

# **STOCK PRICE PREDICTION USING DATA SCIENCE TECHNIQUES**

**A Project Report Phase – II**

Submitted in partial fulfillment of the requirements for the award of  
Bachelor of Engineering degree in Electronics and Communication  
Engineering

by

**Paluri Chandrasekhar Pavan (39130336)**

**Yerukala Chandrasekhar (39130528)**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING  
SCHOOL OF ELECTRICAL AND ELECTRONICS**

## **SATHYABAMA**

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

**Accredited with Grade "A" by NAAC**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI -  
600 119.**

**APRIL - 2023**



# **SATHYABAMA**

INSTITUTE OF SCIENCE AND TECHNOLOGY  
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC  
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119  
[www.sathyabama.ac.in](http://www.sathyabama.ac.in)

**DEPARTMENT OF ELECTRONICS AND COMMUNICAITON ENGINEERING**

## **BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the bonafide work of **Paluri Chandrasekhar Pavan (39130336)** and **Yerukala Chandrasekhar (39130528)** who carried out the project entitled "**Stock price prediction using data science techniques**" under our supervision from November 2022 to April 2023.

**Internal Guide**

**Dr. P. KAVIPRIYA, M.E., Ph.D.,**  
**ASSOCIATE PROFESSOR,**  
**ECE Dept**

**Head of the Department**

**Dr. T. RAVI, M.E., Ph.D.,**

---

Submitted for Viva voce Examination held on \_\_\_\_\_

**19.04.2023**

**Internal Examiner**

**External Examiner**

## DECLARATION

We, **Paluri Chandrasekhar Pavan (39130336)** and **Yerukala Chandrasekhar (39130528)**, hereby declare that the Project Report entitled “**STOCK PRICE PREDICTION USING DATA SCIENCE TECHNIQUES**” done by us under the guidance of **Dr. P. Kavipriya, M.E., Ph.D., Associate Professor, ECE Dept**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE: 19.04.2023

PLACE: Chennai

SIGNATURE OF THE CANDIDATE(S)

1. 
2. 

## ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA, M.E., Ph.D.**, Prof. & Dean, School of Electrical and Electronics and **Dr. T. RAVI, Ph.D.**, Prof. & Head, Department of **Electronics and Communication Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express my sincere and deep sense of gratitude to our Project Guide **Dr. P. KAVIPRIYA, M.E., Ph.D., ASSOCIATE PROFESSOR**, Department of **Electronics and Communication Engineering** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

## **ABSTRACT**

In this project, Persuasion of the topic is aimed at predicting the stability of future markets stocks. The movement of future market evolution has been studied by many researchers. Data fluctuates in stocks, so it becomes a source of efficiency. The purpose of this article is to propose a machine-learning-based model for predicting the stock price Index value from comparing supervise classification machine learning algorithms by predicting stock price increases or stable states. Moreover, the aim of this study is to compare and discuss the performance of various machine learning algorithms based on the data from the transport traffic department. For this project, logistic regression, random forest, decision trees, and naive bayes theorem were used to calculate stock price predictions. An evaluation of the proposed data science technique has been made with regard to identifying the confusion matrix, categorizing data based on priority and comparing it with the best levels of precision, recall and the F1 score.

# TABLE OF CONTENTS

| CHAPTER No. | TITLE                                                     | PAGE No. |
|-------------|-----------------------------------------------------------|----------|
| 1           | INTRODUCTION                                              | 1        |
| 2           | LITERATURE SURVEY                                         | 3        |
| 3           | EXISTING AND PROPOSED METHODOLOGY                         | 5        |
|             | 3.1 SCOPE OF THE PROJECT                                  | 5        |
|             | 3.2 AIM OF THE PROJECT                                    | 5        |
|             | 3.3 EXISTING SYSTEM                                       | 6        |
|             | 3.4 PROPOSED SYSTEM                                       | 7        |
|             | 3.5 WORK FLOW OF THE PROJECT                              | 8        |
| 4           | EXPERIMENTAL OR MATERIALS AND METHODS,<br>ALGORITHMS USED | 11       |
|             | 4.1 HARDWARE REQUIREMENTS AND SOFTWARE<br>REQUIREMENTS    | 11       |
|             | 4.2 INPUT DESIGN                                          | 11       |
|             | 4.3 OUTPUT DESIGN                                         | 11       |
|             | 4.4 FEASABILITY STUDY                                     | 12       |
|             | 4.5 ARTIFICIAL NEURAL NETWORK                             | 14       |
|             | 4.6 SUPPORT VECTOR MACHINE                                | 16       |
|             | 4.7 LINEAR REGRESSION ALGORITHM                           | 19       |
|             | 4.8 RANDOM FOREST ALGORITHM                               | 20       |
|             | 4.9 ARIMA ALGORITHM                                       | 22       |
|             | 4.10 METHODOLOGY OF ALGORITHMS                            | 23       |
|             | 4.11 COLLECTIVE WORKING OF ALGORITHMS                     | 25       |
|             | 4.12 XAMP SERVER                                          | 29       |
|             | 4.13 BACKEND DEVELOPMENT                                  | 30       |
| 5           | RESULTS, DISCUSSIONS AND PERFORMANCE<br>ANALYSIS          | 32       |
| 6           | SUMMARY AND RESULTS                                       | 46       |

|                |    |
|----------------|----|
| 6.1 SUMMARY    | 46 |
| 6.2 CONCLUSION | 46 |
| REFERENCES     | 48 |

## LIST OF TABLES

| FIGURE NO | TITLE                     | PAGE No. |
|-----------|---------------------------|----------|
| 4.1       | USE OF SVM IN THE PROJECT | 18       |

## LIST OF FIGURES

| FIGURE NO | TITLE                                           | PAGE No. |
|-----------|-------------------------------------------------|----------|
| 3.1       | BASIC ARCHITECTURE OF EXISTING SYSTEM           | 6        |
| 3.2       | PROPOSED SYSTEM ARCHIRECURE                     | 7        |
| 3.3       | COMMAND PROMPT                                  | 9        |
| 3.4       | SERVER DATABASE                                 | 10       |
| 3.5       | STOCK PRICES OUTPUT                             | 10       |
| 4.1       | ARTIFICAL NEURAL NETWORK                        | 15       |
| 4.2       | SUPPORT VECTOR MACHINE                          | 17       |
| 4.3       | LINEAR REGRESSION ALGORITHM                     | 20       |
| 4.4       | RANDOM FOREST ALGOITHM                          | 21       |
| 4.5       | ARIMA ALGORITHM                                 | 23       |
| 5.1       | MODEL RESULT OF STOCK PRICE PREDICTION          | 33       |
| 5.2       | OVERALL PERFORMANCE                             | 33       |
| 5.3       | RESULT OF APPLE INC STOCK PREDICTION            | 34       |
| 5.4       | RESULT OF RELIANCE INDS SROCK PREDICTION        | 34       |
| 5.5       | RESULT OF TATA CONSULTANCY STOCK<br>PREDICTION  | 35       |
| 5.6       | RESULT OF HDFC BANK STOCK PREDICTION            | 35       |
| 5.7       | RESULT OF INFOSYS LTD STOCK PREDICTION          | 36       |
| 5.8       | RESULT OF ICICI BANK STOCK PREDICTION           | 36       |
| 5.9       | RESULT OF HINDUSTAN UNIEVER STOCK<br>PREDICTION | 37       |
| 5.10      | RESULT OF HOUSING DEVEL FIN STOCK<br>PREDICTION | 37       |
| 5.11      | RESULT OF STATE BK OF INDIA STOCK<br>PREDICTION | 38       |
| 5.12      | RESULT OF ITC LTD STOCK PREDICTION              | 38       |

|      |                                                         |    |
|------|---------------------------------------------------------|----|
| 5.13 | RESULT OF BHARATI RS. 1.25 PPD UP STOCK<br>PREDICTION   | 39 |
| 5.14 | RESULT OF LIFE INSURA CORP OF INDIA STOCK<br>PREDICTION | 39 |
| 5.15 | RESULT OF BAJAJ FINANCE LTD STOCK<br>PREDICTION         | 40 |
| 5.16 | RESULT OF KOTAK MAHINDRA BAN STOCK<br>PREDICTION        | 40 |
| 5.17 | RESULT OF HCL TECHNOLOGIES STOCK<br>PREDICTION          | 41 |
| 5.18 | RESULT OF LARSEN & TURBO STOCK PREDICTION               | 41 |
| 5.19 | RESULT OF MARUTI SUZUKI IND STOCK<br>PREDICTION         | 42 |
| 5.20 | RESULT OF ADANI ENTERPRISES STOCK<br>PREDICTION         | 42 |
| 5.21 | RESULT OF SIEMENS INDIA LTD STOCK<br>PREDICTION         | 43 |
| 5.22 | RESULT OF AXIS BANK STOCK PREDICTION                    | 43 |
| 5.23 | RESULT OF WIPRO LTD STOCK PREDICTION                    | 44 |
| 5.24 | RESULT OF OIL & NATURAL GAS STOCK<br>PREDICTION         | 44 |
| 5.25 | RESULT OF TATA MOTORS LTD STOCK<br>PREDICTION           | 45 |

# **CHAPTER 1**

## **INTRODUCTION**

Forecasting the movement of stock prices has long been recognized as a challenging task in time series prediction. While numerous studies have addressed this issue, the most accurate findings are typically associated with mature financial markets. However, the uncertainty inherent in stock markets makes predicting trends or prices a difficult endeavour. There are two primary methodologies for stock analysis: fundamental analysis and technical analysis. Fundamental analysis entails assessing a company's financial performance, along with economic and political variables. On the other hand, technical analysis considers historical data such as past closing prices, highest and lowest prices, etc. to predict stock trends. Technical analysis is more feasible for prediction as it does not require measuring intrinsic value of stocks, but rather relies on analysing stock charts to identify patterns and trends. In the preliminary phase of studying stock market prediction, classical methods were initially employed. However, due to the non-stationary nature of stock market data as a time series, their effectiveness was limited. As a result, nonlinear techniques from the field of Data Science, such as Artificial Neural Networks (ANN) and Support Vector Machine (SVM), have been extensively utilized. In this project, we have employed both of these techniques to forecast stock trends and assessed their accuracy.

Predicting stock prices using data science techniques is a dynamic and exhilarating endeavour that aims to anticipate future trends in the stock market. Given the high volatility of the stock market, influenced by diverse factors such as political events, economic indicators, and market sentiment, accurate stock price prediction can offer invaluable insights for investors and traders, enabling them to make informed decisions and optimize their profits. This project employs a diverse range of data science techniques, including statistical analysis, machine learning, and time-series analysis, to construct predictive models capable of forecasting future stock market trends. The project entails extensive collection and analysis of historical data, such as stock prices, trading volumes, and news articles, to discern patterns and trends. Furthermore, the project encompasses the creation and assessment of the advanced

diverse predictive models leveraging well-known machine learning algorithms, including Random Forest, Support Vector Machines, and Neural Networks. The performance of these models will be evaluated using a range of metrics, such as Mean Absolute Error, Mean Squared Error, and R-squared, to ensure the reliability and precision of the predictions.

## **CHAPTER 2**

### **LITERATURE SURVEY**

In recent times, there has been a surge in research efforts aimed at predicting the trends and movements of the stock market. Scholars have delved extensively into this subject, with a plethora of studies conducted in academia. In this section, we will review the existing literature on Stock Market Prediction.

In the research study titled "Predicting stock and stock price index movement using Trend Deterministic Data Preparation and Data Science techniques" conducted by Jigar Patel, Sahil Shah, Priyank Thakkar, and K. Kotecha in 2016, the authors utilized various features, including Simple 10 Day Moving Average, Weighted 10 Day Moving Average, Momentum, Stochastic K%, for predicting stock market trends. The authors utilized Support Vector Machines (SVM) and Artificial Neural Networks (ANN) as their predictive models in the study. They reported an accuracy of 84% when employing SVM with a Polynomial function, and an accuracy of 80% when using SVM with an RBF (Radial Basis Function) kernel function. On average, they achieved an accuracy of 78.71% with SVM and 74.94% with ANN for stock market trend prediction. These findings highlight the potential of using data science techniques, particularly SVM and ANN, in predicting stock market trends and movements.

The results of the study showed that using ANN, they achieved an accuracy of 75.74% in predicting the direction of stock price index movement. On the other hand, using Polynomial SVM with a degree of 3 and a value of C set to 100, they obtained an accuracy of 71.52%. These findings suggest that ANN performed slightly better than Polynomial SVM in predicting stock price trends in the context of the Istanbul Stock Exchange data.

The study titled "A Survey on Stock Market Prediction Using Machine Learning Techniques" conducted by Polamuri Subba Rao, K. Srinivas, and A. Krishna Mohan addresses the challenges of forecasting stock prices in the dynamic and volatile stock market. The paper discusses various prediction techniques that have been employed for stock market forecasting, including Artificial Neural Network.

The advantages and disadvantages of these techniques are thoroughly analysed within the framework of the paper. The writers also conduct an assessment and comparative analysis of distinct parameters employed in predicting stock market behaviour, employed to evaluate its performance and discern trends. The study aims to provide insights into different techniques used for stock market prediction and highlights the potential of combining multiple methods to enhance prediction accuracy. It offers a comprehensive review of the existing literature in this field and presents a fresh approach to improve stock market prediction outcomes.

The study titled "Literature Survey on Stock Price Prediction Using Machine Learning" conducted by Anusha J Adhikar, Apeksha K Jadhav, Charitha G, and Karishma KH concludes that RNN (Recurrent Neural Networks) and LSTM (Long Short-Term Memory) libraries are effective in determining stock price trends in comparison to the actual market trend. However, the authors note that the Python libraries used in the training process may not be optimal in terms of training speed. The study suggests that mathematical functions used for training have faster speed and more detailed designs with significant improvements when tested under various situations. On the other hand, Python library functions are considered more adaptable. The authors also highlight that certain stock trends can be predicted based on general rules and regulations of the stock market, which is the reason behind the existence of private placement institutes. However, there is room for improvement in terms of optimizing neural network parameters and the training process. The study acknowledges that further steps in the research would involve addressing these areas of improvement and optimizing the neural network parameters and training process. This suggests that the authors recognize the need for ongoing research and improvements in the field of stock price prediction using machine learning techniques.

## **CHAPTER 3**

### **EXISTING AND PROPOSED METHODOLOGY**

#### ***3.1 SCOPE OF THE PROJECT***

Investing in the stock market is a formidable undertaking. It entails the potential for both financial loss and substantial gains. Thus, an ideal model that encapsulates stock market trends would be invaluable. Leveraging advanced Data Science techniques, we can make use of predictive analytics to forecast the direction of the stock market, enabling informed investment decisions aligned with such projections.

