for reducing the complexity of financial time series. The suggested method is evaluated using an LSTM model that compares both series using conventional metrics such as MAPE, WAPE, direction correctness, TheilsU, and Arv, among others [3].

In this paper, they used an upgraded method of attitude analysis to study the predictability of stock market movement directions. They looked at the stocks of ten important corporations that were part of distinct NASDAQ stock domains. This methodology includes a comprehensive causality analysis, an algorithmic feature selection, and a number of machine learning approaches, including regularised model stacking, that round out this methodology. The NASDAQ-100 firms from 2008 to 2018 were used in this study, and the data was collected from Yahoo Finance. The training dataset spans 2008 to 2017, accounting for 80% of the total dataset, while the testing dataset spans 2016 to 2018. This accounts for 20% of the total dataset. For stock price prediction, machine learning techniques such as Nave Bayes, Logistic Regression, Support Vector Machine (SVM), Artificial Neural Netowrk (ANN), random forest, and XGBoost are utilised. We used the Random Forest, XGBoost, and Recursive Feature Selection techniques to choose features. We utilised Linear Principal Component Analysis (PCA) and Kernel PCA transformations to reduce dimensionality [4].

Because different factors have different effects on different stocks, it's important to find the right combination for each one. In this paper, the author proposes using a Genetic Algorithm for feature selection and developing an optimised Long-Short Term Memory neural network stock prediction model. They first used Ga to determine the relevance of a factor, then used a trial-and-error process to find the best combination of elements. This study relied on the CSI 300 stock dataset [5].

For the purposes of this article, Foxconn company data from the Taiwan Stock Exchange is used for testing. The author proposes that the average of the previous five days' stock market information be used as a new value, and that this value be used for prediction, with the forecast value being used as the average of the stock price information for the next five days. The author of this work has provided a quick overview of RNN and LSTM. Other methodologies such as Stochastic Oscillator, Moving Average Convergence/Divergence, Relative Strength Index, On Balance Volume, and others were used in this study. On foxocon stock, the accuracy of the ARIMA model MAPE and RMSE is 2.29 and 3.18, correspondingly [6].

The author proposes Linear Regression, Support Vector Machine, K-nearest Neighbor, Random Forest, Gradient Boosting Decision Tree, and Long Short Term Memory as ensemble machine learning algorithms in this study. The data for the Chinese stock market is selected for 18 years, from 2000 to 2017. In this study, Moving Average, Exponential Moving Average, Double Exponential Moving Average, Kaufman's Adaptive Moving Average, Simple Moving Average, Parabolic SAR, and other Overlap Indicators are utilised. As volume indicators [7], the Average Directional Movement Index, Price Oscillator Absolute, Balance of Power, Commodity Chanel Index, Moving Average Convergence, Money Flow Index, Momentum, and Relative Strength Index are used as volume indicators [7].

Four types of architecture are utilised in this article to anticipate the company's stock prices based on historical prices. Convolutional Neural Network, Multilayer Perceptron, Recurrent Neural Network, Long-Short Term Memory The National Stock Exchange of India and the New York Stock Exchange were employed in this investigation. The network was trained on a single NSE stock and then forecasted for five other National Stock Exchange and New York Stock Exchange businesses. The CNN outperforms other models, and the findings were compared to those obtained using the ARIMA model [8].

The author of this study examines the impact of macroeconomic variables on India's NSE and BSE. This study considers a total of forty-four macroeconomic indicators during a period of eight years, from 2011 to 2018, and it determines the correlation matrix of all analysed indicators. The Principal Component Analysis approach is used to reduce dimensionality to seven factors, after which the varimax rotation method is used to determine factors with the greatest variance. The analysis is based on the NSE Nifty and BSE Sensex indices. They have an accuracy of 92 percent and 87 percent, respectively, on the NSE Nifty and the BSE Sensex [9].

The author has depicted a neuro-fuzzy system made up of an adaptive neuro-fuzzy inference system controller that is used to regulate the stock market process model in this paper. The concept of an ANFIS controller was introduced by the author using neuro-fuzzy techniques. The information used was obtained from the New York Stock Exchange (NYSE) [10].

The author has given a system for predicting stock prices using the datasets KSE 100 Index, Lucky Cement Stock, and Engro Fertilizer Limited in this research. They must use the R programme to conduct multiple regression on these datasets. The value of R-Squared in the "KSE 100 Index" dataset is 0.95, indicating that the predictor can measure the variation in stock return up to 95 percent. The value of multiple R-Squared for our model on the "Lucky Cement" dataset is 0.89, indicating that the predictor can quantify the variation in stock return up to 89 percent. Finally, the "Engro Fertilizer Limited" dataset produces a multiple

R-squared value of 0.97, indicating that our model is 97 percent accurate [11].

An ARIMA model is utilised in this research to forecast stock market movement. An ARIMA model is a dynamic uni-variate forecasting method for projecting the future values of a time series. Identification of the problem, data collection, preliminary investigation, model selection, and evaluation This is what predicting entails. Auto-regressive Integrated Moving Average (ARIMA) is an abbreviation for AR+I+MA (Auto-regressive Integrated Moving Average). The Box-Jenkins model is another name for this concept. This model is also known as the (p, d, q) model, where "p" stands for the "order of auto-regressive portion," "d" stands for the "degree of difference," and "q" stands for the "order of moving average part". In this technique, the first step is to analyse the time series, then identify stationary in the data, estimate the ACF and PACF parameters, and finally run the ARIMA model to forecast future stock market prices. In this study, they are forecasting the Nifty and Sensex (2010-2016). The anticipated time series varies by about 5% mean percentage error [12].

The Bombay Stock Exchange and the National Stock Exchange are the two primary stock exchanges in India now. In this paper, the closing price of the stocks of Tata Consultancy Services, Wipro, Dr. Reddy's Laboratories, and Sun Pharmaceutical, which are all listed on the BSE, is predicted. 70 percent of the data is used for training, while the remaining 30 percent is used for testing. The neural network utilised in this study forecasts stock closing prices for the next five days. The Artificial Neural Network concept is based on the functions of the human brain.A transfer function is an activation function that each neuron uses to process information. The hard limit, simply linear, sigmoid, and tan sigmoid functions are the most commonly used transfer functions. The neural network bias is represented as an input value of -1, with its corresponding weight equal to the total of the other input weights.The input layer, the hidden layer, and the output layer are the three layers that make up an ANN. There is only one input layer and one output layer, but the number of hidden layers can range from 0 to any number. After specifying the architecture and transfer function for each neuron, the weights between the neurons are changed, which is referred to as neural network training.Weights for the networks are first assigned from the range [-1, 1], after which the first sample's input is presented to the network, which computes an output. The calculated output is compared to the sample's target value, and the weights of the network are adjusted to minimise the error between the calculated output and the target. Mean Squared Error and Mean Absolute Error are two types of errors. The most frequent error routines are listed below. In incremental training, there are two types of training: batch training and incremental training. Weights are updated each time each of the input samples is presented to the network, which is known as incremental training. The weights are modified in batch training only after all of the training samples have been delivered to the network. The training algorithm is the technique for updating weights. Back-propagation is the most widely used algorithm, which propagates errors back during training and adjusts weights based on these errors. The activation function of this network is a logistical sigmoid function, which is implemented using the dynamic back propagation paradigm.The dataset was analysed using Neuroph and Matlab R2010, using the following inputs: opening price, closing price, high, low, and volume of the stocks. Five input neurons, two hidden neurons, and one output neuron make up the network [13].

CHAPTER 3

PROBLEM DEFINITION AND SCOPE

3.1 PROBLEM STATEMENT

Stock market forecasting is a process that involves technical analysis as well as fundamental analysis.Fundamental analysis is used to determine the inherent worth of a particular stock, whereas technical analysis is used to forecast future price patterns.The investment function is handled by fundamental analysis, while the trading function is handled by technical analysis. Technical analysis is undertaken by examining price movements and patterns displayed on charts, whereas fundamental analysis is conducted by analysing many economic elements.Fundamental analysis is done by long-term traders, whereas technical analysis is done by swing and short-term traders.External news has little bearing on fundamental analysis, but external news has an impact on technical analysis.Economic reports, industry statistics, brokerage analysis, financial statements, management processes, news events, and other sources of data are used for fundamental analysis, whereas chart analysis is used for technical analysis.Fundamental analysis employs the ideas of return on assets and returns on equity, whereas technical analysis uses price data and Dow theory.Technical analysis focuses on price and volume, whereas fundamental analysis considers both qualitative and quantitative elements.Long-term investments require fundamental analysis, and short-term investments require technical analysis.Technical analysis just looks at prior data, but fundamental analysis looks at both previous and current data.Fundamental analysis is used to determine a stock's worth based on economic variables, whereas technical analysis is used to determine a stock's price movement in order to forecast future price changes.

