STOCK MARKET PREDICTION USING MACHINE LEARNING FOR ACCURATE ANALYSIS OF SHORT-TERM AND LONG-TERM TRADING

Project Submitted to the
SRM University AP, Andhra Pradesh
for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology
in
Computer Science & Engineering
School of Engineering & Sciences

submitted by

Aditya Dubey (AP21110010729)

Adnan Khan (AP21110011217)

R. Jai Kaushik (AP21110010774)
Simhadri Venkata Mohit (AP21110010768)

Under the Guidance of

Dr. Veeravel V.

Department of Computer Science & Engineering
SRM University-AP
Neerukonda, Mangalgiri, Guntur
Andhra Pradesh - 522 240
May 2025

DECLARATION

I undersigned hereby declare that the project report **Stock Market Prediction using Machine Learning for Accurate Analysis of Short-term and Long-term Trading** submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology in the Computer Science & Engineering, SRM University-AP, is a bonafide work done by me under supervision of Dr. Veeravel V.. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree of any other University.

Place	:	Date	: April 26, 2025
Name of student	: Aditya Dubey	Signature	: .....
Name of student	: Adnan Khan	Signature	: .....
Name of student	: R. Jai Kaushik	Signature	: .....
Name of student	: Simhadri Venkata Mohit	Signature	: .....

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRM University-AP

Neerukonda, Mangalgiri, Guntur

Andhra Pradesh - 522 240

CERTIFICATE

This is to certify that the report entitled **Stock Market Prediction using Machine Learning for Accurate Analysis of Short-term and Long-term Trading** submitted by **Aditya Dubey , Adnan Khan , R. Jai Kaushik , Simhadri Venkata Mohit** to the SRM University-AP in partial fulfillment of the requirements for the award of the Degree of Master of Technology in in is a bonafide record of the project work carried out under my/our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Project Guide	Head of the Department
Name : Dr. Veeravel V.	Name : Dr. Murali Krishna Enduri
Signature:	Signature: .....

ACKNOWLEDGMENT

I wish to record my indebtedness and thankfulness to all who helped me prepare this Project Report titled **Stock Market Prediction using Machine Learning for Accurate Analysis of Short-term and Long-term Trading** and present it satisfactorily.

I am especially thankful for my guide and supervisor Dr. Veeravel V. in the Department of Computer Science & Engineering for giving me valuable suggestions and critical inputs in the preparation of this report. I am also thankful to Dr. Murali Krishna Enduri, Head of Department of Computer Science & Engineering for encouragement.

My friends in my class have always been helpful and I am grateful to them for patiently listening to my presentations on my work related to the Project.

Aditya Dubey , Adnan Khan , R. Jai Kaushik , Simhadri Venkata Mohit
(Reg. No. AP21110010729, AP21110011217, AP21110010774,
AP21110010768)
B. Tech.
Department of Computer Science & Engineering
SRM University-AP

ABSTRACT

Stock price prediction is an unsettled exercise with regard to volatile economic variables, geopolitics, and investor sentiment has significantly influenced the stock prices. This research explores how machine learning frameworks helps for short-term (intraday-weekly) and long-term (monthly-yearly) stock price forecasting. Present study considers closing prices key input, augmented by macroeconomic variables (e.g., GDP, interest rates, inflation, exchange rates, etc), to estimate both short-term price variations and general economic trends.

For short-term forecasting, LSTM networks are applied to discover temporal patterns between price series. Long-term forecasting employs ensemble methods like Random Forest and XGBoost, and Linear Regression, to forecast macroeconomic trends and stability. We select ten companies listed in the Bombay Stock Exchange based on their market capitalisation to train models and RMSE, MAE, and R Square to test and measure accuracy and risk-adjusted returns. Results indicate that LSTMs are approximately 82 percent accurate in predicting short-term volatility, and ensemble methods and linear regression are approximately 75 percent accurate in predicting long-term trends. Overfitting and noise in the market are addressed by applying regularization, feature scaling, and walk-forward validation.

This project demonstrates the robustness of closing-price-based models in achieving a balance between simplicity and predictive power, delivering practical tools for traders' strategic decision-making. Further research can be expanded to multi-asset portfolios or real-time prediction systems.

CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

The stock market, which is a complex and dynamic arena, is influenced by numerous variables from macroeconomic signals and business performance, investor sentiment and geopolitics. Forecasting stock prices accurately is an intimidating task as a result of the inherent volatility and non-linear dynamics in financial time-series data. Traditional statistical techniques typically lose out on such complexities and thus require sophisticated computational methods. The challenge is overcome in this project by extending machine learning (ML) and deep learning (DL) techniques for predicting stock prices for day-by-day trading as well as long-term investment policy based on historic records of 10 high-market-capitalization firms retrieved from Investing.com.