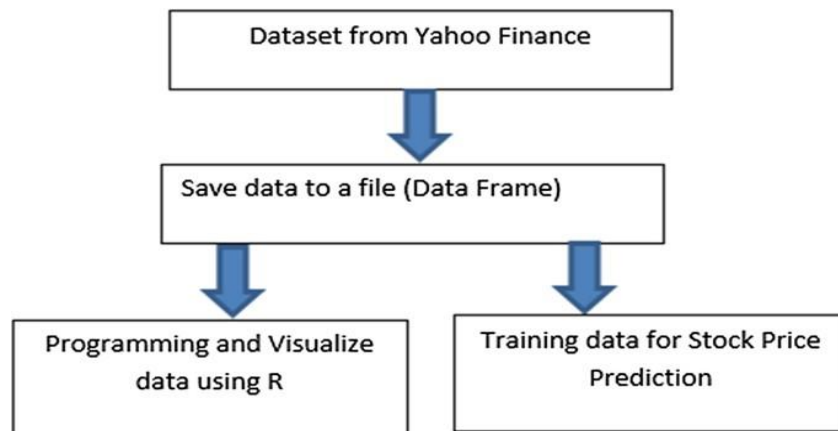
#### ***3.2 AIM OF THE PROJECT***

The goal of this project, centred on stock price prediction through the utilization of cutting-edge data science techniques, is to create a precise and dependable predictive model capable of forecasting future trends in the stock market. Employing a diverse array of data science methodologies, including statistical analysis, machine learning, and time-series analysis, the project endeavours to construct predictive models that can discern patterns and trends within historical stock market data.

The primary aim of implementing this project is to accurately forecast the current condition of the stock market based on the previous day's stock market value. This predictive algorithm can provide valuable insights to users, enabling them to gauge the financial health of a company and assess whether its stock is performing well or not. By analysing the stock's trajectory, whether it is on an upward trend or declining, this project facilitates stock exchange forecasts. While various algorithms exist for stock market prediction, Support Vector Machines (SVM) stands out for its efficiency in delivering stock values in a timely manner. This project is designed to operate within the realm of big data, leveraging data from global financial websites with specific attributes. These attributes include the previous day's opening stock price, high value, low value, volume, and closing price. This data is obtained and processed into .CSV or .XLXV files, which contain substantial amounts of data that serve as the database for prediction. The algorithm leverages this data to generate predictions for the target stock in the market.

### 3.3 EXISTING SYSTEM

Time series forecasting is a burgeoning field of research that presents unique challenges, particularly in the financial realm. One commonly employed technique is Support Vector Regression, which is a variant of Support Vector Machine used for nonlinear regression tasks by constructing input-output mapping functions. A more advanced version of SVR is Least Squares Support Vector Regression, which offers reduced computational complexity and improved efficiency compared to standard SVR. Another promising approach is the Firefly Algorithm, which draws inspiration from nature to optimize various problems. However, the current system's limitations include its focus on analysing the stock market in Taiwan only, with limited generalization to other global markets. Additionally, it lacks the capability to directly import raw data, analyse multivariate time series, and lacks a user-friendly interface for distribution as a web app for personal use.



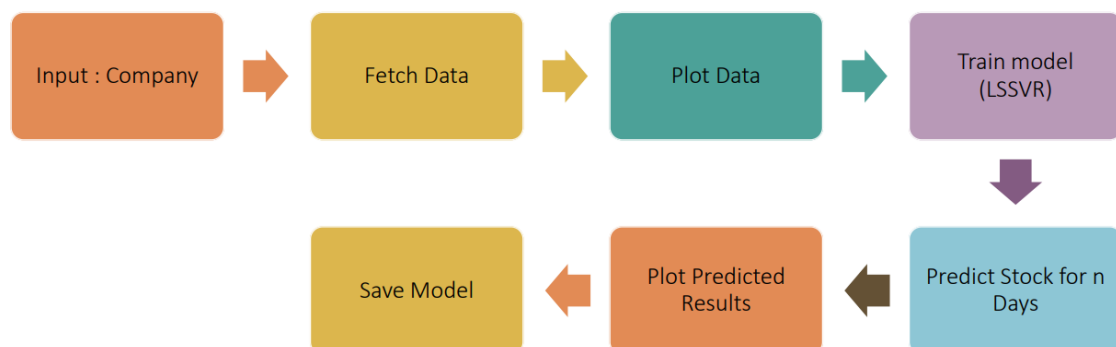
**Fig 3.1: Basic Architecture of Existing System**

The current system employs conventional techniques such as technical analysis and fundamental analysis for predicting stock prices. Technical analysis entails the examination of charts and technical indicators to identify patterns and trends in stock prices. Fundamental analysis encompasses the examination of financial statements, economic indicators, and industry trends to assess a company's stock value. However, the current system may only employ basic data science techniques like linear regression or moving averages, lacking the incorporation of advanced algorithms. These advanced algorithms possess the potential to offer the existed and upcoming

superior accuracy and predictive capabilities in stock market analysis. In the existing system, stock prices are manually analyzed by experienced traders and analysts who rely on their intuition and expertise to make investment decisions. This approach is subjective and can lead to biased predictions. The existing system may focus on a few selected stocks or a specific industry, which limits its predictive power and usefulness. The existing system may not be fully automated, and it may require manual input and intervention from traders and analysts, which can be time-consuming and error-prone.

### 3.4 PROPOSED SYSTEM

To broaden the applicability of the existing system, our work aims to utilize the system for estimating stock prices in other emerging markets and mature markets. Additionally, we plan to enhance the system to analyse multivariate time series data and allow for direct importation of raw datasets. This would enable the system to maximize profits even in scenarios where the corporate stock market has lower value. Additionally, we are contemplating the creation of a web-based application to enhance the overall user-friendliness and usability of the expert system.



**Fig 3.2 : Proposed System Architecture**

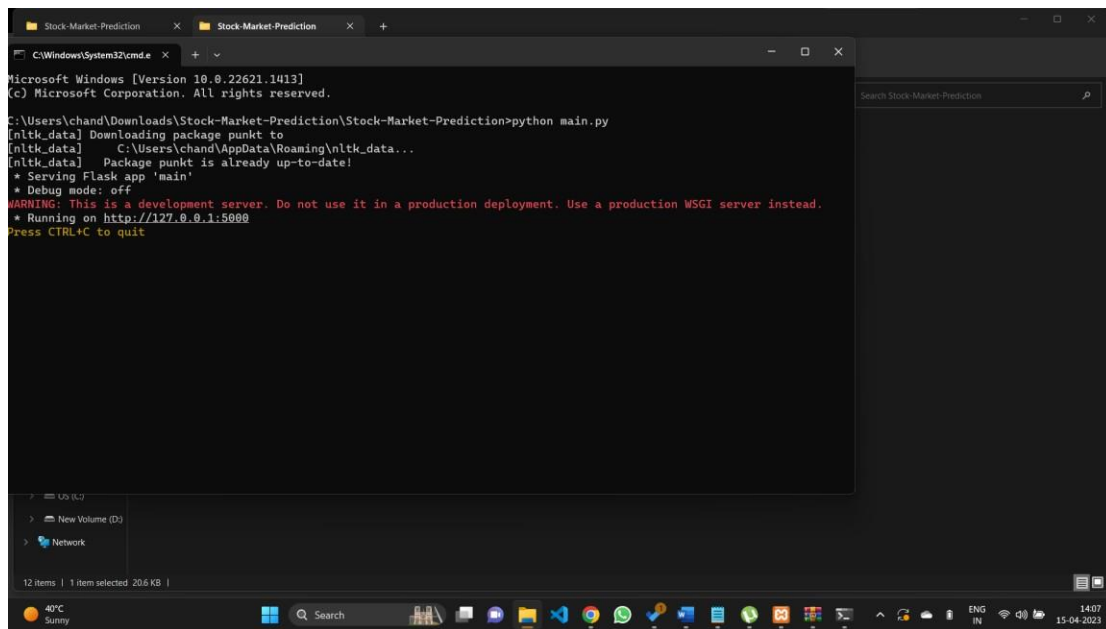
Deciding where and how to invest one's money is a common challenge faced by many individuals. Despite the significant amount of money and interest involved in this decision, there is no single definitive answer on when and where to invest money. The system should gather and cleanse substantial amounts of historical data from diverse source, including stock exchanges, financial news websites, and social media platform.

The data should undergo pre-processing to eliminate duplicates, missing values, and outliers. Relevant features should be extracted from the pre-processed data, and the most significant features should be selected using methods like correlation analysis and feature importance ranking. The pre-processed data must be divided into training and testing sets, with the training set designated for training the Random Forest, Linear Regression, and ARIMA algorithms. The testing set will then be used to assess the performance of these models. The system should develop and optimize the Random Forest, Linear Regression, and ARIMA models using various hyperparameter tuning techniques like Grid Search and Random Search. The performance of the models should be assessed using evaluation metrics.

### ***3.5 WORK FLOW OF THE PROJECT***

We utilized Flask and Python to construct a website that serves as a platform to showcase the output of our project. By leveraging frontend technologies, we were able to create a visually appealing design outlet. To make the website accessible to users, we hosted it through WordPress and GitHub Pages. For managing the backend, we employed MySQL and XAMPP servers. This backend setup proves to be useful when users require data output for large stocks within a specific time period. After writing the code with various algorithms, we conducted thorough testing to ensure the functionality of the data training process. Once we obtained promising results, we proceeded to train the data using a combination of all three algorithms. The accuracy rate was found to be satisfactory after implementing these measures. To handle the database management between the frontend and backend, we utilized REST APIs and chose XAMPP server with MySQL for this purpose. Finally, we hosted the website through WordPress for seamless access by users. In recent years, stock prediction has gained significant attention from both researchers and practitioners. Machine learning algorithms, such as linear regression, have demonstrated effectiveness in predicting stock prices. This report specifically concentrates on implementing the linear regression algorithm in Python for a stock prediction project, with the goal of developing a reliable and accurate model for high-precision stock price predictions and it also gives the better results from

precision stock price predictions. Additionally, the report delves into the utilization of the random forest algorithm, a popular machine learning technique, for stock price prediction in Python. Given that stock prices are influenced by various factors, such as economic conditions, company performance, and market trends, stock price prediction remains a challenging task. Time series analysis, a widely used approach in stock price prediction, is explored in this report through the implementation of the ARIMA (Auto Regressive Integrated Moving Average) algorithm in Python for a stock prediction project.