To overcome some shortcomings in technical as well as fundamental analysis, we are using computational power to perform the prediction. To predict the Nifty Index, we are implementing machine learning and deep learning algorithms, which may give better performance than the other prediction tools or methodologies. So, we are employing a model for the prediction of the Nifty Index using technical indicators.

3.1.1 GOALS AND OBJECTIVES

3.1.1.1 Goals

- Track the performance of the ARIMA model in forecasting the Nifty Index on the basis of closing price, six months of data, one year of data, Simple Moving Average Indicator, and Exponential Moving Average Indicator.

3.1.1.2 Objectives

The following are the dissertation work's objectives:

- To predict the trend of the Nifty Index.
- To achieve the best fit ARIMA model.
- To minimize the Root Mean Squared Error or gain in the accuracy of ARIMA model.
- .To understand the ARIMA model performance.
- To take the financial decision .

3.2 SCOPE

- Implementation of ARIMA model on six month of data.
- Implementation of ARIMA model on one year of data.
- ARIMA model on Simple Moving Average, Exponential Moving Average.

CHAPTER 4

DISSERTATION PLAN

4.1 RECONCILED ESTIMATES

To have a successful project development, we first need to estimate the resources that are required to complete the project development process. Estimation is the process of analysing or approximating the time and cost required to complete a project deliverable. Cost and time estimation is an important part of project development. This consists of two key features of a project, which are on-time and on-budget delivery for the project. Methods for estimating project development times and costs for projects mainly focus on making a simple and easy process of development and breaking it into little phases. Cost and time estimation help to manage all work flows of project development. Estimation assists in breaking down a project into small tasks that can be completed in a specific time frame. Dividing the project into smaller tasks lets a project team get an overview of the duration. Estimation gives information about the overall project system, i.e., the development flow of the system. So, we can develop projects on time. The project is related to software, so the cost requirement to complete this is negligible. The cost requirement for downloading the data set is zero because the Yahoo Finance website provides a free data set for its users to perform analysis.

Fig. 4.1 shows Dissertation work plan for this Dissertation. Dissertation work is divided into 5 phases.

1. Phase 1: Topic selection was done. Detail study of that topic was done based on previous work.

2. Phase 2: Collect the required data set from yahoo finance website. Import the required libraries to perform task.

3. Phase 3: Convert the data set in the required format and perform the Exploratory Data Analysis on the downloaded data set .

4. Phase 4: Forecast the Nifty index using Auto Regressive Integrated Moving Average.

5. Phase 5: To measure the performance of the ARIMA model calculate the Root Mean Squared Error value.

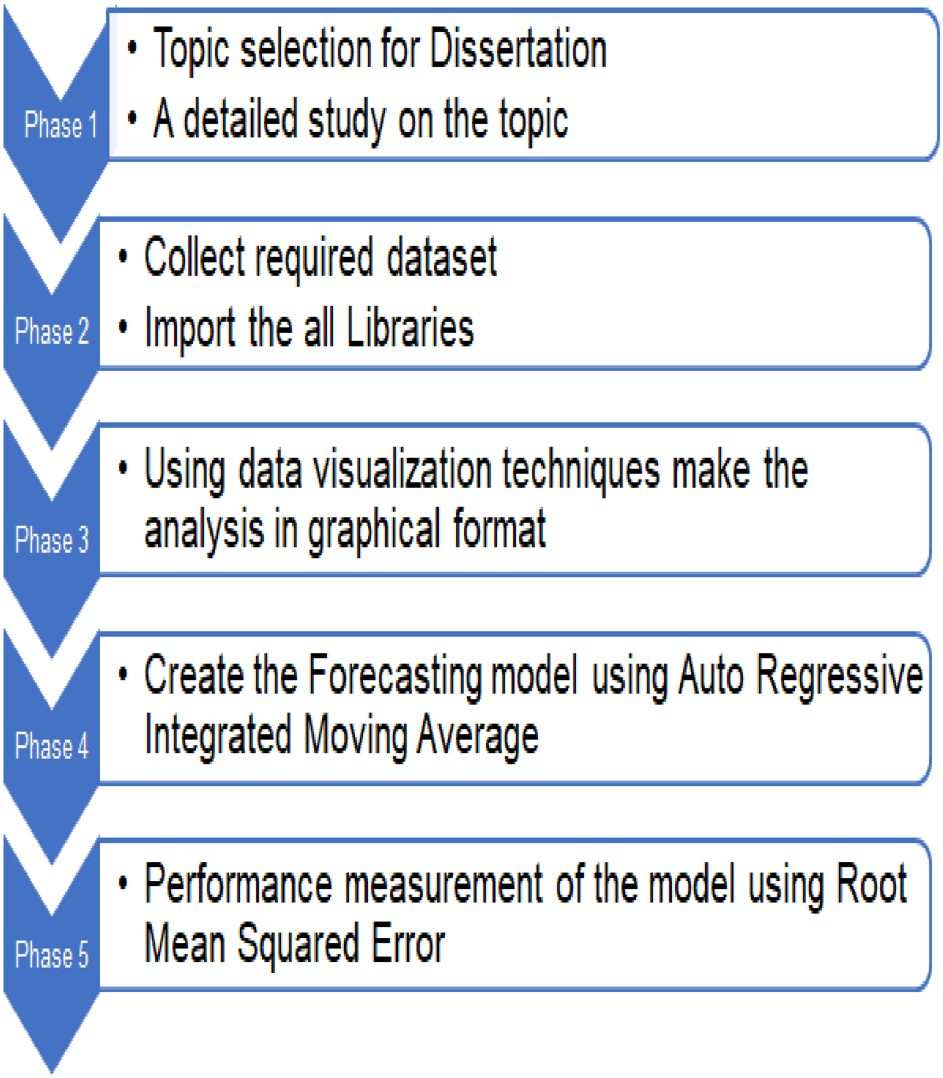

Figure 4.1: Dissertation work plan

4.1.1 TIME ESTIMATES:

Time Estimation is an important part of Project work. Time estimation provides information about time required for each module of system.

The number of lines necessary for each module's implementation can be approximated as follows:

Total Time Estimation for this project:

Total Time estimation = Addition of time required for each module of system

Total Time estimation = 9 Months.

Function	Time estimation (In Months)
Requirements analysis	2.0
Requirements gathering	1.0
ARIMA model Development	1.5
Model Implementation	2.5
Performance parameter	1.0
Testing	1.0

Table 4.1: Time estimation of project

4.1.2 COST ESTIMATE

Cost estimation is estimation of cost required for development of system. Cost estimation is an important factor in development. To management of money and make system in budget is main aim of cost estimation.

4.2 ANALYSIS MODELS: SDLC MODEL TO BE APPLIED

SDLC Model- The Agile Model, or Rapid Application Development Model, is a software development process model which is based on prototyping without any specific planning. It targets developing software in less time. In an Agile model, the systemis divided into small independent modules which can develop concurrently. By using the Agile model, we can develop systems rapidly.

Phases:

1. Analysis of Requirements

   It is the initial stage of software development in which the requirements for the software product to be made are collected. The types of requirements include user requirements, functional requirements, and non-functional requirements. After the requirements are collected, they are examined and analysed for validation, i.e., whether these requirements can be incorporated into the system or not. in a data processing technique that involves transforming raw data into an understandable format. In our case, we are converting the non-stationary data into a stationary format. Real-world

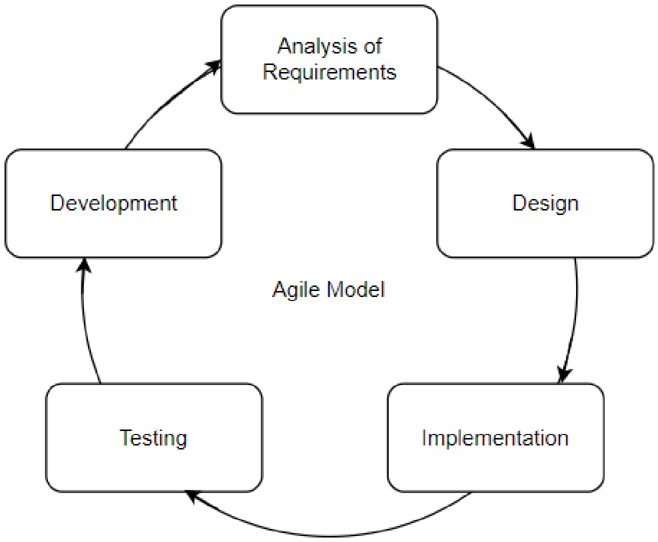

Figure 4.2: Agile SDLC Diagram

data is often incomplete, inconsistent, and/or lacking in certain behaviours or trends, and is likely to contain many errors. Data preprocessing is a proven method of resolving such issues. Data preprocessing prepares raw data for further processing. Data visualisation in Python is perhaps one of the most utilised features for data science in Python in today's day and age. The libraries in Python come with lots of different features that enable users to make highly customized, elegant, and interactive plots. In this project, the required analysis is to make sure that the data is downloaded from the correct or genuine source and also to make sure important attributes are present in the dataset.

2. Design

After confirming the requirements gathering of the software system, the designing of the software product is done. The designing of the software is done based on the requirements collected in the initial stage. An outline of the whole process is created in this phase, which will define the overall system architecture. In this study, the making of Date as an Index is to be done and also the calculation of all the required test cases is done before applying the model.

3. Implementation

   In this phase, the actual implementation of the software system takes place. An executable programming code is written in the Python Programming Language for implementation. The work is divided into different modules, and coding is done in each of these modules. In this project, the implementation of the ARIMA model is done.

4. Testing

   The testing phases follow the coding phases in which testing of the code is done to check whether the systems meet user requirements or not. Testing is required to find out any underlying errors and bugs in the product.

5. Development and Maintenance

   The maintenance code is updated in accordance with changes taking place in real time. maintenance to maintain the software product. Deployment This is the last phase where the product is actually delivered and installed at the customer's end, and, according to need, support is given for that system. After delivering the system, feedback is taken from the customer to ensure the quality of the product.

CHAPTER 5

SOFTWARE REQUIREMENTS AND SPECIFICATION

This Software Requirements Specification details all of the functions and constraints of the **"A Model for Prediction of Nifty Index using Technical Indicators".** The SRS explains the system's issues as well as the steps the development team must take to find a better solution.

5.1 ASSUMPTION AND DEPENDENCIES

Following are the assumptions:

- User able to forecast the Nifty index.
- Open , High, Low, Close, Adj Close, Volume the values are given in the data set.

Dependencies can be:

- System Speed.
- IDE Capacity.

5.2 FUNCTIONAL REQUIREMENTS

5.2.1 ARCHITECTURE MODULE

- The system will be dividing into six modules..
- Data Importing.
- Data Pre processing.
- Data Visualization.
- Data Analysis and test case calculation.
- Model Implementation.
- Results of the project.

5.2.2 USER ENVIRONMENT

- The system will be used on computer based windows operating system.
- The IDE used in this system is Anaconda Environment and the platform will be used for this is Jupyter Notebook.

5.2.3 OPERATING ENVIRONMENT

- The Proposed System Supports.
- OS: Windows 8, Windows

5.3 NONFUNCTIONAL REQUIREMENTS

5.3.1 PERFORMANCE REQUIREMENTS

- **High Speed:** The model should perform forecasting in minimum amount of time.
- **Accuracy:** The performance of the model should be as better as possible so the Root mean square error value as much as low.

5.3.2 RELIABILITY REQUIREMENTS

- The model should be predict the future price of the Data.
- The prediction should be good for Time series data set.
- The model should avoid or remove null values.

5.3.3 USABILITY REQUIREMENTS

- The model should be usable in different time series data set. Any user should be able to use the model.

5.3.4 MODIFIABILITY REQUIREMENTS

- The model should be easily modifiable as per user requirements.

5.3.5 REUSABILITY REQUIREMENTS

- The model should be reusable on different platform.
- The model should be implemented many times for same task.

5.4 SYSTEM REQUIREMENTS

5.4.1 SOFTWARE REQUIREMENTS(PLATFORM CHOICE)

1. Platform: Jupyter Notebook.
2. Technology : Machine learning, Deep Learning .
3. IDE: Anaconda Environment.

5.4.2 HARDWARE REQUIREMENTS

Sr. No.	Parameter	Minimum Requirement	Justification
1	Processor	2.2 GHz	For Fast Processing
2	SSD	256 GB	For Fast Processing
3	RAM	12 GB	For Fast Processing
4	Monitor, Keyboard and Mouse	1 Quantity	None

Table 5.1: Hardware Requirements

CHAPTER 6

DETAIL DESIGN DOCUMENT

6.1 SYSTEM ARCHITECTURE

In the form of several layers, the system architecture offers the architecture for the proposed system.