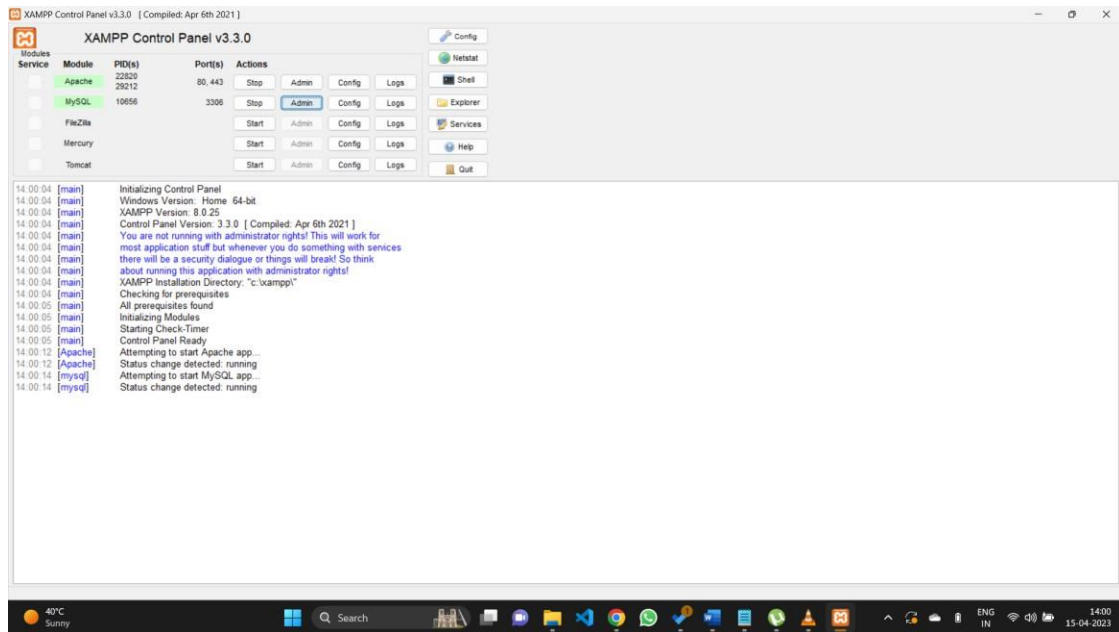


```
Microsoft Windows [Version 10.0.22621.1413]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chand\Downloads\Stock-Market-Prediction\Stock-Market-Prediction>python main.py
[nltk_data] Downloading package punct to
[nltk_data]   C:\Users\chand\AppData\Roaming\nltk_data...
[nltk_data]   Package punct is already up-to-date!
* Serving Flask app 'main'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

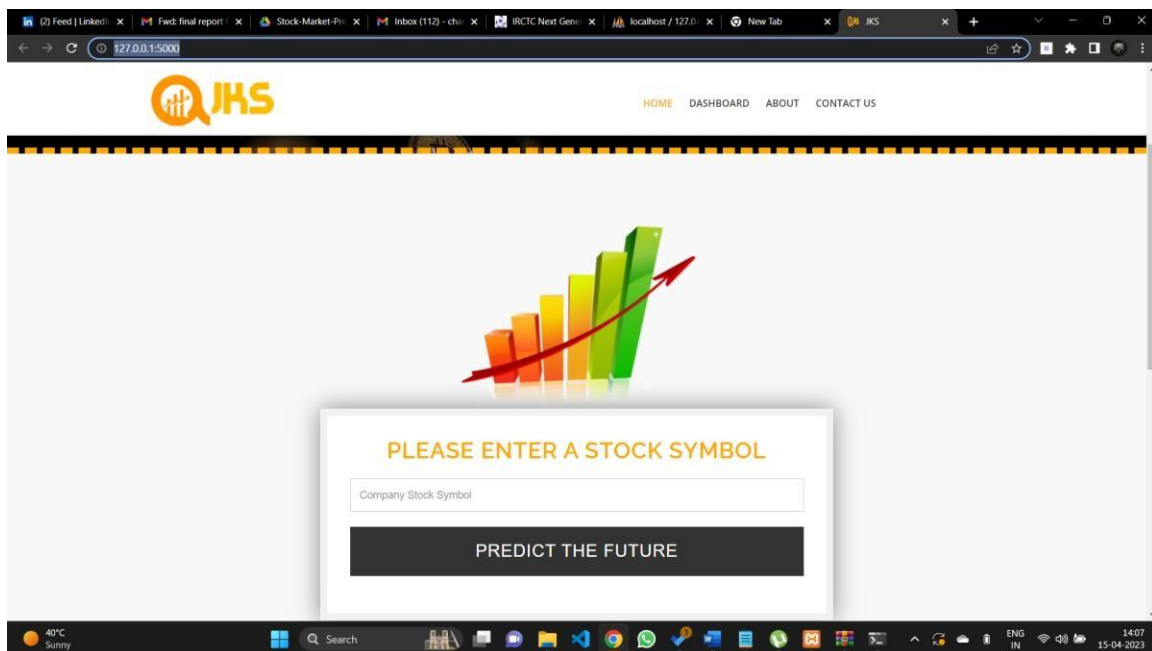
**Fig 3.3 : Command Prompt**

Once the primary code file is executed in the command prompt, the program will commence its execution, enabling the user to access the website by entering the provided address. As illustrated in the image, the user can click on the HTTP link while holding down the Ctrl key.



**Fig 3.4 : Server Database**

We need to enable the XAMPP server to manage the data with the help of MYSQL after clicking the start button on Apache and MySQL we need to make my SQL as an admin to start the project.



**Fig 3.5 : Stock Prices Output**

The output of the website after hosting through the word press.

## **CHAPTER 4**

### **EXPERIMENTAL OR MATERIALS AND METHODS, ALGORITHMS USED**

#### ***4.1 HARDWARE REQUIREMENTS AND SOFTWARE REQUIREMENTS***

- System - Pentium-IV
- Speed - 2.4GHz
- Hard disk - 40GB
- Monitor - 15" VGA colour
- RAM - 512MB
- Operating System - Windows XP
- Coding language - Python, SQL

#### ***4.2 INPUT DESIGN***

The process of input design serves as a crucial link between the information system and the end user, encompassing the development of specifications and procedures for data preparation in order to convert transaction data into a format that is usable for processing. This can be achieved through various methods, such as computer-based inspection to read data from written or printed documents, or direct inputting of data into the system through manual keying. The primary objectives of input design are to ensure effective control over the amount of input required, minimize errors, avoid delays, eliminate unnecessary steps, and maintain simplicity in the process. Additionally, input design should prioritize factors such as security, ease of use, and privacy retention, in order to ensure that the input process is efficient and user-friendly for the end users.

#### ***4.3 OUTPUT DESIGN***

The excellence of an output is determined by its capacity to fulfil the demands of the end user and effectively communicate information in a concise and comprehensible manner. Outputs serve as a means of communication between the system and its users or other systems. Output design also involves in determining how

information should be presented for immediate use, as well as generating hard copy outputs when necessary. Outputs are crucial as they serve as the primary source of information for users. Well-designed outputs can significantly impact the system's relationship with users, facilitating effective decision-making.

#### **4.4 FEASIBILITY STUDY**

In this phase, an assessment is carried out to evaluate the viability of the project, and a business proposal is presented, accompanied by a high-level project blueprint and preliminary cost projections. As part of the system analysis, a feasibility study is conducted to determine if the proposed system will impose an excessive burden on the company. Understanding the primary system requirements is critical for conducting the feasibility analysis. Three key factors that are considered during the feasibility analysis are:

- Financial Feasibility
- Technical Feasibility
- Social Feasibility

This analysis is conducted to evaluate the economic impact of the system on the organization. The company has limited budget for research and development, and expenditures must be justified. Therefore, the system development was kept within budget by utilizing freely available technologies, with only customized products requiring purchase. This ensures that the system aligns with the company's financial constraints while still achieving the desired objectives. This study is conducted to assess the technical feasibility of the system, which includes evaluating its technical requirements. It is important that the system does not excessively strain the available technical resources, as this would burden the client. Therefore, the developed system is designed to have modest resource requirements, with minimal or no changes needed for its implementation. This ensures that the system can be smoothly integrated into the existing technical infrastructure without imposing undue demands on the client's resources. The examination of the study revolves around evaluating the degree of system acceptance from the user's perspective. This encompasses the user training process aimed at enhancing system efficiency. It is imperative that the user does not perceive the system as a threat, but the user will also imperative with

rather embraces it as a vital tool. The extent of user acceptance relies heavily on the methodologies employed to educate and familiarize the user with the system. Boosting the user's confidence is crucial, as it enables them to provide constructive feedback, which is valuable as the ultimate end-user of the system. Unit testing encompasses the creation of test cases to verify the precise functioning of the internal program logic and validate that program inputs yield valid outputs. This type of testing covers all decision branches and internal code flow within individual software units of the application. It is performed after the completion of each individual unit, prior to integration, and is regarded as a form of structural testing that necessitates a thorough understanding of the internal code structure. Unit tests are conducted at the component level, focusing on specific business processes, applications, or system configurations.

Integration tests are specifically designed to verify if integrated software components function as a single cohesive program. These tests are event-driven and primarily focus on the overall outcomes of screens or fields. The purpose of integration testing is to demonstrate that, although the individual components may have passed unit testing, they work correctly and consistently when combined together. Integration testing aims to identify any issues that may arise from the interaction and combination of components, ensuring their proper integration into the overall system. Functional tests are designed to systematically demonstrate the availability of tested functions in accordance with the business and technical requirements, system documentation, and user manuals. Functional testing emphasizes the following aspects: Valid Input: Testing if the system accepts identified classes of valid input as expected. Invalid Input: Testing if the system rejects identified classes of invalid input as expected. Functions: Exercising the identified functions to ensure they perform as intended. Output: Testing the identified classes of application outputs to verify their accuracy. Systems/Procedures: Invoking interfacing systems or procedures to ensure seamless integration. System testing is conducted to validate that the entire integrated software system complies with the specified requirements. This entails testing the system

configuration to ensure consistent attainment of expected outcomes. An instance of system testing is the configuration-oriented system integration test, where the focus is on verifying the integration of diverse components or subsystems. System testing is guided by process descriptions and flows, with emphasis on predefined process links and integration points to ensure seamless operation and adherence to requirements.

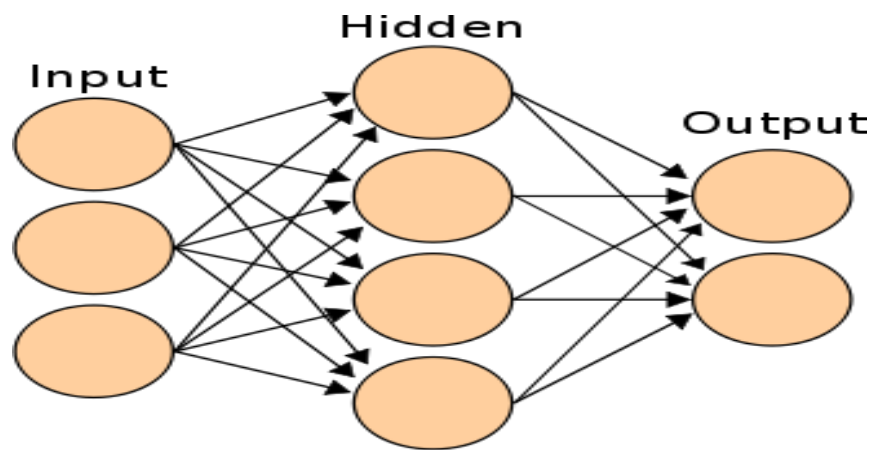
White box testing is a type of testing in which the software tester possesses knowledge about the internal mechanisms, structure, and programming language of the software, or at least its intended functionality. It is employed to evaluate aspects of the software that are not reachable or testable at a black box level, where only inputs and outputs are considered. White box testing allows for comprehensive testing of internal logic, code flow, and data flows within the software, and is often used to identify defects or vulnerabilities that may not be apparent through black box testing alone. It requires a deep understanding of the software's internal design and implementation, and is typically performed by software developers or testers with expertise in programming and system architecture. White box testing involves examining the internal logic and code of the software to identify any potential issues, validate the correct functioning of individual components, and ensure that the software operates as intended based on its internal design and structure.

Black box testing is a type of testing where the software is tested without any prior knowledge of its internal workings, structure, or programming language. Testers approach the software as a "black box," meaning they do not have visibility into the internal logic or code. Black box tests are typically designed based on definitive source documents, such as specifications or requirements documents. Test inputs are provided to the software and the corresponding outputs are observed, without considering how the software works internally. This type of testing solely focuses on the inputs and outputs, without delving into the internal implementation details of the software.