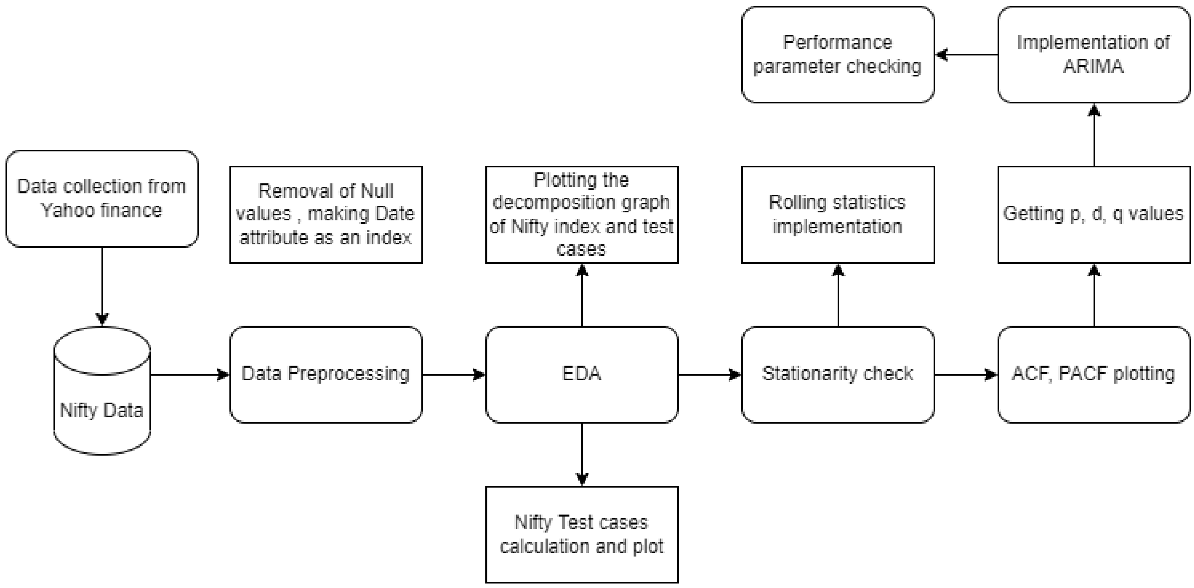

Figure 6.1: System Architecture

6.2 DATA FLOW DIAGRAMS

6.2.1 DFD LEVEL 0

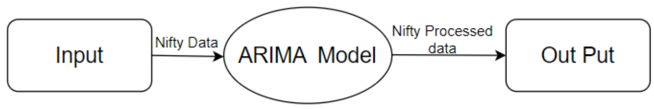

Figure 6.2: DFD Level 0 Diagram

6.2.2 DFD LEVEL 1

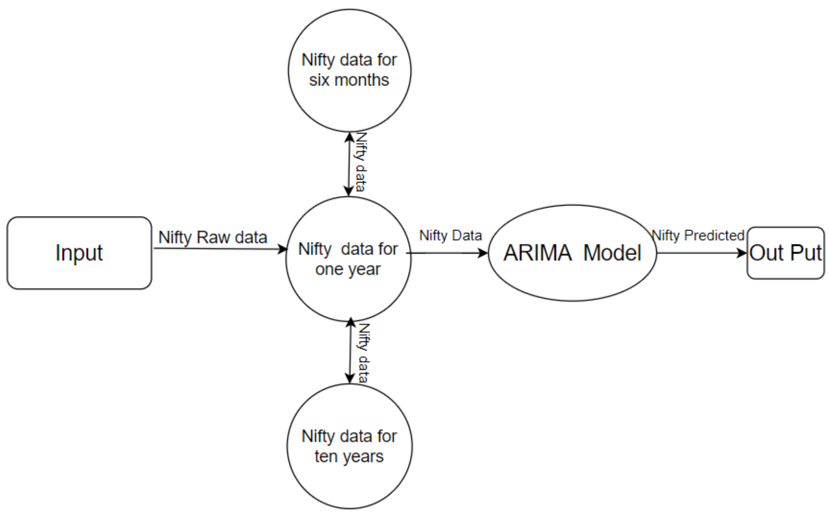

Figure 6.3: DFD Level 1 Diagram

6.2.3 DFD LEVEL 2

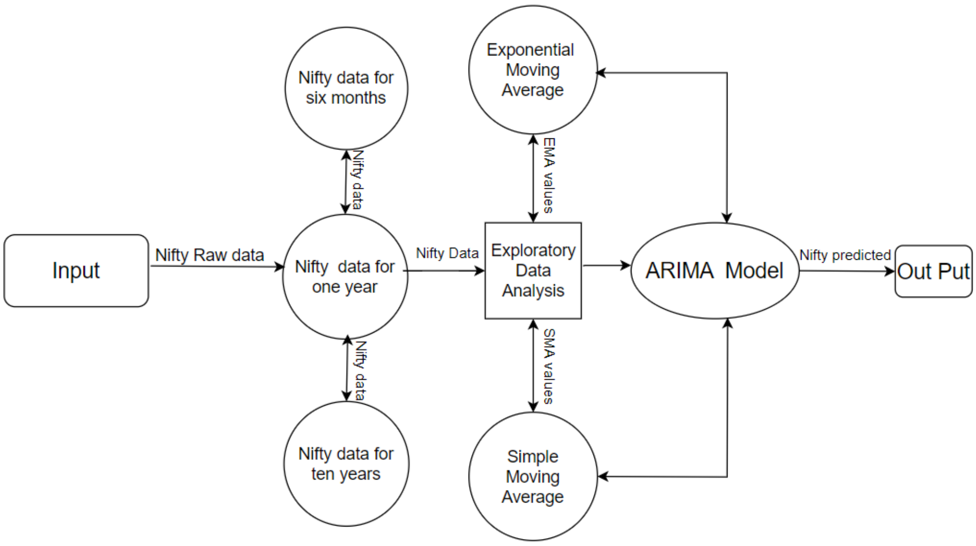

Figure 6.4: DFD Level 2 Diagram

6.3 UML DIAGRAMS

6.3.1 USE CASE DIAGRAM

Figure 6.5: Use Case Diagram

6.3.2 SEQUENCE DIAGRAM

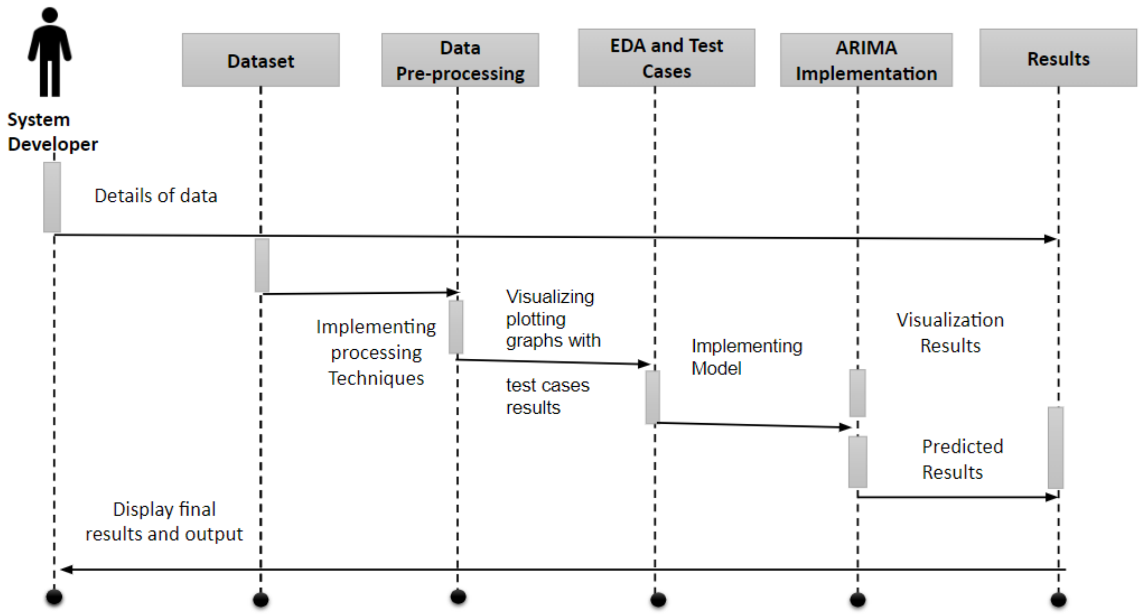

Figure 6.6: Sequence Diagram for developer

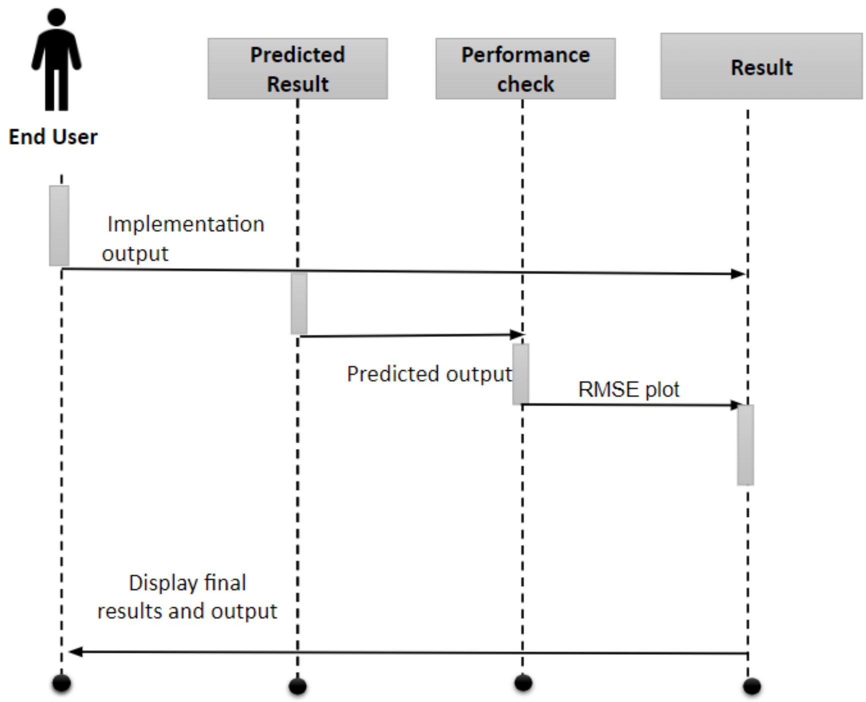

Figure 6.7: Sequence Diagram for end user

CHAPTER 7

IMPLEMENTATION

7.1 IMPLEMENTATION DETAIL

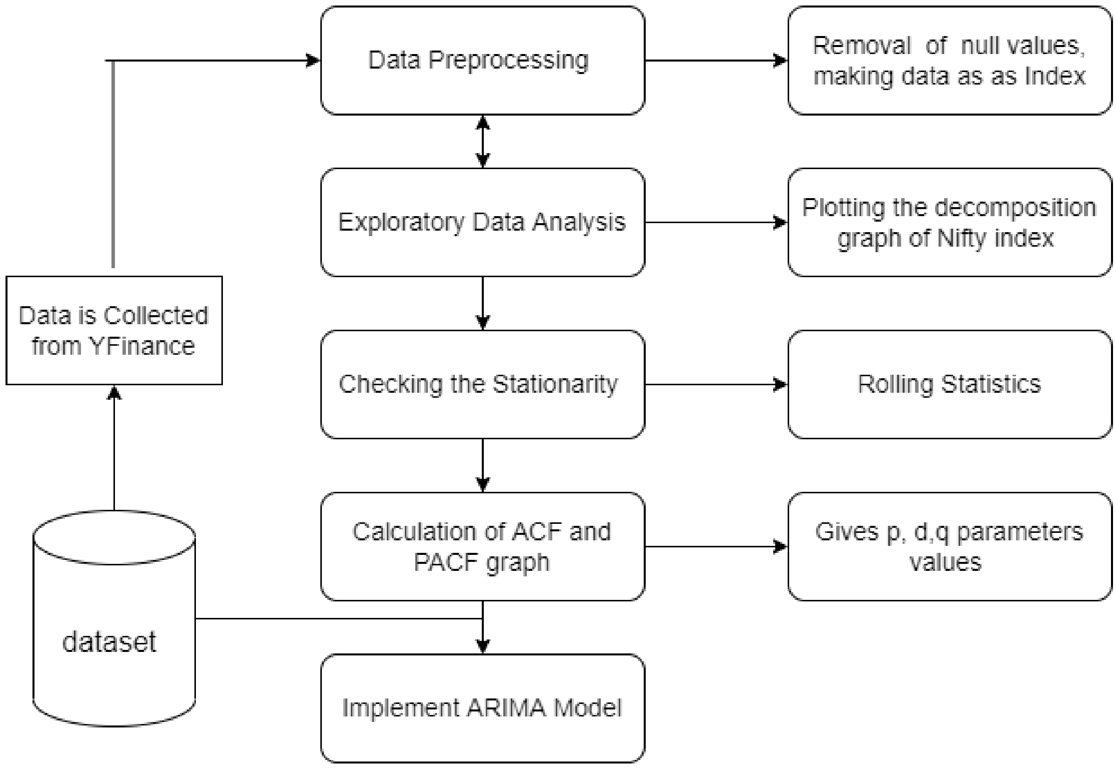

Figure 7.1: System Overview Diagram

Figure 7.1 shows the system overview diagram. In this study, the Auto Regressive Integrated Moving Average model is implemented for the forecasting of the Nifty index price based on its closing price. The Autoregressive Integrated Moving Average (ARIMA) Model converts non-stationary data to stationary data before working with it. This is one of the most commonly used models for predicting linear time series data.Since it was discovered to be dependable, efficient, and capable of predicting short-term share market changes, the ARIMA model has been widely used in banking and economics.

7.1.1 TIME SERIES DATA

A time series is a collection of measurements taken at regular intervals.Depending on the frequency, the time series can be yearly, monthly, weekly, daily, hourly, etc.

Forecasting is the next step in the process, and it involves predicting the future values.Because anticipating a time series (such as demand and sales) can be extremely profitable.It is the major factor driving the essential business planning, procurement, and production processes in most manufacturing businesses. Any forecasting errors will affect the entire supply chain, or any business scenario for that matter. As a result, it's crucial to get the forecasts right in order to save money and succeed.The tools and concepts behind time series forecasting are relevant in any business, not only manufacturing. Time series forecasting in a time series is divided into two types: uni-variate time series and multi-variate time series. ARIMA (Auto Regressive Integrated Moving Average) is a forecasting technique based on the premise that past values of a time series can be used to predict future values by themselves.

7.1.2 INTRODUCTION OF ARIMA MODEL

ARIMA (Auto Regressive Integrated Moving Average) is a class of models that "explains" a time series based on its own previous values, that is, its own lags and lagged prediction errors, so that the equation can be used to anticipate future values. An ARIMA model can be categorised by 3 terms: p, d, and q.

P is the order of the auto-regressive term in the model, q is the order of the moving average term in the model, and d is the number of differences required to make the series stationary. If a time series has seasonal trends, seasonal terms must be added, and the time series becomes SARIMA, short for 'Seasonal ARIMA.'

The first step of the ARIMA model is to make time series stationary, so how to make time series stationary?

The most popular method is to differentiate it. That is, take the old value and subtract it from the present value. Depending on the complexity of the series, multiple differencing may be required at times.As a result, the value of d is the smallest number of differences required to make the series stationary. And $d = 0$ if the time series is already stationary.

The 'Auto Regressive' (AR) term's order is 'p'. It's the number of Y delays that will be utilised as predictors. The order of the 'Moving Average' (MA) word is 'q'. It relates to how many lag forecast mistakes should be included in the ARIMA Model.

A pure Auto Regressive (AR alone) model is one in which Yt is only determined by its own lags. To put it another way, Yt is a function of the 'lags of Yt.'

$\mathrm{Y}_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Yt - 2 + .. + \beta_p Y_{t-p} + \epsilon_1$

A pure Moving Average (MA alone) model, on the other hand, is one in which Yt is only determined by the lagged forecast errors.

$\mathrm{Y}_t = \alpha + \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + ... + \phi_q \epsilon_{t-q}$

The purpose of varying is to make the time series stationary. However, you must be cautious not to over-dislike the series. Because an over-differenced series may still be stationary, the model parameters will be affected.

The least differencing required to obtain a near-stationary series that roams around a defined mean and the ACF plot quickly hits zero is the proper sequence of differencing.If the autocorrelations are positive for a large number of lags (10 or more), the series should be differentiated further. If the lag 1 auto correlation is too negative, on the other hand, the series is most likely over-differenced.

Rolling statistics and the Dickey Fuller test are used to find whether the data set is stationary or not. In rolling statistics, if the standard mean and the standard deviation are constant over the period of time, then the time series data is stationary, and if it is not, then it is non-stationary.

In terms of the A Dickey-Fuller test, if it gives a p value less than 0.05, then it is stationary data, and if it gives a p value greater than 0.05, then the series is non-stationary.

series, after making stationary plotting of the Auto Correlation Function and Partial Auto Correlation Function graphs, and then implementation of the ARIMA model with the p, d, and q parameters given by the ACF and PACF graph.

7.2 ALGORITHM

Algorithm of ARIMA model shown below.

Algorithm 1 Nifty prediction Model

```
// Input: Data set is downloaded from yahoo finance
// Output: Predicted price of Nifty Index
// function: ARIMA Model
Start
Download the data set
Data pre processing and making Date as Index
Test case calculation and Visualization
Stationarity check
if Non − stationary : then
    Apply Rolling statistics
Making mean and standard deviation constant
Making data stationary
Plot ACF , PACF parameter
Implement ARIMA model with p, d, q values
Performance parameter calculation
Stop
```

7.3 EXPERIMENTAL SETUP

The proposed system has implemented the ARIMA model for prediction of the Nifty Index price using python programming on a Jupyter notebook. In this system, primarily employment of the ARIMA model on the Nifty closing price for six months, further implementation of the ARIMA model on the Nifty Index closing price for one year of data, and then similarly for ten years of data. The Simple Moving Average and Exponential Moving Average calculations are done and plotted the visualizations. Finally, I used the ARIMA on the SMA and EMA. For checking the performance of the model, Mean Squared Error, Mean Absolute Error, and Root Mean Squared Error are calculated for every test case.

CHAPTER 8

TEST SPECIFICATION

8.1 TESTING PRINCIPLES

1. All tests should be traceable to end user requirements
2. Tests should be planned long before testing begins
3. Testing should begin on a small scale and progress towards testing in large
4. To be most effective testing should be conducted by a independent third party

8.2 TYPE OF TESTING USED

8.2.1 PERFORMANCE TESTING

The performance of the system is measured by the Mean Absolute Error, Mean Squared Error and Root Mean Squared Error

8.3 TEST CASES AND TEST RESULTS

Below table shows Test Cases and Result of each Test Cases.

Test no	Input	p, d, q Values	Mean Squared Error	Mean Absolute Error	Root Mean Squared Error
1	Nifty Closing price for six months of data.	(3,1,3)	12228892.879	920.247	1105.844
2	Nifty Closing price for one year of data.	(1,1,1)	817498.307	753.216	904.156
3	Nifty Closing price for ten years of data.	(5,1,5)	969014.858	909.440	984.385
4	Simple Moving Average for Closing price of one year.	(1,1,1)	764.984	23.52	27.65
5	Exponential Moving Average for Closing price of one year.	(1,1,1)	907.827	25.48	30.13

Table 8.1: Test Cases

CHAPTER 9

RESULTS AND DISCUSSIONS

9.1 OUTCOMES

In this work, Nifty data is used for prediction. The Auto Regressive Integrated Moving Average model is used for forecasting the Nifty price. Because this is time series data, the date attribute should be used primarily as an index.selection. and plotting the 6 months of data for implementation of the first test case. Then Plotting the graph of one year of data for Nifty and selecting and plotting the graph of 10 years of data. Shown below,

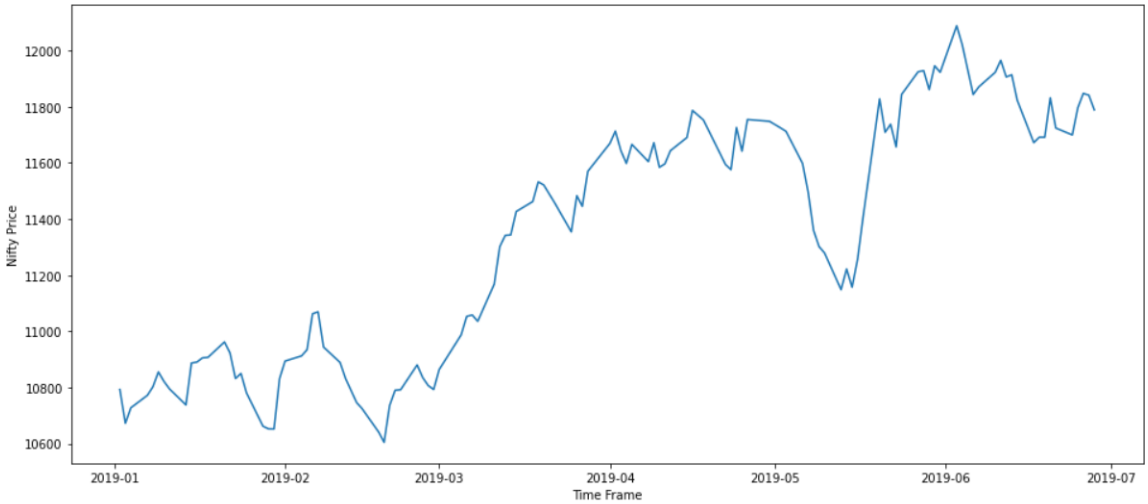

Figure 9.1: Nifty Index six months data

Figure 9.1 shows the line chart for the Nifty Index based on its closing price. The time period taken for this plot is six months.