#### **4.5 ARTIFICIAL NEURAL NETWORK (ANN ALGORITHM)**

Artificial neural networks, often referred to as "neural" networks, are a fundamental

output layer, neural networks excel at identifying complex or diverse patterns that may be challenging for human programmers to define and teach the machine to recognize. Although neural networks, also known as "perceptron's," have existed since the 1940s, they have gained significant traction in artificial intelligence in recent years due to the development of a technique called "backpropagation." Another crucial advancement is the emergence of deep learning neural networks, where different layers of a multilayer network extract various high-level features until it can accurately identify the intended object. To grasp the concept of how a deep learning neural network learns, envision a factory assembly line. Once the raw materials (input data) are fed into the line, they are sequentially processed through different stops or layers, extracting different sets of high-level features.



***Fig 4.1 : Artificial Neural Network***

Once the network is trained, analysts can assign labels to the output and use backpropagation to correct any errors that may have occurred. Over time, the network can autonomously perform classification tasks without constant human intervention. The network comprises three layers, with the first two layers utilizing the rectified linear unit (ReLU) activation function, while the last layer uses the sigmoid activation function. Backpropagation with Gradient Descent Optimizer is employed to minimize error. ReLU is chosen as it creates sparser data when  $W \cdot X + B < 0$  and reduces the likelihood of vanishing gradients, resulting in faster learning. The choice of sigmoid or ReLU activation function in the output layer yields similar accuracy, but sigmoid will also be

selected for its ability to model non-linearity. Additionally, the dataset is normalized. The accuracy achieved in the artificial neural network (ANN) model is 71.73%. Backpropagation, also referred to as "error backpropagation," is a supervised learning algorithm used in artificial neural networks for gradient descent. It entails computing the gradient of the error function with respect to the network's weights. It extends the delta rule from perceptron's to multi-layer feedforward neural networks. The term "backward" originates from the fact that gradient computation progresses in reverse through the network, starting with the gradient of the last layer of weights and concluding with the gradient of the first layer of weights. This approach enables reuse of partial derivatives in calculating the gradient for previous layers, resulting in efficient gradient computation compared to independent computation for each layer. Backpropagation has regained popularity due to its widespread use in deep neural networks for tasks like image recognition and speech recognition. It is regarded as an efficient algorithm, with modern implementations leveraging specialized GPUs for further performance optimization.

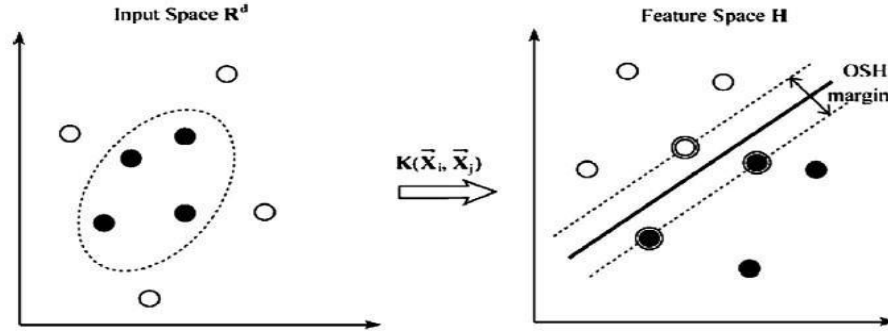
#### **4.6 SUPPORT VECTOR MACHINE**

Support Vector Machine is a supervised learning approach introduced by Vapnik that serves as a versatile model for both classification and regression tasks. For regression, Support Vector Regressor is employed. The primary objective of SVM is to achieve maximum separation of data points from a hyperplane. A hyperplane acts as a plane that partitions data points based on their labels. Apart from linear classification, SVM has the ability to classify non-linear data by utilizing the kernel trick. The kernel trick involves transforming data into a higher-dimensional space where it becomes linearly separable. In essence, SVM defines a hyperplane in a higher-dimensional space that is capable of classifying or dividing data points. Some commonly used kernel functions include the polynomial kernel, Gaussian kernel, Radial Basis Function (RBF) kernel, linear kernel, and sigmoid kernel.

For binary classification, suppose  $x_i$  be the input vectors and  $y_i$  be the corresponding output labels. Here for binary classification  $y_i \in \{1, -1\}$ . SVM maps the input vector into more dimensional space  $\phi(x_i)$  and make a hyperplane with

maximized distance between two classes. Mapping is done by a kernel function  $K(x_i, x_j)$ . And the final classifier is given in the equation below.

$$f(x) = \text{sgn} \left( \sum_{i=1}^N y_i \alpha_i \cdot K(x, x_i) + b \right)$$



**Fig 4.2 : Support Vector Machine**

$$\begin{aligned} &\text{Maximize } \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j \cdot y_i y_j \cdot K(x_i, x_j) \\ &\text{subject to } 0 \leq \alpha_i \leq c \\ &\sum_{i=1}^N \alpha_i y_i = 0 \quad i = 1, 2, \dots, N \end{aligned}$$

Where  $c$  is a regularization perimeter, which suggest the trade off between distance of hyperplane from the classes and misclassification error. Larger value  $c$ , smaller margin hyperplane and less misclassification and smaller the value of  $c$ , larger margin hyper plane but they may misclassify the bounty points.

$$\phi_j(x) = \exp\left\{-\frac{\|x - \mu_j\|^2}{2\sigma_j^2}\right\}$$

Here  $x$  is the input vector and  $\mu$  and  $\sigma$  represent the centre and spread of dataset points. To adjust the  $\mu$  and  $\sigma$  below formula is used.

$$\mu_{Class_k} = \frac{1}{|Class_k|} \sum_{x^{(n)} \in Class_k} x^{(n)}$$

$$\sigma_{Class_k}^2 = \frac{1}{|Class_k|} \sum_{x^{(n)} \in Class_k} ||x^{(n)} - \mu_k||^2$$

**TABLE 4.1 : USE OF SVM IN THE PROJECT**

| NO. | Kernel Function | d | C   | Avg. Accuracy |
|-----|-----------------|---|-----|---------------|
| 1   | Polynomial      | 2 | 100 | 52.79%        |
| 2   | Polynomial      | 3 | 100 | 52.79%        |
| 3   | Radial Basis    | - | 100 | 51.52%        |

The formulation for Least Squares Support Vector Regression (LSSVR) involves reformulating the minimization problem as a least squares problem. The kernel function used in LSSVR is typically the Radial Basis Function

$$\min J_2(w, b, e) = \frac{\mu}{2} w^T w + \frac{\zeta}{2} \sum_{i=1}^N e_{c,i}^2,$$

(RBF).

$$K(x, x_i) = \exp\left(-\|x - x_i\|^2 / \sigma^2\right),$$

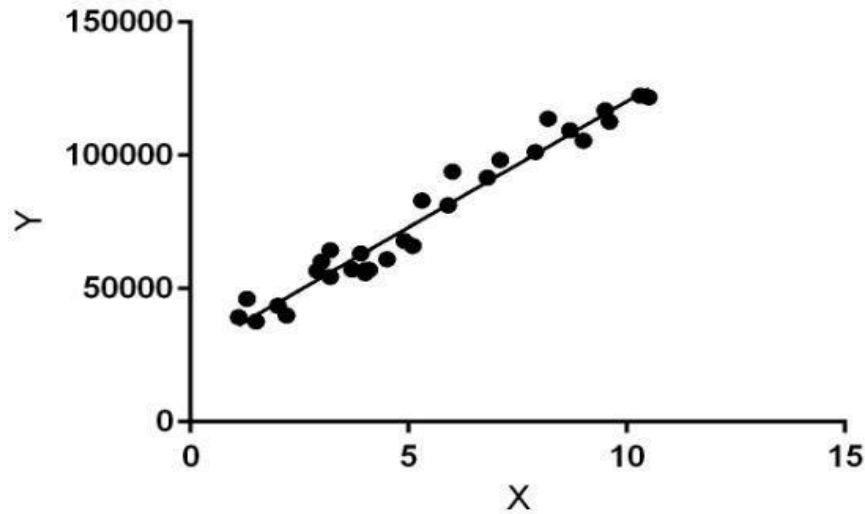
The Least Squares Support Vector Regression (LSSVR) algorithm, introduced by Suykens in 2001, stands out from Support Vector Regression (SVR) by employing

equality constraints instead of inequality constraints and utilizing a least squares objective function. This modification results in a significant reduction in computational complexity and improved efficiency compared to the standard SVM, as LSSVR involves solving linear equations instead of a quadratic programming problem.

#### **4.7 LINEAR REGRESSION ALGORITHM**

Linear regression is a statistical technique utilized to model the association between a dependent variable and one or more independent variables. In the context of stock prediction, the stock price acts as the dependent variable, while the independent variables can encompass a diverse range of factors such as market trends, company performance, and economic indicators. Python's scikit-learn library can be employed to implement the linear regression algorithm. The steps for implementing linear regression algorithm in Python are as follows:

Import libraries: The first step is to import the necessary libraries, including pandas, numpy, and scikit-learn. Load the data: The next step is to load the data into a pandas data frame. The data should include the stock prices as well as the independent variables that will be used for prediction. Split the data: This separation is vital for unbiased assessment of the model's generalization capabilities and to guard against overfitting. The training set enables the model to learn from the data, while the testing set serves as an independent benchmark for model evaluation. Proper data partitioning is a crucial step in machine learning to ensure reliable and accurate assessment of the model's performance. Feature scaling: The independent variables should be scaled to ensure that they are on the same scale. This is important to avoid bias towards any particular variable. Fit the model: The linear regression model can be fitted to the training set using the scikit-learn library. Predict the stock prices: Once the model is fitted, it can be used to predict the stock prices on the testing set. Evaluate the model: The model's performance can be evaluated by comparing the predicted stock prices with the actual stock prices.



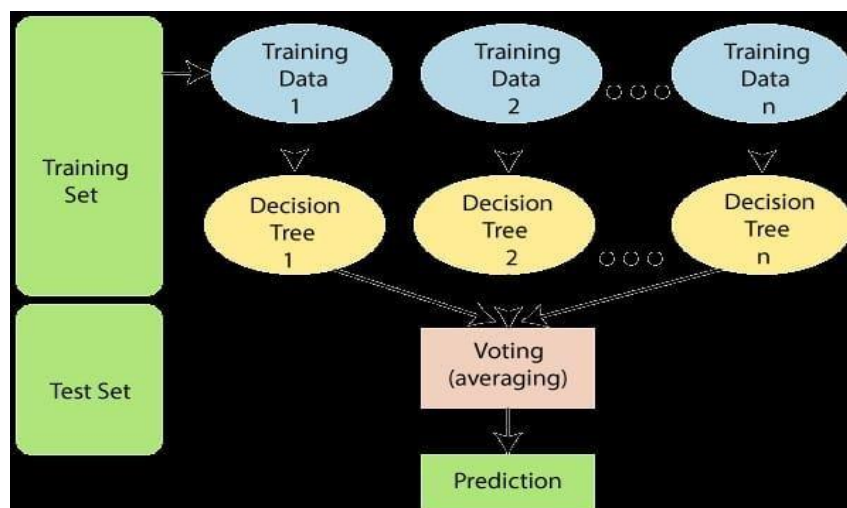
**Fig 4.3 Linear Regression Algorithm**

Linear regression is a powerful tool for stock prediction projects, and it can be easily implemented using Python's scikit-learn library, which offers a user-friendly interface. By following the steps outlined above, one can develop a linear regression model for stock prediction in Python. However, it is crucial to acknowledge that stock prediction is a complex task, and no single algorithm can guarantee accurate results. Therefore, it is important to explore other machine learning algorithms and consider using a combination of algorithms for stock prediction. Diversifying the approach and experimenting with various techniques can enhance the accuracy and reliability of stock prediction models.

#### **4.8 RANDOM FOREST ALGORITHM**

The random forest algorithm is a type of ensemble learning technique that utilizes decision trees. It involves combining multiple decision trees, each trained on different subsets of the data. Each decision tree generates its own prediction, and the final prediction is obtained by averaging the individual predictions. This approach helps to reduce variance and enhance the accuracy of the model. To implement the random forest algorithm in Python for stock price prediction, the following steps can be followed: Import libraries: Load the data: The next step is to load the data into a pandas data frame. The data should include the stock price as well as the independent variables

that will be used for prediction and also it significantly loads the data from the listed set. Split the data: The training set is used to train the model, while the testing set is utilized to evaluate the performance of the trained model. This approach helps to ensure that the model is able to generalize well to unseen data, as it is evaluated on data that was not used during training. Feature scaling: The independent variables should be scaled to ensure that they are on the same scale. This is important to avoid bias towards any particular variable. Fit the model: The random forest model can be fitted to the training set using the scikit-learn library. After fitting the model, it can be utilized to make predictions on the stock prices dataset. Evaluate the model: The performance of the model can be assessed by comparing the predicted stock prices with the actual stock prices, allowing for an evaluation of the model's accuracy and effectiveness.



**Fig 4.4 Random Forest Algorithm**

Random forest algorithm has several advantages over other machine learning algorithms. One of the biggest advantages is that it can handle a large number of independent variables without overfitting. It is also robust to outliers and missing data. In conclusion, random forest algorithm is an effective tool for stock price prediction project. Python's scikit-learn library offers a convenient interface to implement the steps outlined above for building a random forest model for stock price prediction. It's important to note, however, that stock price prediction is a challenging task, and no single algorithm can guarantee accurate results. Therefore, it's advisable to explore

other machine learning algorithms and consider using a combination of approaches for stock price prediction to improve the overall accuracy and effectiveness of the model.

#### **4.9 ARIMA ALGORITHM**

ARIMA, a statistical technique, utilizes a combination of autoregressive (AR), integrated (I), and moving average (MA) elements to model time series data. The autoregressive (AR) component considers the influence of past values on the current value, while the moving average (MA) component captures the impact of past errors on the current value. The integrated (I) component represents the number of differencing operations needed to transform the data into a stationary form, which is a key requirement for time series analysis. By combining these elements, ARIMA provides a robust approach to modeling and forecasting time series data, making it a widely used technique in various fields such as finance, economics, and weather forecasting. The steps for implementing ARIMA algorithm in Python for stock price predictions as follows.

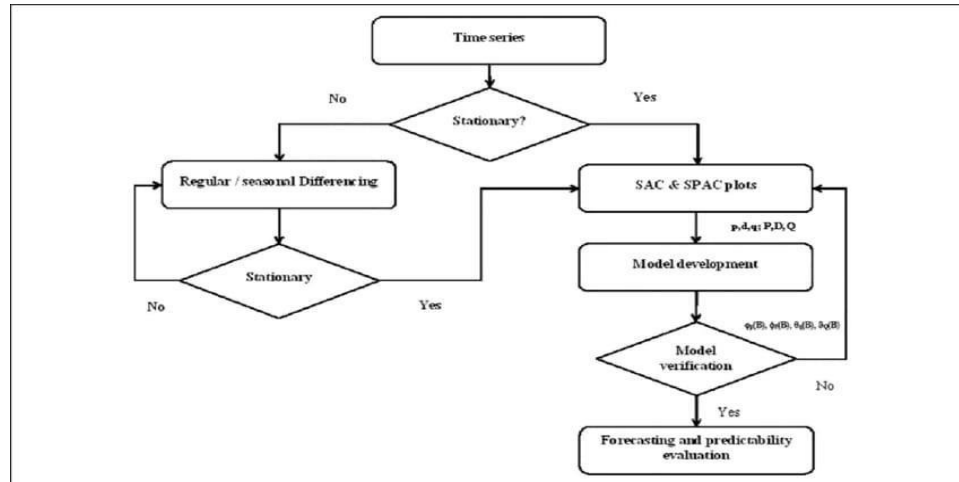
**Import libraries:** The first step is to import the necessary libraries, including pandas, NumPy, matplotlib, and stats models and it import the data from libraries.

**Load the data:** The next step is to load the data into a pandas data frame. The data should include the stock prices for the time period being analysed.

**Data pre-processing:** The data should be pre-processed to ensure that it is stationary. This can be done by applying differencing or logarithmic transformation.

**Train the model:** The ARIMA model can be trained on the pre-processed data using the stats models library and helps in developing the data pre-processing extraction.

**Test the model:** After training the ARIMA model, it can be utilized to generate forecasts for the test dataset. The performance of the model can be evaluated using various metrics like mean squared error, root mean squared error (RMSE), and other diverse measures. To anticipate future stock prices, the tested ARIMA model can be utilized for forecasting. The projected stock prices can be visualized utilizing the matplotlib library, allowing for graphical representation of the predicted trends.



**Fig 4.5 Arima Algorithm**

ARIMA algorithm has several advantages over other machine learning algorithms. One of the biggest advantages is that it is based on statistical methods and does not require a large amount of data. It can also handle non-linear trends and seasonality in the data. In conclusion, ARIMA algorithm is an effective tool for stock price prediction project. It can be implemented using Python's stats models library, which provides an easy-to-use interface. By following the steps outlined above, one can implement an ARIMA model for stock price prediction in Python. Therefore, it is important to explore other machine learning algorithms as well and use a combination of algorithms for stock price prediction.

#### **4.10 METHODOLOGY OF ALGORITHMS**

Prior to delving into the intricacies of individual algorithms, let us initially grasp the fundamental concept underpinning the prognosis of stock prices. Stock prices are impacted by diverse factors encompassing a company's financials, market trends, and geopolitical events. Foretelling the value of a stock is an intricate undertaking, entailing the scrutiny of copious data and discerning patterns that can yield foresight into forthcoming price fluctuations. Utilizing data science methodologies, one can analyse past stock data and discern recurring patterns that offer insight into potential future price shifts. In this project, algorithms such as ARIMA, linear regression, and random forest can be employed to generate prognostications.

ARIMA is a well-known statistical algorithm utilized for time series forecasting, which decomposes data into autoregressive (AR), integrated (I), and moving average (MA) components. The AR component represents the dependence of the current value on past values, the MA component captures the dependency on previous errors, while the I component denotes data integration. As a robust algorithm for time series forecasting, ARIMA can be effectively utilized to model stock price data. In this project, we can leverage ARIMA to model historical stock data and make predictions regarding future prices. The ARIMA algorithm comprises five key steps: Data Preparation: The initial step is to prepare the historical data for analysis. This involves cleansing the data, addressing missing values, and potentially transforming the data. Model Identification: The subsequent step is to identify the optimal ARIMA model for the data. This entails examining the autocorrelation and partial autocorrelation functions of the data and determining the appropriate values for the  $p$ ,  $d$ , and  $q$  parameters. Model Estimation: The third step is to estimate the parameters of the ARIMA model using maximum likelihood estimation. Model Validation: The fourth step is to validate the ARIMA model by assessing its accuracy on a separate validation dataset. Model Forecasting: The final step is to leverage the ARIMA model to generate future predictions based on the identified patterns in the historical data.

#### **Linear Regression Algorithm :**

Linear regression is a statistical technique that models the correlation between a dependent variable and one or more independent variables. In the context of this project, linear regression can be employed to model the historical stock data and make predictions about future prices. The linear regression algorithm encompasses four key steps: Data Preparation: The initial step is to prepare the historical data for analysis. This entails data cleansing, handling missing values, and potential data transformations.

Model Estimation: The subsequent step is to estimate the parameters of the linear regression model using the historical data. Model Validation: The third step is to validate the linear regression model by assessing its accuracy on a separate validation dataset. Model Forecasting: The final step is to employ the linear regression model to generate future predictions based on the established parameters and patterns in the historical data.

#### **Random Forest Algorithm :**

Random forest is an ensemble learning algorithm that amalgamates multiple decision trees to generate predictions. In this project, random forest can be utilized to model the historical stock data and generate forecasts regarding future prices.