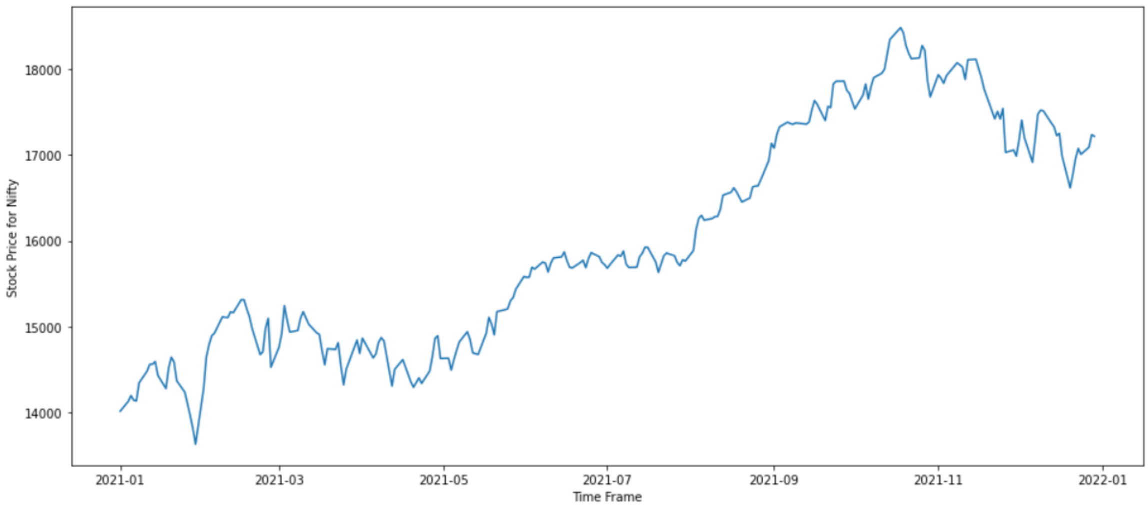

Figure 9.2: Nifty Index one year data

Figure 9.2 shows the line chart for the Nifty Index based on its closing price. The time period taken for this plot is one year.

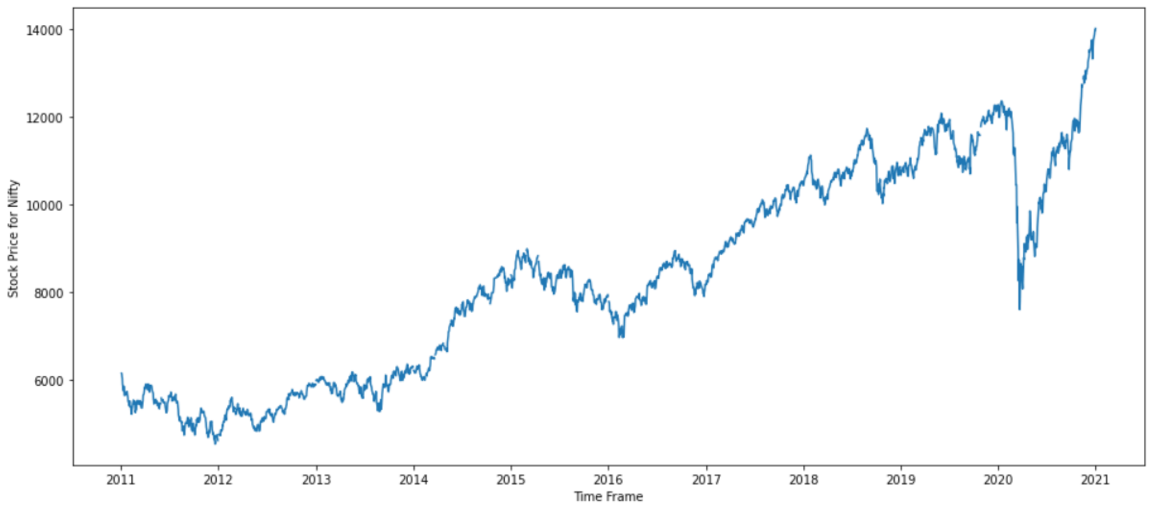

Figure 9.3: Nifty Index ten year data

Figure 9.3 shows the line chart for the Nifty Index based on its closing price. The time period taken for this plot is ten years.

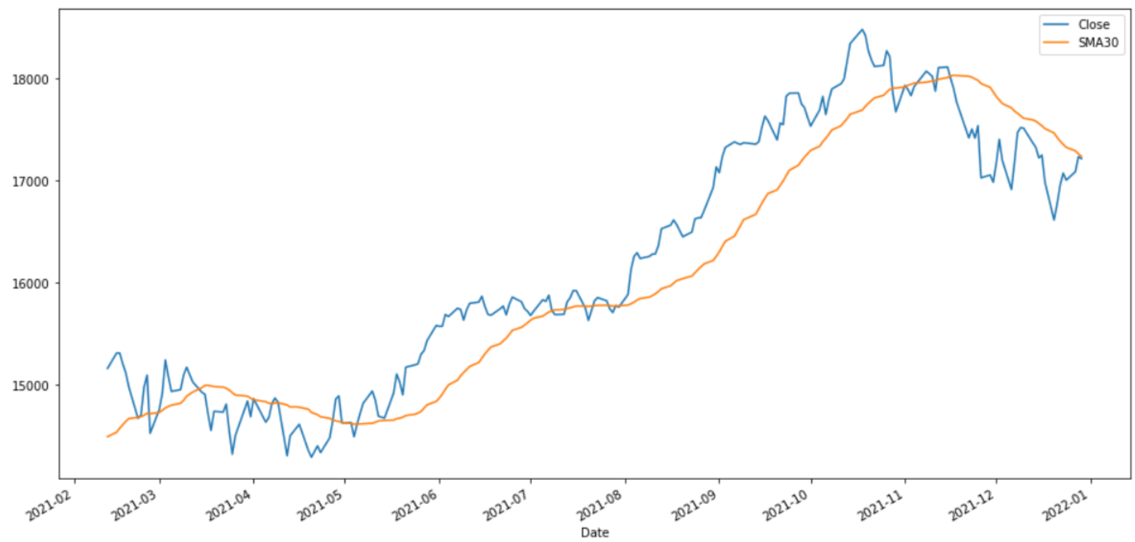

Figure 9.4: SMA of Nifty Index for one year data

Figure 9.4 shows the line chart for the SMA of Nifty Index based on its closing price. The time period taken for this plot is one year.

$$\begin{aligned}\overline{p}_{\text{SM}} &= \frac{p_M + p_{M-1} + \cdots + p_{M-(n-1)}}{n} \\ &= \frac{1}{n}\sum_{i=0}^{n-1} p_{M-i}\end{aligned}$$