The random forest algorithm comprises four primary steps:

Data Preparation: The initial step involves preparing the historical data for analysis, which includes data cleansing, handling missing values, and potential data transformations. Model Training: Following that, the subsequent step involves training the random forest model using historical data, enabling it to glean insights from patterns and relationships within the dataset. Model Validation: The third step is to validate the accuracy of the random forest model by evaluating its performance on a separate validation dataset to ensure it is reliable and accurate. Model Forecasting: The final step is to leverage the trained random forest model to generate future predictions based on the established patterns in the historical data, providing insights into future stock price movements.

#### **4.11 COLLECTIVE WORKING OF ALGORITHMS**

To integrate the functionalities of three algorithms, an ensemble approach, such as stacking or ensemble learning, can be employed. This involves combining the predictions of the three algorithms to generate a final prediction. The fundamental concept behind stacking is to train multiple models on the same data, utilize their predictions as features, and then train a meta-model on these features to generate the ultimate prediction. In this project, ARIMA, linear regression, and random forest

algorithms can be trained on the historical stock data, using their predictions as features. Subsequently, a meta-model, such as a linear regression or a neural network, can be trained on these features to generate the final prediction. This ensemble approach can potentially enhance the accuracy and robustness of the stock price prediction model by leveraging the strengths of multiple algorithms. Let me provide a brief overview of each step: Data Preparation: This step involves cleaning the historical stock data, filling in any missing values, and transforming the data if needed. This ensures that the data is in a suitable format for training and prediction. Algorithm Training: In this step, ARIMA, linear regression, and random forest algorithms are trained on the prepared historical stock data. Each algorithm learns from the historical data to make predictions based on their respective modelling approaches. Prediction Generation: Using the trained algorithms, predictions are generated for future stock prices. Each algorithm provides its own set of predictions based on its individual modelling techniques. Feature Generation: The predictions generated by each algorithm in the previous step are used as features for the meta-model. These predictions serve as additional input features for the meta-model, which aims to combine the strengths of multiple algorithms. Meta-model Training: In this step, a meta-model, such as a linear regression or a neural network, is trained on the features generated from the predictions of the individual algorithms. The meta-model learns to combine the predictions of the base algorithms and make the final prediction. Model Validation: The ensemble model is validated by checking its accuracy on a validation dataset. This helps to assess the performance and reliability of the ensemble model in making predictions. Model Forecasting: Finally, the trained ensemble model is used to make future predictions of stock prices. The ensemble model leverages the combined insights

from the base algorithms to generate a more accurate and robust prediction for future stock prices. It's important to note that the performance of the ensemble approach depends on the quality of the data, the choice of algorithms, and the hyperparameter tuning of the models. Proper validation and evaluation techniques should be applied to ensure the accuracy and reliability of the ensemble model in stock price prediction.

The benefit of employing an ensemble methodology lies in its ability to alleviate the limitations of individual algorithms and yield more precise predictions. For instance, ARIMA could excel at capturing trends and seasonality, linear regression might excel at capturing linear relationships, and random forest could excel at capturing nonlinear relationships and interactions. By amalgamating the unique strengths of these algorithms, we can enhance the accuracy of our predictions. Algorithms are a series of predefined instructions designed to solve specific problems. In the field of data science and machine learning, algorithms are utilized to analyse data and make predictions. Here's a brief overview of how some commonly used algorithms work.

**ARIMA Algorithm:** ARIMA (Autoregressive Integrated Moving Average) is a time series forecasting algorithm. It models the relationship between an observation and its lagged values, as well as the difference between the observation and its past values. ARIMA is often used to make short-term predictions in finance and economics. The ARIMA algorithm involves several steps, including:

- Data Preparation:** The data is cleaned and prepared for analysis.
- Stationarity Testing:** The data is checked to see if it is stationary (i.e., has constant mean and variance over time).
- Parameter Estimation:** The parameters of the ARIMA model are estimated based on the data.
- Model Fitting:** The ARIMA model is fitted to the data.
- Prediction:** The model is used to make predictions for future time periods.

**Linear Regression Algorithm:** It facilitates predictions based on the established relationship among the variables. The Linear Regression algorithm encompasses several steps, which may include.

Data Preparation: The data is cleaned and prepared for analysis. Feature Selection: The most relevant features are selected to be used in the model. Model Fitting: The linear regression model is fitted to the data. Prediction: The model is used to make predictions for new data. Random Forest Algorithm: Random Forest is a machine learning algorithm utilized for both classification and regression tasks. It operates by generating multiple decision trees and aggregating their predictions to arrive at a final prediction. The Random Forest algorithm encompasses several steps, which may include. Data Preparation: The data is cleaned and prepared for analysis. Random Sampling: A subset of the data is randomly sampled to create multiple decision trees. Feature Selection: The most relevant features are selected to be used in the model. Model Fitting: Multiple decision trees are created and fitted to the data. Prediction: The predictions from each decision tree are combined to make a final prediction. Backend Usage Working Details: Backend development involves writing server-side code to handle requests from the client-side of a web application. Here's an overview of how backend development is typically done. Server Setup: The server is configured to handle requests from the client-side of the web application. This entails installing server software (such as WAMP), configuring the server settings, and establishing a directory structure for storing server files. Database Creation: A database is established to store data for the web application. This involves selecting a database management system (such as PostgreSQL), creating tables and columns for data storage, and defining relationships between tables. Backend Development: Backend code is written in a programming language such as Ruby or Java. This code is responsible for managing requests from the client-side of the web application, interacting with the database, and executing business logic. API Development: An API (Application Programming Interface) is created to facilitate communication between the client-side and server-side of the web application. This includes defining endpoints (e.g., "/login" or "/register") and implementing functions to handle requests to these endpoints.

Data Retrieval: Backend code is utilized to retrieve data from the database and return it to the client-side of the web application. This data can be in various formats, such as XML or CSV. Security Measures: Backend security measures are implemented to safeguard the web application and its data from potential attacks. These measures may include: Authentication: Users are required to authenticate with a username and password to access the web application, preventing unauthorized access to sensitive data. Encryption: Data is encrypted before storage in the database or transmission between the client-side and server-side of the web application, safeguarding against interception or theft. Input Validation: User input is validated to ensure it conforms to expected formats and does not contain malicious code (e.g., SQL injection attacks). Access Control: Access to specific parts of the web application is restricted based on user roles and permissions, preventing unauthorized access to sensitive data or functionalities. Logging and Monitoring: Logs are maintained of all web application requests, errors, and security incidents, aiding in identification and response to potential security breaches. Backend Responsibility: The backend is accountable for managing the server-side of the web application, including database interactions, business logic, and security. Algorithms can be integrated with the backend to provide advanced functionalities such as predictive analytics.