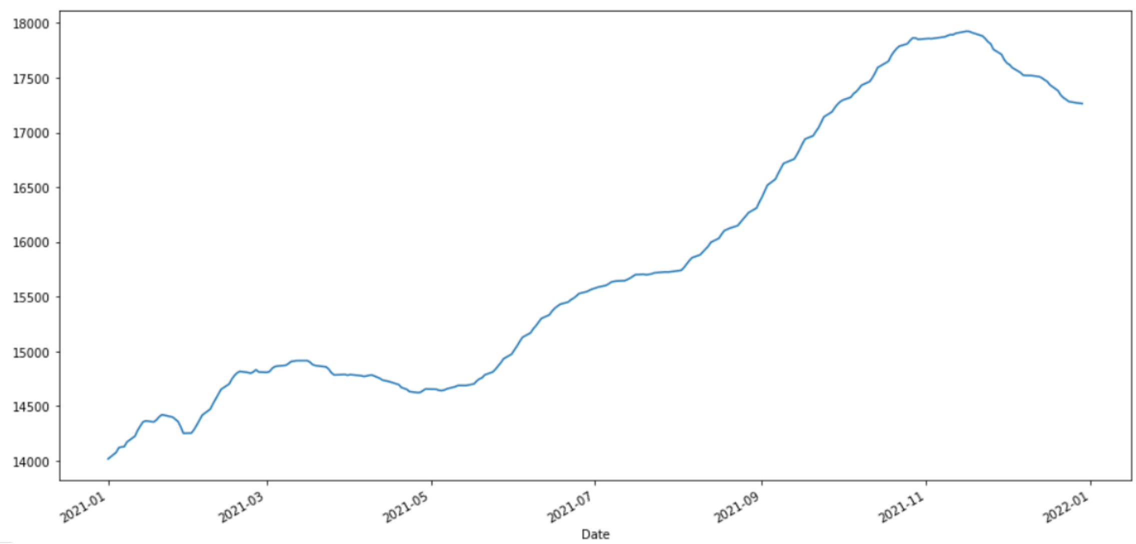

Figure 9.5: EMA of Nifty Index for one year data

Figure 9.5 shows the line chart for the EMA of Nifty Index based on its closing price. The time period taken for this plot is one year.

EMA=(2/n + 1)*(Close-PreviousEMA) + $Previous$EMA

9.2 STATIONARITY CHECK

Before applying the ARIMA model, it is necessary to check whether the data is stationary or not. So, that is this project. A Dickey Fuller test and the rolling statistics are calculated for the same. If the rolling mean and the standard deviation are constant over the period of time, and the ADF test P value is greater than 0.5, then it is shown that the time series data is not stationary. And it is necessary to make it stationary for further implementation of the ARIMA model.

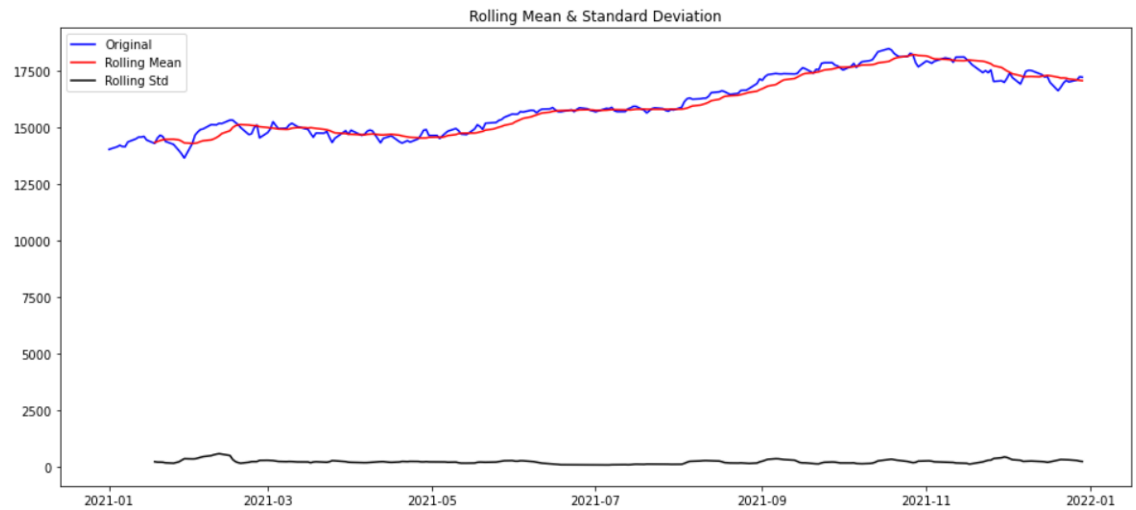

Figure 9.6: stationarity check for Nifty

In the above figure, 9.6 on the X-axis is a time period and the Y-axis is the value of the Nifty closing price. The rolling mean and the rolling standard deviation are not constant over the period of time, and the ADF test gives the p value as 0.65, so it is shown that the data set is not stationary.

To make data stationary, first take it in log form and then differencing it; this process will make data stationary.

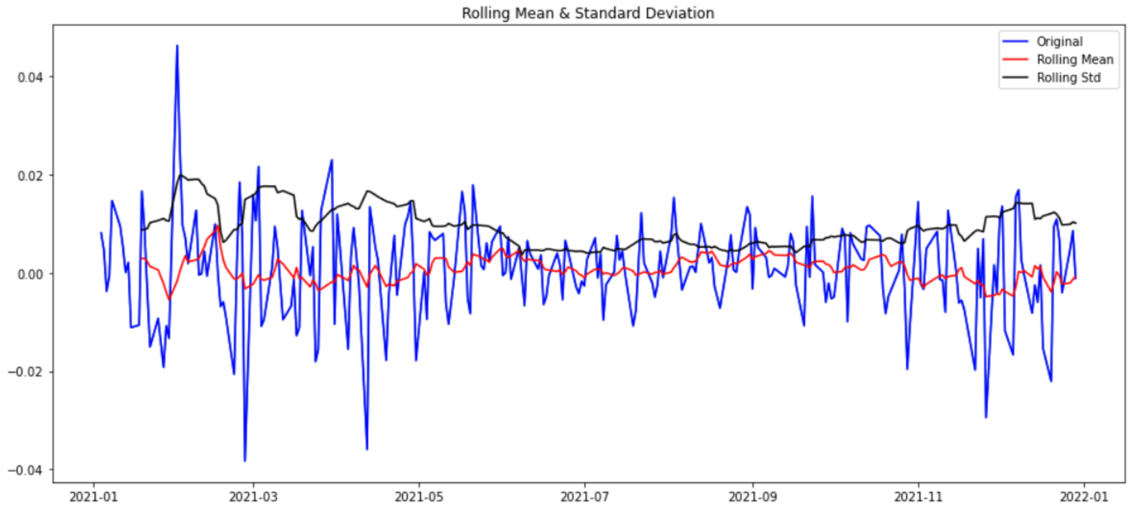

Figure 9.7: Nifty stationary data

The figure 9.7 on the X-axis represents a time period and the Y-axis is the value of the Nifty Closing price. The mean and the standard deviation are constant over the period, so that the data set is stationary.

9.3 ACF , PACF PARAMETER PLOT

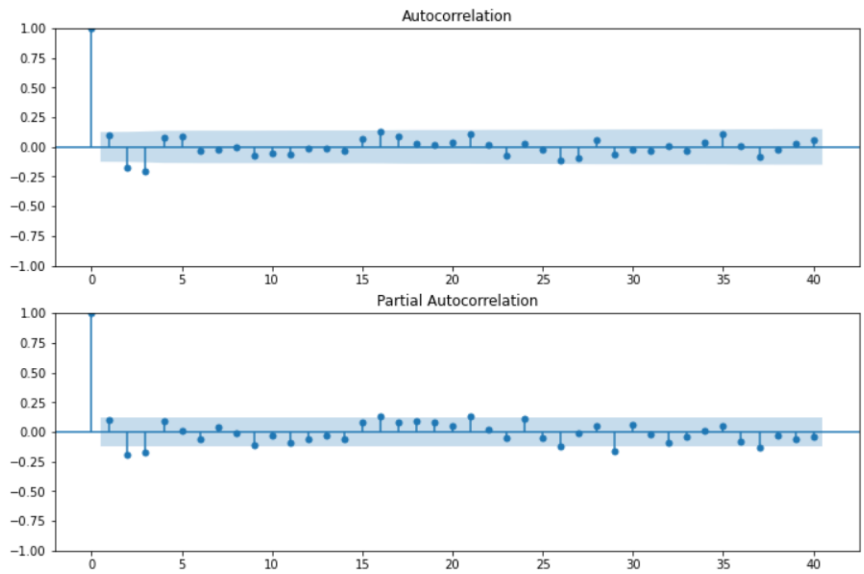

Figure 9.8: ACF, PACF graph for one year of Nifty Price

Figure 9.8 depicts the auto correlation function and partial auto correlation function graphs of the Nifty for a year, yielding (p, d, q) values of (1,1,1), respectively.