#### ***4.12 XAMPP SERVER (DATABASE SECTION)***

XAMPP is a cross-platform web server solution that includes Apache, MySQL, PHP, and Perl, making it a popular choice for web developers due to its ease of setup and use. Here are some unique ways in which XAMPP can be utilized in a project. Local Development: XAMPP enables hosting of web applications on a local machine for development and testing, allowing developers to test their applications locally before deploying to a remote server. Database Management: XAMPP includes MySQL for managing databases, allowing developers to create, modify, and query databases for their applications.

PHP Development: XAMPP includes PHP, a popular server-side scripting language for web development, facilitating local testing and debugging of PHP scripts. Content Management Systems: XAMPP can be used to host content management systems like WordPress, Drupal, and Joomla on a local machine, allowing developers to build and test CMS-based websites locally before deployment. E-commerce Development: XAMPP supports hosting of e-commerce platforms like Magento, OpenCart, and WooCommerce on a local machine, enabling developers to test their e-commerce websites locally before deploying to a remote server. In summary, XAMPP offers a convenient and user-friendly solution for web developers to set up a local server environment for testing and development purposes. It allows developers to work on their applications in a controlled environment before deployment to a production server, reducing the risk of errors and downtime.

#### **4.13 BACKEND DEVELOPMENT**

XAMPP is a versatile web server solution that encompasses Apache, MySQL, PHP, and Perl, making it a flexible choice for hosting web applications locally for development and testing purposes. XAMPP's ease of installation and usage makes it a popular option among web developers. XAMPP allows for hosting web applications on a local machine, facilitating local development and testing. This eliminates the need for remote server deployment and enables developers to assess their applications in a controlled environment before going live. MySQL, included in the XAMPP package, is a renowned open-source relational database management system that utilizes SQL as its query language. MySQL can be used to create, modify, and query databases for web applications, offering robust data management capabilities. Backend development entails crafting server-side code in languages such as PHP, Python, or Java to handle requests from the client- side of a web application. XAMPP and MySQL provide a local server environment that allows developers to write and test backend code on their own machine, interacting with the MySQL database through SQL queries. MySQL is a versatile tool that can be employed to create tables, insert, update, and delete data, and execute SQL queries to interact with the database. PHP or other programming languages can also be involved in developing the database program which will also

be used to connect to MySQL, retrieve data from the database, and display it on the front-end of the web application. A login system, for instance, can be implemented using PHP and MySQL, where user credentials are stored in a database table. PHP code can then execute queries to verify the existence of users in the database and validate their credentials. Successful validation initiates a login session, while incorrect information prompts an error message.

For example, a login system can be implemented using PHP and MySQL. The user credentials can be stored in a database table. When a user enters their username and password on the login form, the PHP code can execute a query to check if the user exists in the database. If the user account exists and the password provided is correct, the user is successfully logged in, and their session is initiated. However, if the user account does not exist or the password provided is incorrect, an error message is displayed to notify the user about the unsuccessful login attempt.

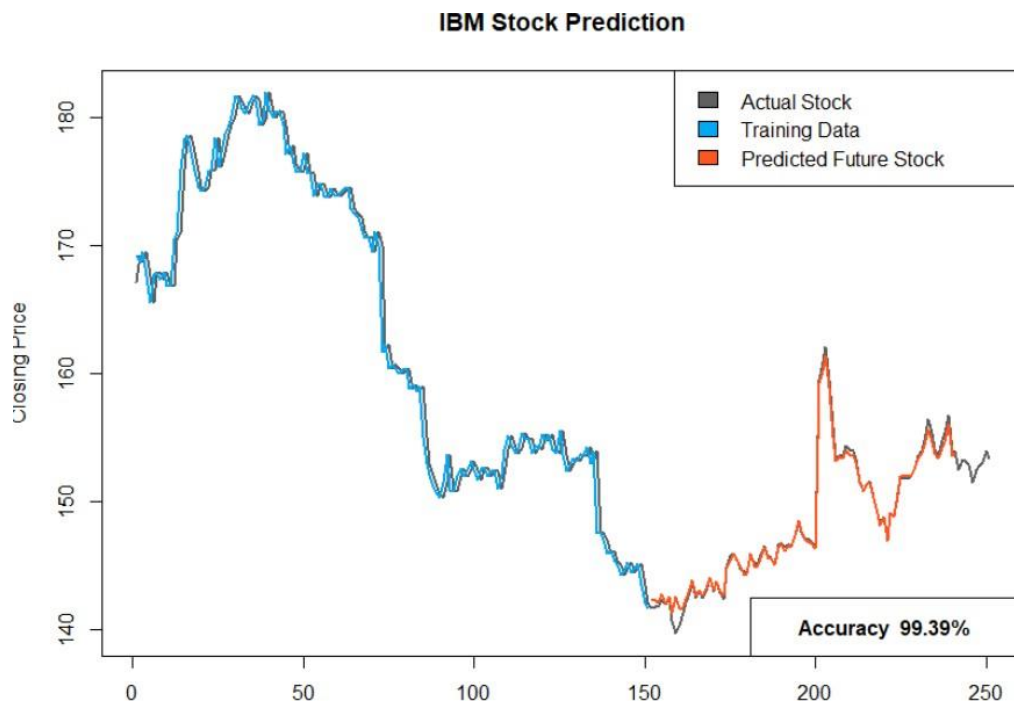
In summary, XAMPP and MySQL provide a user-friendly and powerful combination for backend development and database management in web development. XAMPP facilitates local server setup, while MySQL offers robust data management capabilities. The synergy of these technologies empowers developers to create efficient and secure web applications.

## **CHAPTER 5**

### **RESULTS, DISCUSSIONS AND PERFORMANCE ANALYSIS**

Stock price prediction using data science techniques such as machine learning and time series analysis can provide accurate and reliable results. These techniques can help in identifying the underlying patterns and trends in the data that affect the stock prices. Data science algorithms such as linear regression, random forest, and neural networks can be used for stock price prediction. These algorithms can take into account various factors such as economic conditions, company performance, market trends, etc., and can provide accurate predictions. Similarly, time series analysis techniques such as ARIMA, exponential smoothing, and seasonal decomposition can be used for stock price prediction. These techniques can help in modelling the past behaviour of stock prices and can be used to forecast future prices. The accuracy of the results obtained using data science techniques for stock price prediction depends on various factors such as the quality of the data, the choice of algorithm, and the amount of historical data available. However, with proper data preparation, feature engineering, and algorithm selection, data science techniques can provide accurate and reliable results for stock price prediction.

In conclusion, data science techniques such as machine learning and time series analysis can provide accurate and reliable results for stock price prediction. By using these techniques, one can identify the underlying patterns and trends in the data that affect the stock prices and can make informed decisions regarding stock investments.

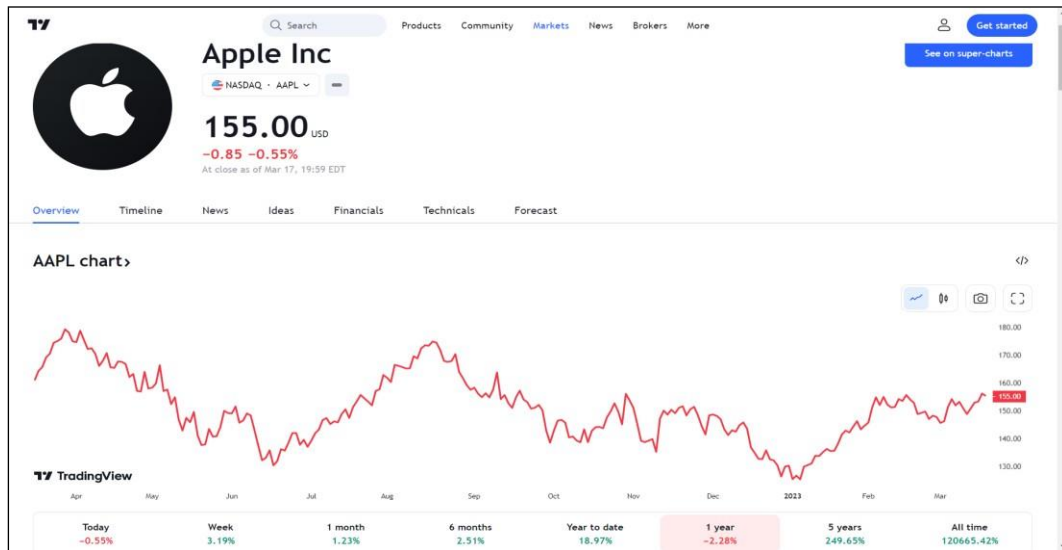


**Fig 5.1 : Model Result of Stock Price Prediction**

The above figure shows the model result of the stock price prediction on how the output for the price prediction would be for the companies.

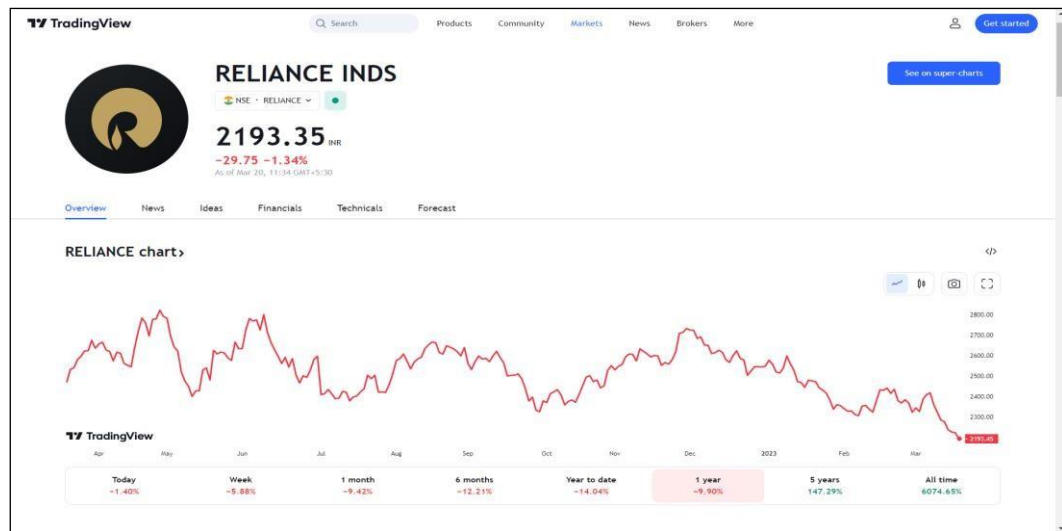


**Fig 5.2 : Overall Performance**



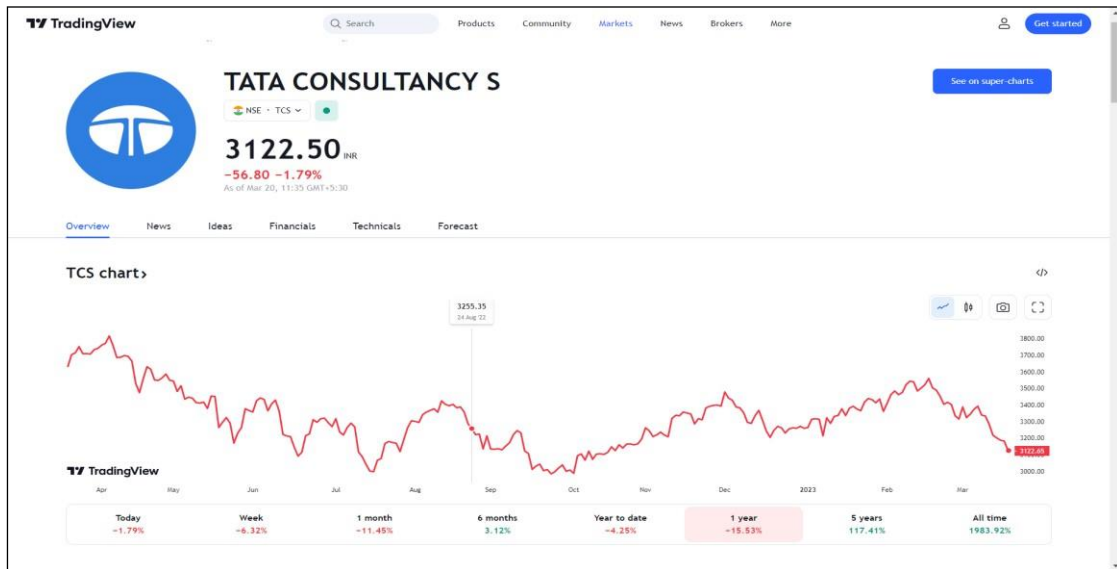
**Fig 5.3: Result of Apple Inc Stock Prediction**

The stock being presented is APPLE, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



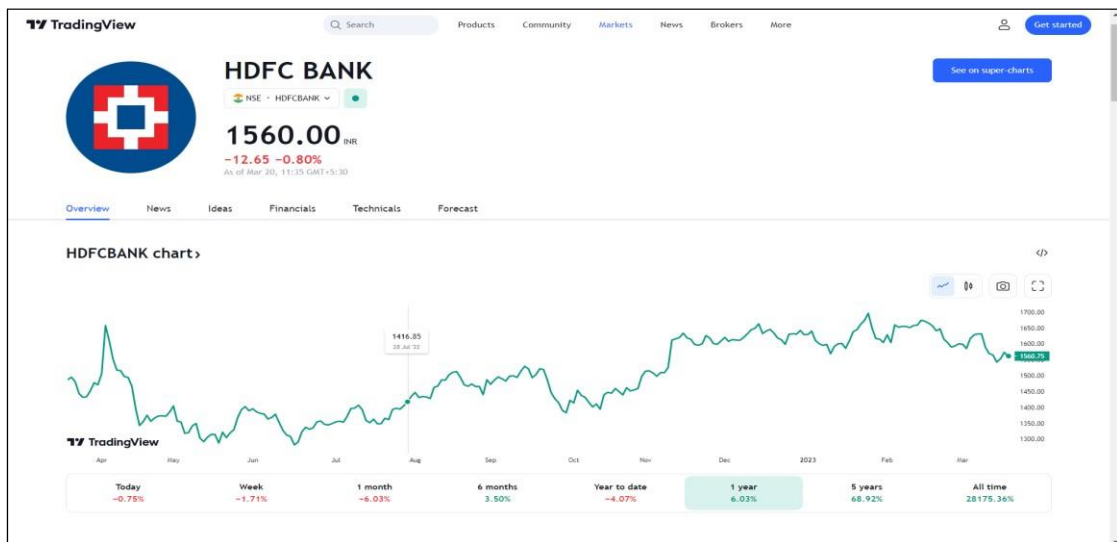
**Fig 5.4: Result of Reliance INDS Stock Prediction**

The stock being presented is RELIANCE, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



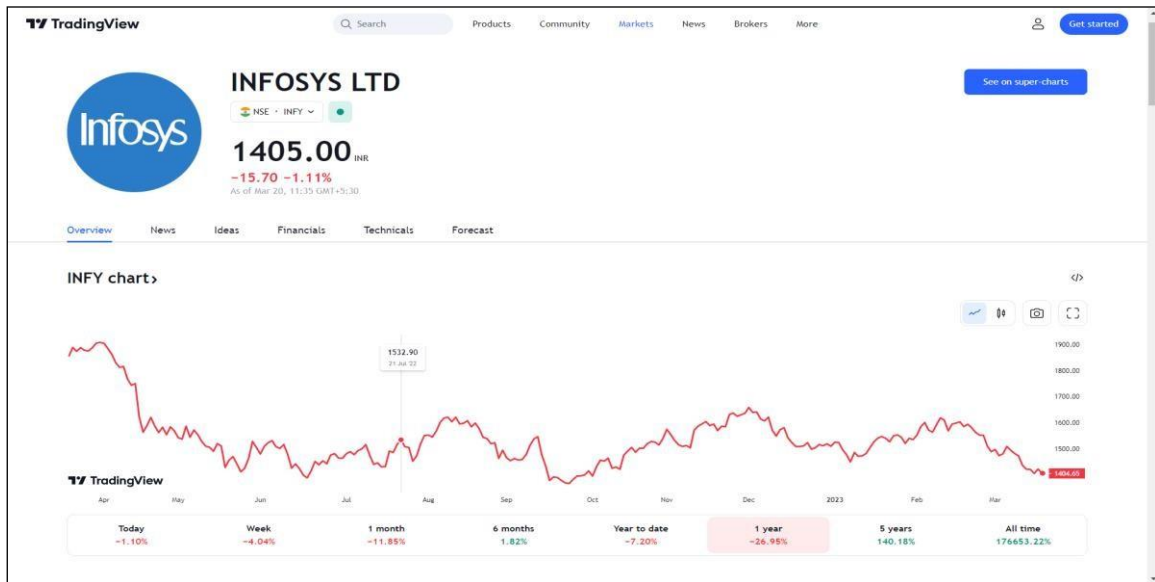
**Fig 5.5: Result of Tata Consultancy Stock Prediction**

The stock being presented is TATA, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



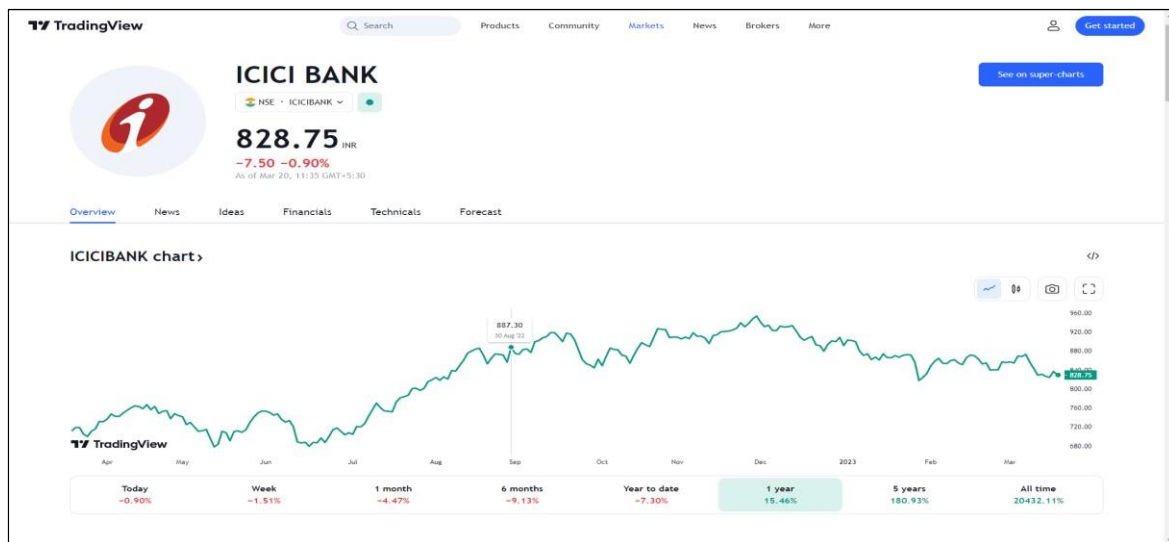
**Fig 5.6: Result of HDFC Bank Stock Prediction**

The stock being presented is HDFC, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



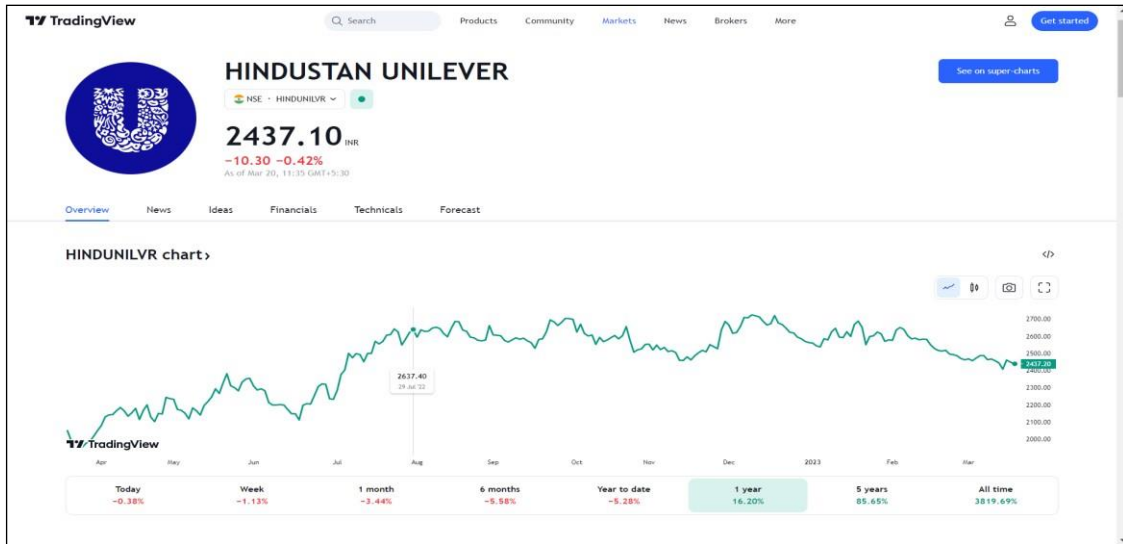
**Fig 5.7: Result of Infosys Ltd Stock Prediction**

The stock being presented is INFOSYS, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



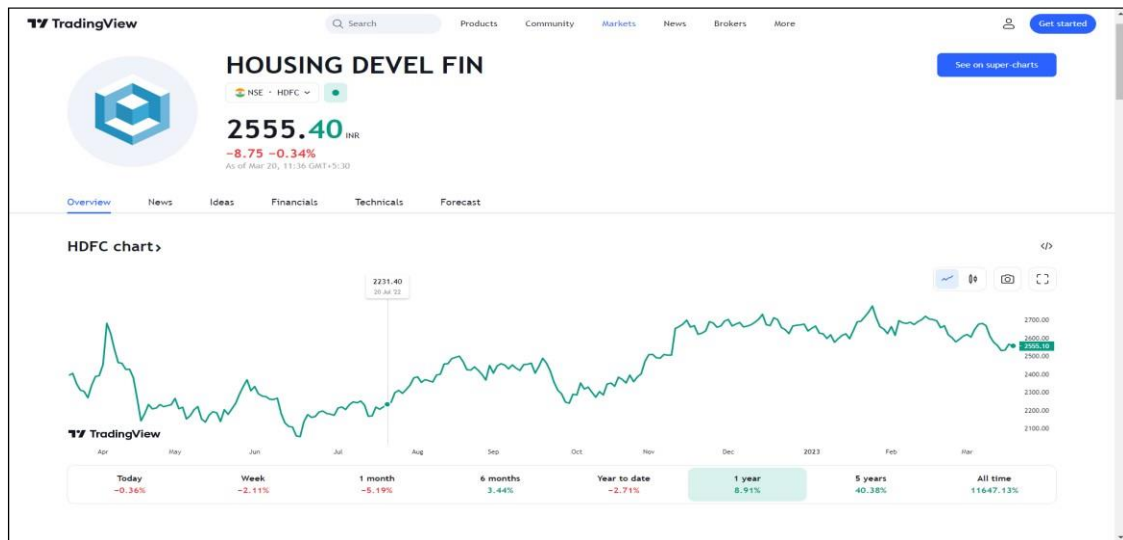
**Fig 5.8: Result of ICICI Bank Stock Prediction**

The stock being presented is ICICI, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



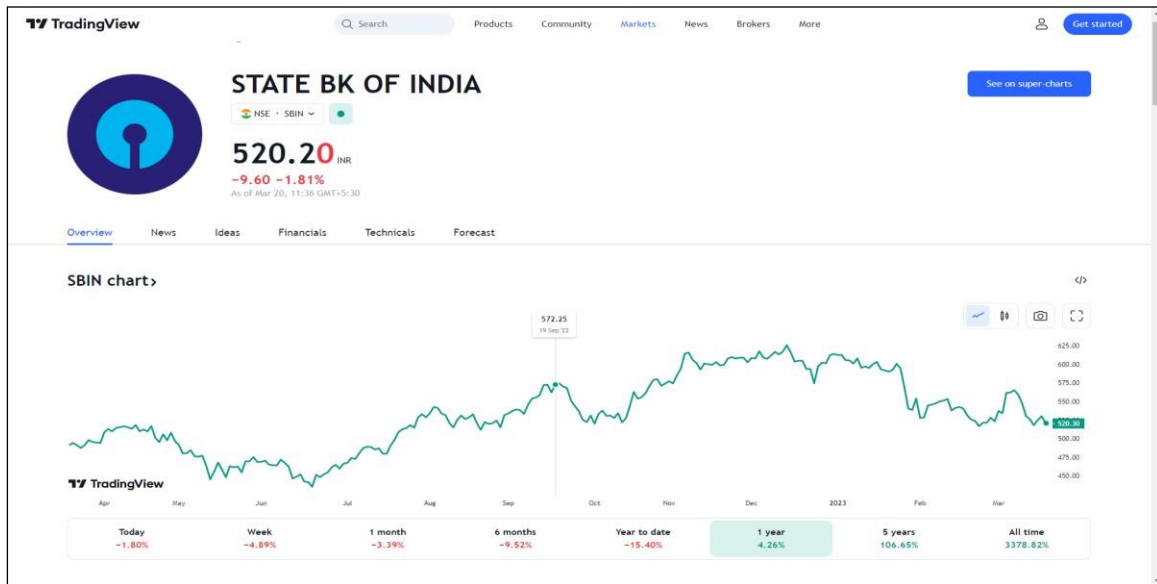
**Fig 5.9: Result of Hindustan Unilever Stock Prediction**

The stock being presented is UNILIVER, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



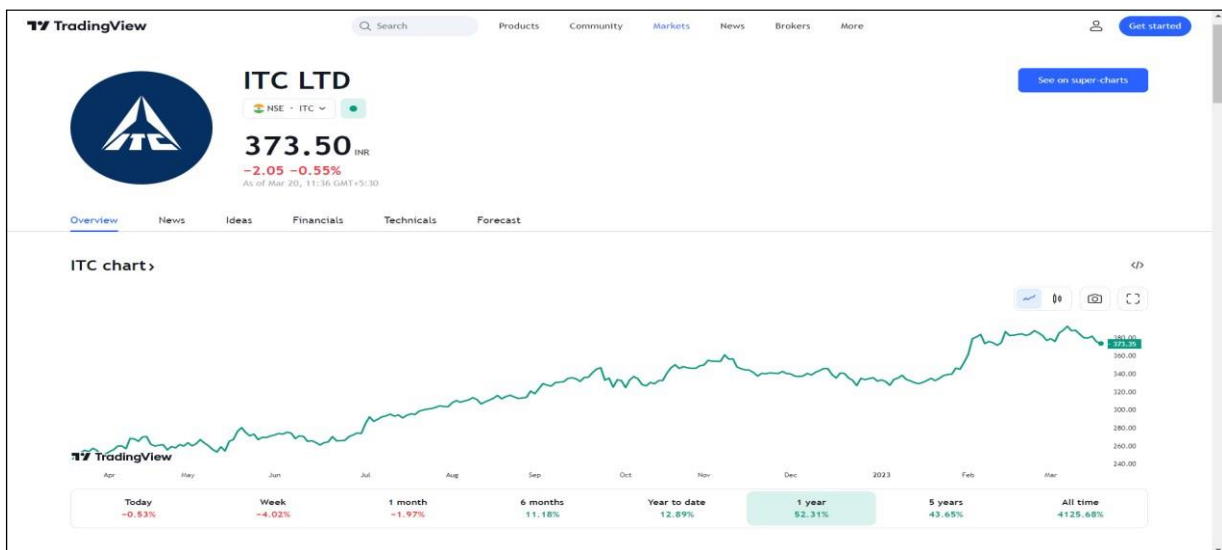
**Fig 5.10: Result of Housing Devel Fin Stock Prediction**

The stock being presented is HOUSING DEVEL FIN, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



**Fig 5.11: Result of State Bank of India Stock Prediction**

The stock being presented is SBI, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



**Fig 5.12: Result of ITC Ltd Stock Prediction**

The stock being presented is ITC, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



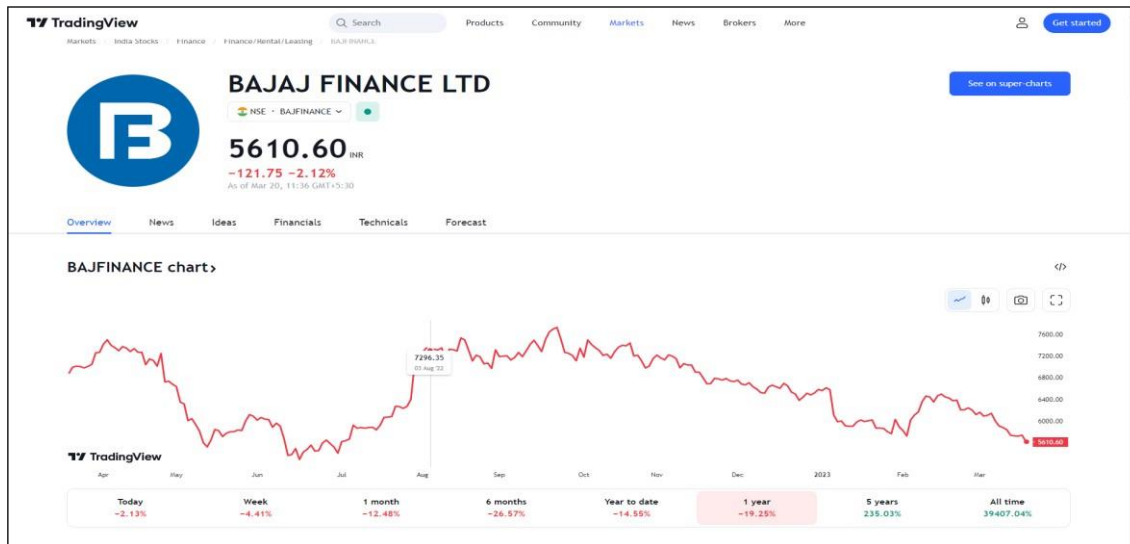
**Fig 5.13 : Result of Bharati Rs. 1.25 PPD Up Stock Prediction**

The stock being presented is AIRTEL, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



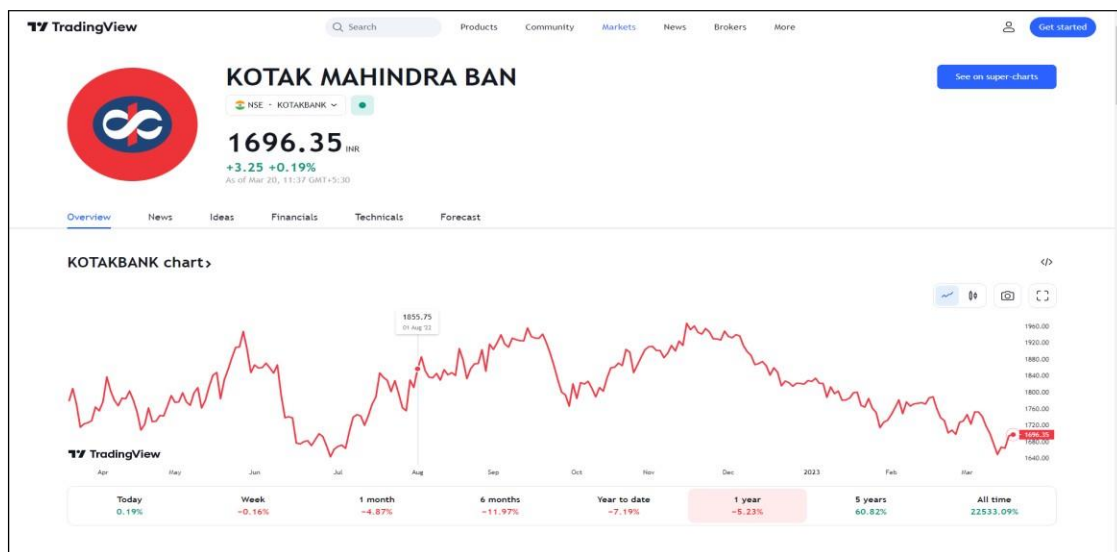
**Fig 5.14: Result of Life Insura Corp of India Stock Prediction**

The stock being presented is LIC, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage



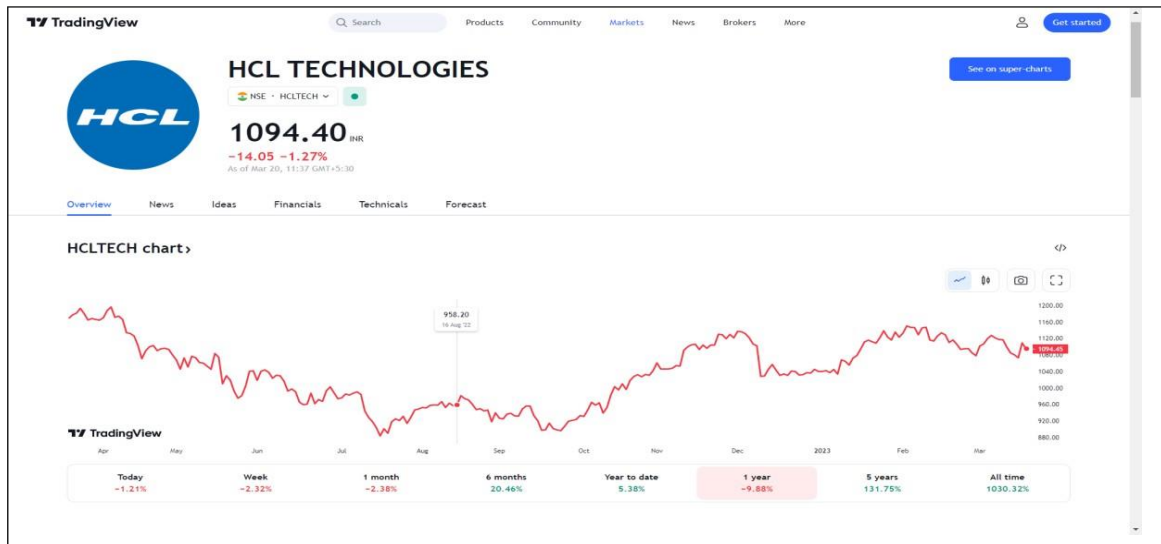
**Fig 5.15: Result of Bajaj Finance Ltd Stock Prediction**

The stock being presented is BAJAJ, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



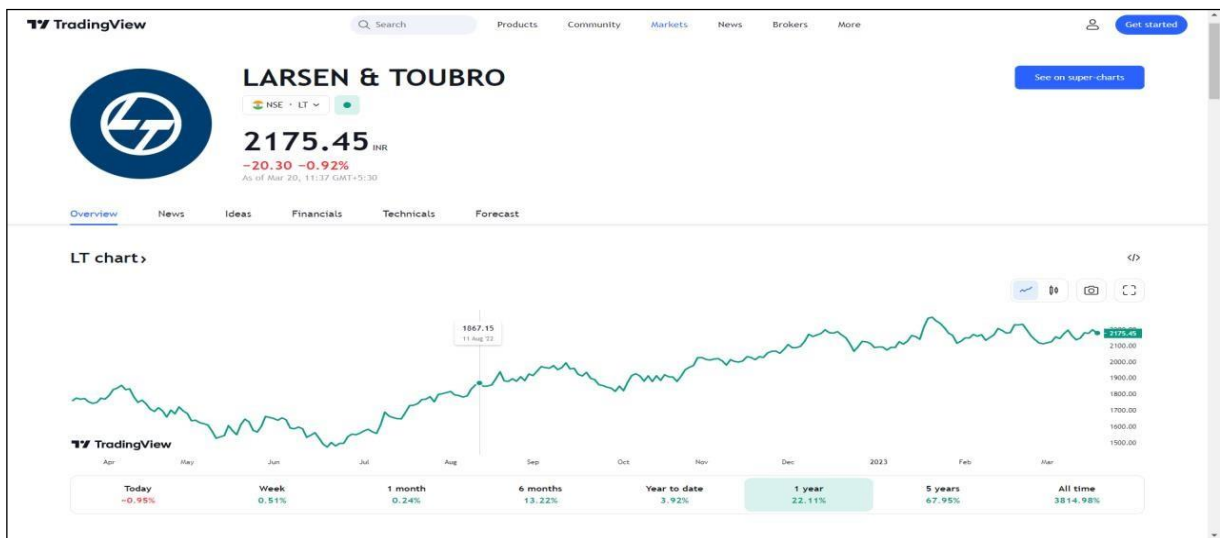
**Fig 5.16: Result of Kotak Mahindra Ban Stock Prediction**

The stock being presented is KOTAK, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



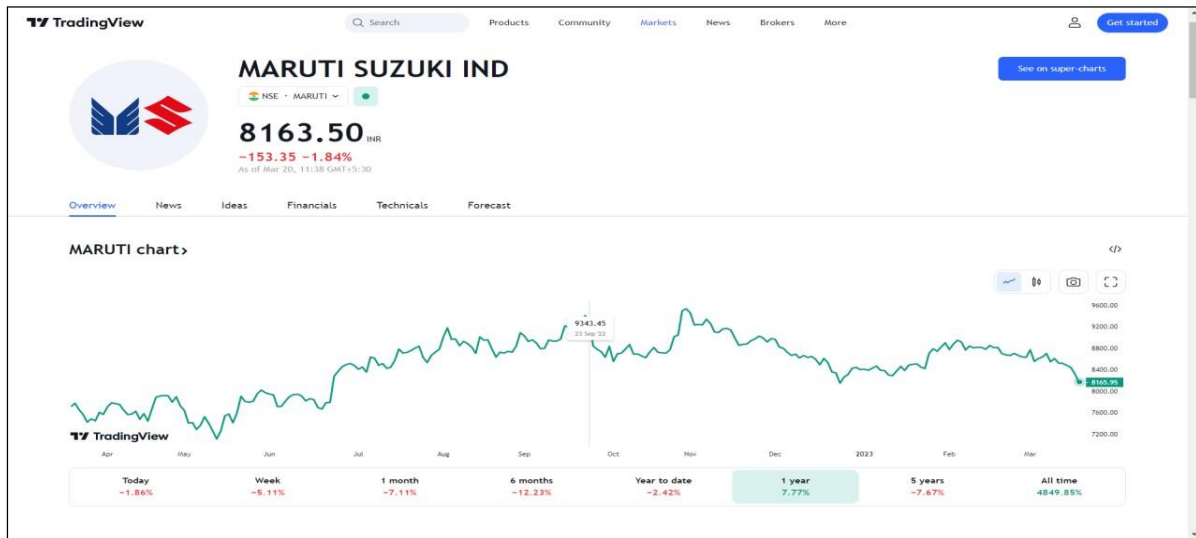
**Fig 5.17: Result of HCL Technologies Stock Prediction**

The stock being presented is HCL, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



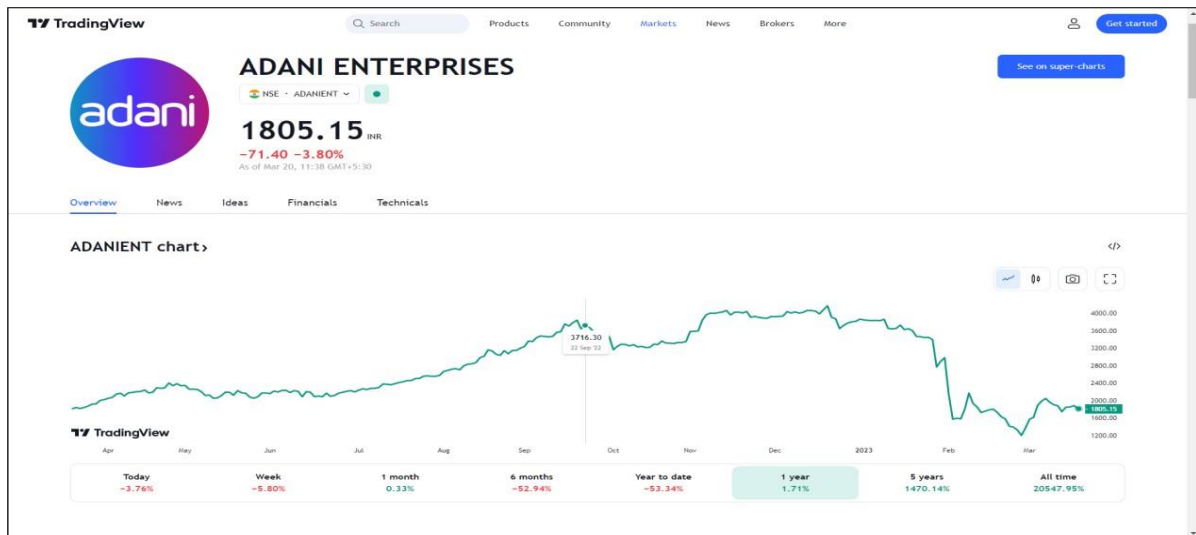
**Fig 5.18: Result of Larsen & Turbo Stock Prediction**

The stock being presented is L&T, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



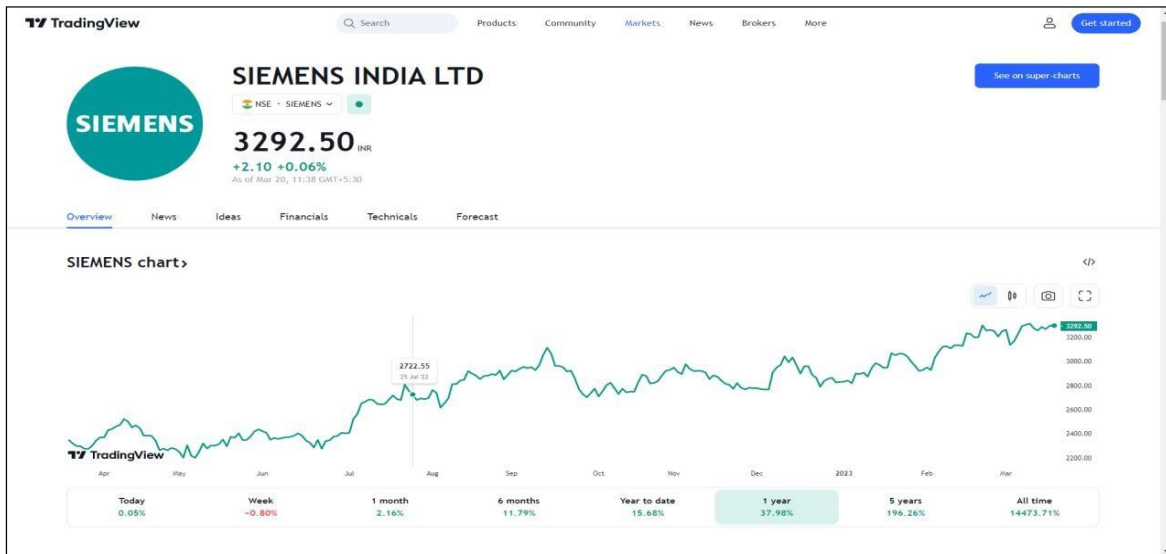
**Fig 5.19: Result of Maruti Suzuki Ind Stock Prediction**

The stock being presented is MARUTI SUZUKI, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



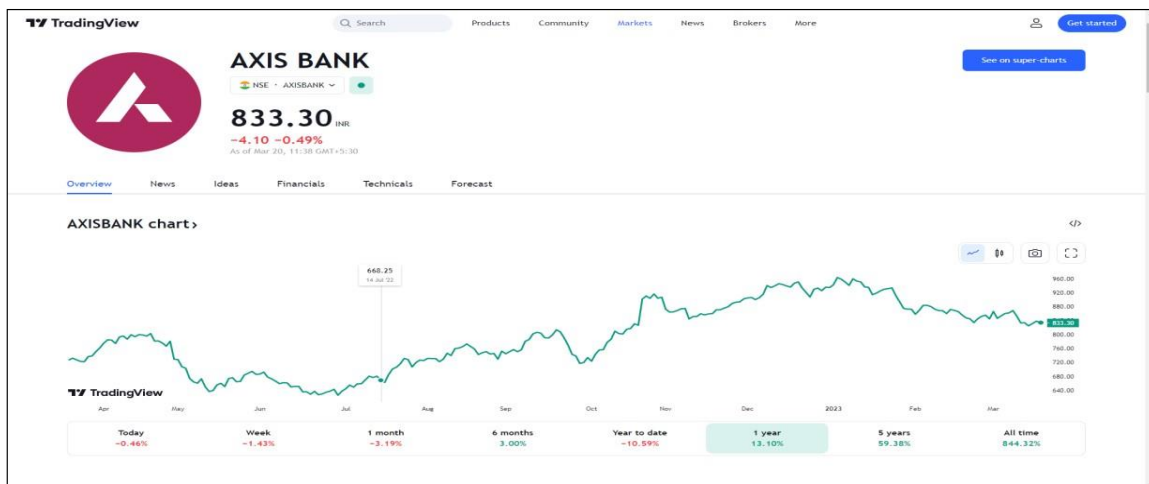
**Fig 5.20: Result of Adani Enterprises Stock Prediction**

The stock being presented is ADANI, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



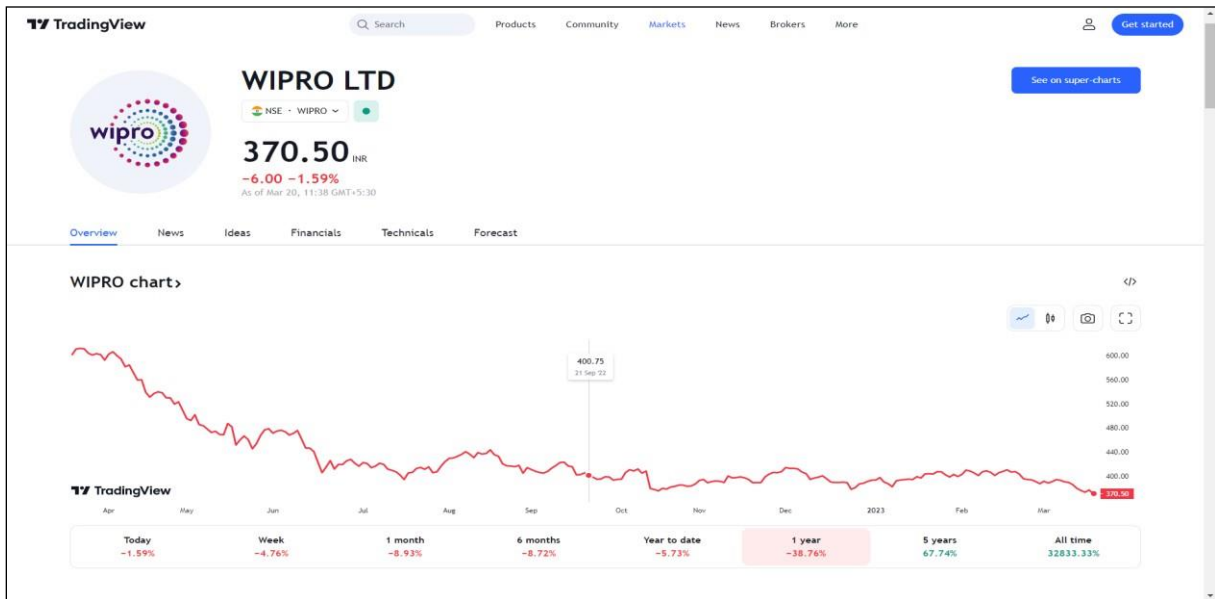
**Fig 5.21: Result of Siemens India Ltd Stock Prediction**

The stock being presented is SIEMENS, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



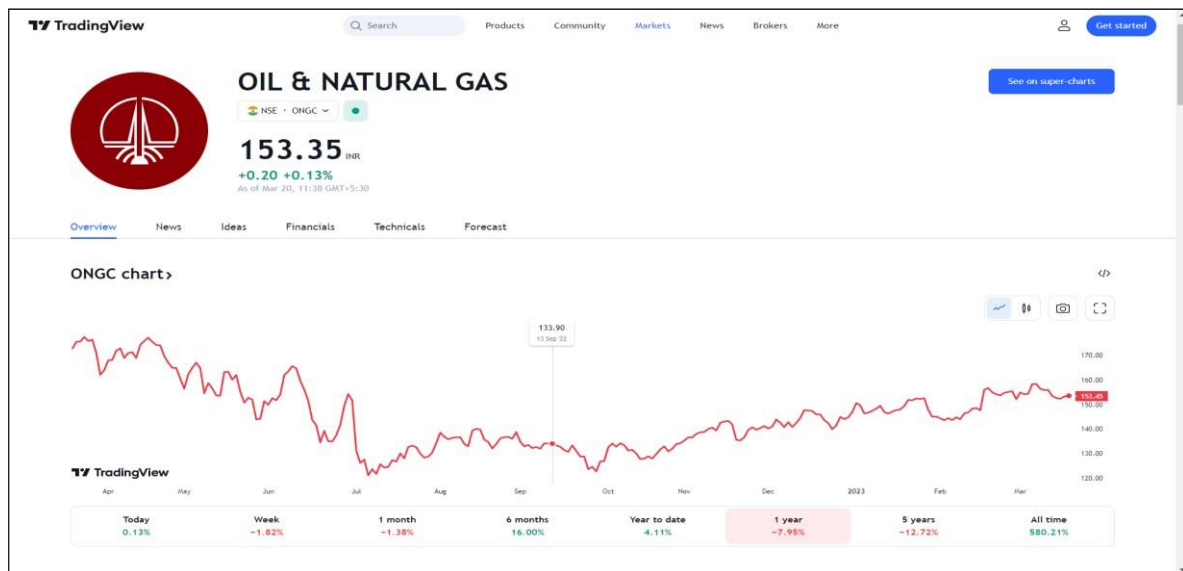
**Fig 5.22: Result of Axis Bank Stock Prediction**

The stock being presented is AXIS, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



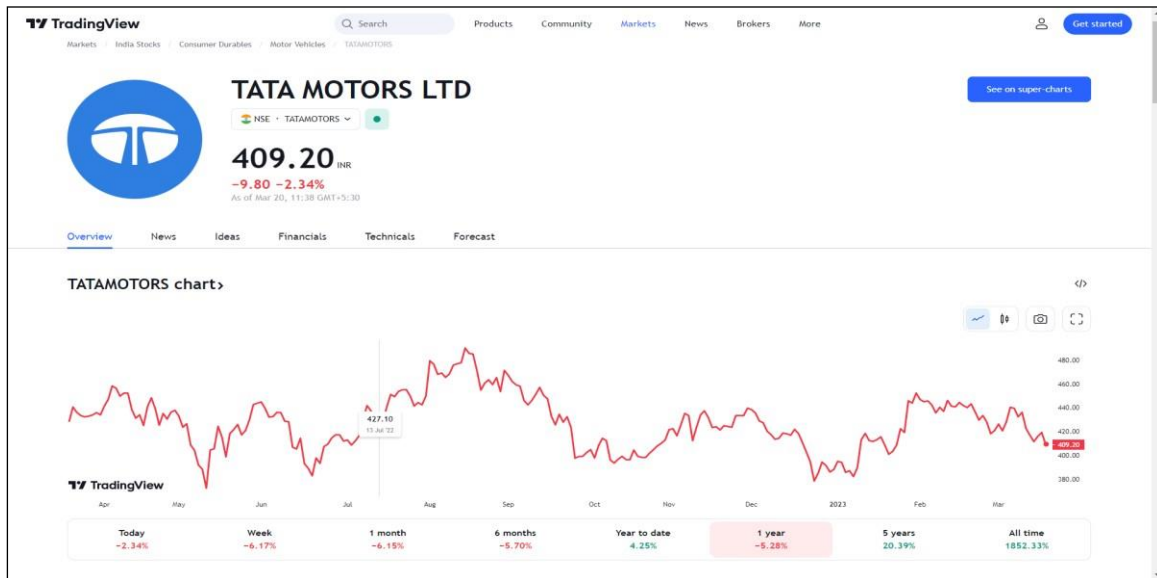
**Fig 5.23: Result of Wipro Ltd Stock Prediction**

The stock being presented is WIPRO, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



**Fig 5.24: Result of Oil & Natural Gas Stock Prediction**

The stock being presented is OIL AND NATURAL GAS, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.



**Fig 5.25: Result of Tata Motors Ltd Stock Prediction**

The stock being presented is TATA MOTORS, which is traded on BSE. Over the past 10, the stock has shown positive/negative performance, with increase/decrease in value by percentage.

## **CHAPTER 6**

### **SUMMARY AND CONCLUSIONS**

#### **6.1 SUMMARY**

In this project we have implemented two different models for predicting stock market price of yahoo stock for next day. SVM and ANN is compared, and it is found that ANN is good choice over SVM in terms of accuracy and SVM is faster than ANN. As stock market fluctuates with numerous factors exact prediction of trend is next to impossible. The project called "Stock Market Price Prediction using "Data Science Techniques" is focused on developing predictive models that can forecast future stock prices. The project employs a diverse array of data science methodologies, including machine learning, statistical analysis, and data visualization, to conduct an in-depth analysis of historical stock market data and discern meaningful patterns and trends. The project involves collecting and pre-processing the data, selecting appropriate features, and training the predictive models using algorithms. The efficacy of the models is evaluated using metrics like mean squared error, R-squared, and accuracy. Furthermore, the project encompasses the development of a user interface that empowers users to input stock market data and receive forecasts for future stock prices. The overarching objective of the project is to provide insights into stock market trends and facilitate informed decision-making for investors in terms of stock trading.

#### **6.2 CONCLUSION**

In the project, we proposed utilizing data from diverse global financial markets in conjunction with data science algorithms to forecast stock index movements. Our findings can be summarized as follows: SVM algorithm proves effective in handling large datasets from different financial markets without overfitting issues. Correlation analysis uncovers significant connections between the stock index and global markets that close just before or at the start of trading hours. Several data science- based models are proposed for predicting daily market trends, and numerical results exhibit high efficiency. A practical trading model is constructed based on a well- trained

predictor, which generates higher profits compared to selected benchmarks. In summary, the project "Stock Market Price Prediction using Data Science Techniques" showcases the effective utilization of data science techniques for forecasting future stock prices. Through historical data analysis and pattern identification, predictive models were developed, providing valuable insights into stock market behaviour. The project encompassed data collection, pre-processing, feature selection, and training of predictive models using various algorithms. Model performance was evaluated using diverse metrics, indicating accurate predictions. Additionally, the significance of data visualization and user interface design in making the model results accessible was demonstrated. The user interface facilitated input of stock market data and retrieval of future price predictions. Overall, the project contributes to the field of finance, highlighting the potential of data science techniques in predicting stock market prices. However, it is important to exercise caution as stock market behaviour is complex and unpredictable, and predictions from these models should be carefully considered and not solely relied upon for investment decisions.

## REFERENCES

- [1] Akshit Malik; Priyanshi Gupta; Surbhi Vijh, "Towards a Stock Price Prediction on Time Series Data using Long-Short Term Memory Method", 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence).
- [2] Amit Kumar; Meeta Chaudhry, "Review and Analysis of Stock Market Data Prediction Using Data mining Techniques", 2021 5th International Conference on Information Systems and Computer Networks (ISCON).
- [3] E Naresh; Babu J Ananda; K S Keerthi; M R Tejonidhi, "Predicting the Stock Price Using Natural Language Processing and Random Forest Regressor" , 2022 IEEE International Conference on Data Science and Information System (ICDSIS).
- [4] F. Z. Xing, E. Cambria, and R. E. Welsch, "Intelligent asset allocation via market sentiment views," IEEE Computer Intelligence, vol. 13, no. 4, pp. 1–20, 2018.
- [5] F. Z. Xing, E. Cambria, and R. E. Welsch, "Natural language based financial forecasting: a survey," Artificial Intelligence Reverend, vol. 50, no. 1, pp. 49– 73, 2018.
- [6] Indronil Bhattacharjee; Pryonti Bhattacharja, "Stock Price Prediction: A Comparative Study between Traditional Statistical Approach and Machine Learning Approach" , 2019 4th International Conference on Electrical Information and Communication Technology (EICT).
- [7] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," J. Computer Science, vol. 2, no. 1, pp. 1–8, 2011.
- [8] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock and stock price index movement using Trend Deterministic Data Preparation and Data Science techniques," Expert System Application, vol. 42, no. 1, pp. 259–268, 2015.
- [9] Julee Kumari; Vivek Sharma; Sansar Chauhan, "Prediction of Stock Price using Machine Learning Techniques: A Survey" , 2021 3rd International

Conference on Advances in Computing, Communication Control and Networking (ICAC3N).

- [10] M.S. Roobini; K. Babu; Jophra Joseph; G Ieshwarya. , “Predicting Stock Price using Data science technique”, 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS).
- [11] P. Swarna Lakshmi; N. Deepika; V. Lavanya; L.Jenitha Mary; D. Raj Thilak; A. Adlin Sylvia, “Prediction of Stock Price Using Machine Learning”, 2022 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI).
- [12] S. Agrawal, M. Jindal, and G. N. Pillai, “Momentum analysis based stock market prediction using adaptive Neuro-Fuzzy Inference System(ANFIS),” in The international multiconference of engineers and computer scientists, 2010, vol. I, pp. 526–531.
- [13] S. Barik, S. Das, and S. K. Sahoo, “A hybrid forecasting model for stock value prediction using soft computing skill,” Int. J. Computer. Science Eng., vol. 5, no. 4, pp. 40–45, 2017.
- [14] S. D. Patel, D. Quadros, V. Patil, M. Pawale, and Harsha Saxena, “Stock prediction using neural networks,” Int. J. Engineering Management Research, vol. 7, no. 2, pp. 490–493, 2017.
- [15] S. Dutta and R. Rohit, “Stock market prediction using data mining techniques with R,” Int. J. Engineering Science Computer, vol. 7, no. 3, pp. 5436–5441, 2017.
- [16] S. Nithya Tanvi Nishitha; Shahana Bano; G. Greeshmanth Reddy; Pujitha Arja; Gorsa Lakshmi Niharika, “Stock Price Prognosticator using Machine Learning Techniques”, 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA).
- [17] S. Sarvesh; R. V. Sidharth; V. Vaishnav; J Thangakumar; S Sathyalakshmi, “A Hybrid Model for Stock Price Prediction using Machine Learning Techniques with CNN”, 2021 5th International Conference on Information Systems and Computer Networks (ISCON).

- [18] Shahnawaz Khan; Mustafa Raza Rabbani; Abu Bashar; Mustafa Kamal, “Stock Price Forecasting Using Deep Learning Model” , 2021 International Conference on Decision Aid Sciences and Application (DASA).
- [19] Shailender Kumar; Rounak Gupta; Puneet Kumar; Naman Aggarwal, “A Survey on Artificial Neural Network based Stock Price Prediction Using Various Methods”, 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS).
- [20] Y. S. Bu-Mostafa and A. F. Atiya, “Introduction to financial forecasting,” Application Intelligence vol. 6, no. 3, pp. 205–213, 1996.
- [21] Z. Wang, A. Tan, F. Li, and S.-B. Ho, “Comparisons of learning based methods for stock market prediction,” in The 4th International Conference on Cloud Computing and Security (ICCCS 2018), 2018.