9.4 ARIMA PLOT

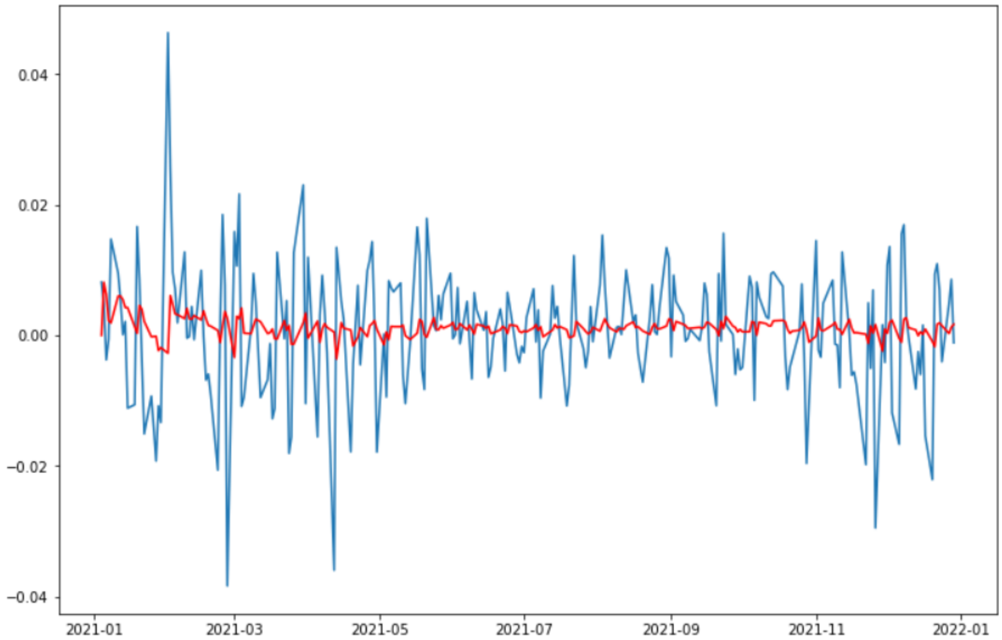

Figure 9.9: ARIMA result for Nifty Closing price for one year

The above figure 9.9 shows the predicted and original values of the Nifty Closing price for one year. The blue line is the original data and the red line is the predicted values of Nifty.

9.5 PERFORMANCE PARAMETER

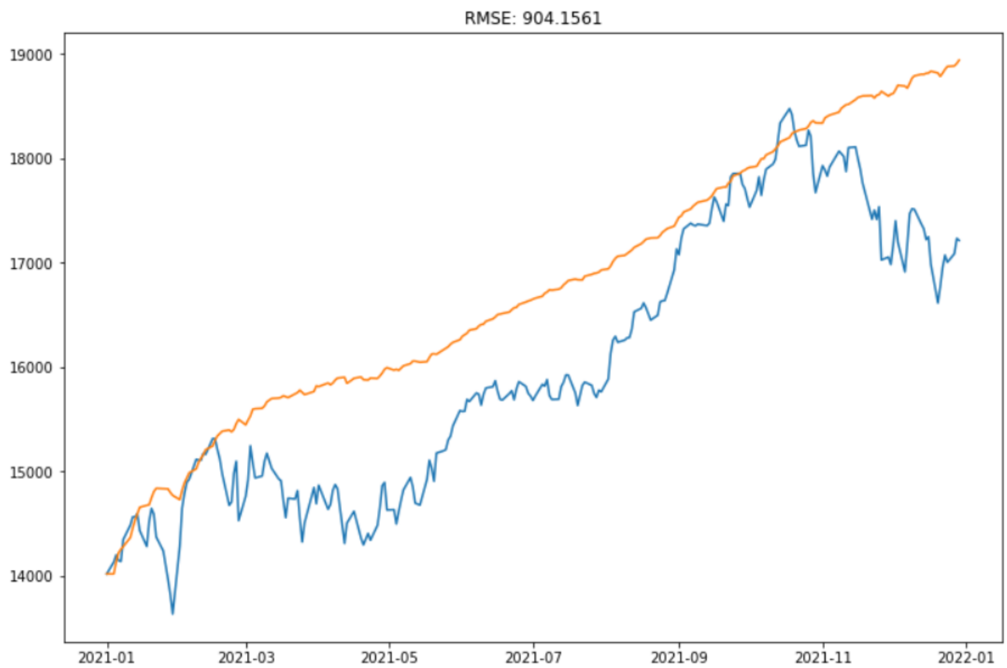

Figure 9.10: RMSE result for Nifty Closing price for one year

The above figure 9.10 shows the Root Mean Squared Error for the Nifty Closing Price for one year, which is 904.1561. The RMSE value for six months of Nifty data is 1105.844%.For ten years of Nifty data, the RMSE value is 984.385%.The RMSE value for the Simple Moving Average of Nifty Data for one year is 27.65%.The RMSE value for the Exponential Moving Average of Nifty Data for one year is 30.13%.

Root Mean Square Error : $\sqrt(\sum_{i=1}^{D}(x_i - y_i)^2)$

$MeanAbsoluteError : \sum_{i=1}^{D} |x_i - y_i|$

$MeanSquareError : \sum_{i=1}^{D}(x_i - y_i)^2$

CHAPTER 10

CONCLUSIONS

10.1 CONCLUSION

In this system implementation of the ARIMA model on Nifty time series data on closing prices for one year, six months, and ten years and also on Simple Moving Average and Exponential Moving Average, it is observed that the Root Mean Square Error value is getting very low, i.e. 27.65%, which means the ARIMA model works well with the Simple Moving Average data of the Nifty Index for one year as compared to the other test cases.

CHAPTER 11

FUTURE WORK

11.1 FUTURE WORK

the future, this model will be implemented on other stock exchanges or in other stocks also.This model can be implemented in the live market.

• REFERENCES

[1] A. Carrio, J. Tordesillas, S. Vemprala, S. Saripalli, P. Campoy and J. P. How, "Onboard Detection and Localization of Drones Using Depth Maps," IEEE Access, vol. 8, pp. 30480- 30490, 2020, doi: 10.1109/ACCESS.2020.2971938.

[2] Petrevska, B. (2017). Predicting tourism demand by ARIMA models. Economic research-Ekonomska istraživanja, 30(1), 939-950.

[3] Saha, S., Gao, J., Gerlach, R. (2021). Stock ranking prediction using list-wise approach and node embedding technique. IEEE Access, 9, 88981-88996.

[4] Reddy, V. K. S. (2018). Stock market prediction using machine learning. International Research Journal of Engineering and Technology (IRJET), 5(10), 1033-1035.

[5] Pahwa, N., Khalfay, N., Soni, V., Vora, D. (2017). Stock prediction using machine learning a review paper. International Journal of Computer Applications, 163(5), 36-43.

[6] Parmar, I., Agarwal, N., Saxena, S., Arora, R., Gupta, S., Dhiman, H., Chouhan, L. (2018, December). Stock market prediction using machine learning. In 2018 first international conference on secure cyber computing and communication (ICSCCC) (pp. 574-576). IEEE

[7] Nabipour, M., Nayyeri, P., Jabani, H., Shahab, S., Mosavi, A. (2020). Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis. IEEE Access, 8, 150199-150212.

[8] Yuan, X., Yuan, J., Jiang, T., Ain, Q. U. (2020). Integrated long-term stock selection models based on feature selection and machine learning algorithms for China stock market. IEEE Access, 8, 22672-22685..

[9] Chacón, H. D., Kesici, E., Najafirad, P. (2020). Improving financial time series prediction accuracy using ensemble empirical mode decomposition and recurrent neural networks. IEEE Access, 8, 117133-117145.

[10] Chen, S., Zhou, C. (2020). Stock prediction based on genetic algorithm feature selection and long short-term memory neural network. IEEE Access, 9, 9066-9072..

[11] Lai, C. Y., Chen, R. C., Caraka, R. E. (2019, July). Prediction stock price based on different index factors using LSTM. In 2019 International conference on machine learning and cybernetics (ICMLC) (pp. 1-6). IEEE..

[12] Lin, Y., Liu, S., Yang, H., Wu, H. (2021). Stock Trend Prediction Using Candlestick Charting and Ensemble Machine Learning Techniques With a Novelty Feature Engineering Scheme. IEEE Access, 9, 101433-101446

[13] Hiransha, M., Gopalakrishnan, E. A., Menon, V. K., Soman, K. P. (2018). NSE stock market prediction using deep-learning models. Procedia computer science, 132, 1351-1362..

[14] Devadoss, A. V., Ligori, T. A. A. (2013). Stock prediction using artificial neural networks. International Journal of Data Mining Techniques and Applications, 2(1), 283-291.

[15] Idrees, S. M., Alam, M. A., Agarwal, P. (2019). A prediction approach for stock market volatility based on time series data. IEEE Access, 7, 17287-17298