The primary objective is to develop a robust, data-driven model that integrates time-series analysis, technical indicators, and ensemble learning with the objective of predicting future closing prices. The project is implemented in a two-step manner:

Deep Learning: A stacked Long Short-Term Memory (LSTM) network is utilized to learn temporal relationships in sequence of price data.

Traditional Machine Learning: Linear Regression, Decision Tree, Random Forest, and XGBoost models are utilized for sequence-based forecasting for comparison with LSTM.

In the context of the datasets, the 10 companies' stock prices have been utilized for this research work. Volatile stock firms have been selected

to train the models more efficiently and accurately with the ability to predict stock prices. The companies shortlisted are: AXIS Bank, Reliance Industries, ICICI Bank, HDFC Bank, Hindustan Unilever, TCS, Bharti Airtel, Bajaj Finance, Infosys, and SBI ETF.

The stock information covers a period of 10 years — from January 2015 to April 2025, and was obtained from Investing.com and the Bombay Stock Exchange (BSE). We selected this companies top 10 compines according to their Market Capital

1.1 ABOUT STOCK MARKET

The stock market is a marketplace where buyers and sellers trade shares of publicly listed companies. It's a core component of the global economy, enabling businesses to raise capital and investors to own a stake in companies, with the potential to profit from their growth.

1.2 STOCK EXCHANGE

The stock exchange is a forum where shares in quoted companies can be bought and sold. It is a secondary type of market. To go public and sell its shares to investors out in the market, a company must find a niche for itself on one of the established stock exchanges, and then a promoter must sell a large amount of its shares to the public retail investors, which once successfully done, then further trading can be carried on in the secondary market or stock exchange. In India, there are prevalently two major stock exchanges namely, the Bombay stock exchange (BSE) comprising approximately 5000 listed entities, and National stock exchange (NSE) with about 1600 listed companies. NSE and BSE both have similar functionality

and trading mechanisms. Demat and Trading accounts are used mainly for stock market trading. Stock exchanges help the general public save their money and channel their funds while the companies have an oversupply of investments into their ventures. The stock market has brought about a revolution in the arena of Indian investments. Faced with increasing inflation, declining bank interest rates, and a demand for greater returns the middle-class investors are now shifting towards the equity market. This in totality sums up the ever-growing need and importance of stock trades

1.3 OHLC GRAPHS

There are several ways and methods of interpreting or analyzing OHLC charts. the markings and denotations to understand OHLC values.

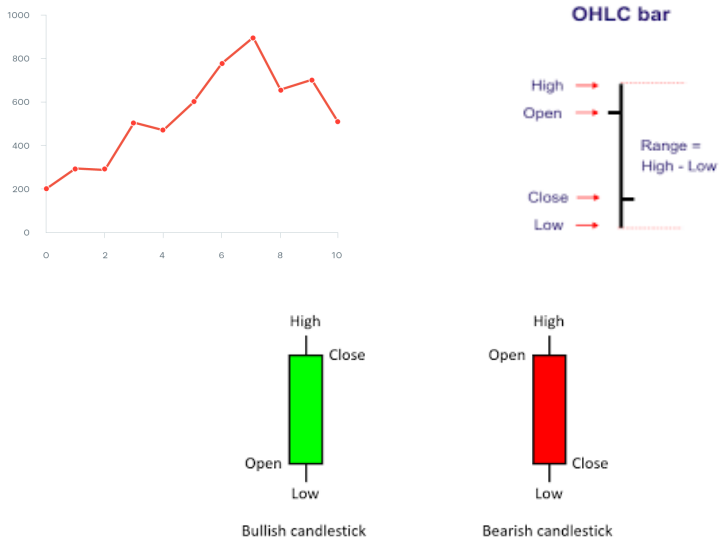


Figure 1.1: OHLC Graphs.

This project demonstrates the revolutionary potential of machine learning in finance, equipping traders with capabilities to handle volatility and optimize returns. It demonstrates how LSTM networks are best

at handling short-term price patterns and **ensemble methods** like Random Forest and XGBoost provide robust long-term analysis, highlighting the strength of hybrid modeling for trading horizons of different durations. Issues like overfitting, high costs, and sensitivity to market changes highlight the importance of strong data curation and regularization. Tools like Streamlit make predictive analytics accessible to non-technical users, enabling data-driven decision-making. Ethically, it challenges dependence on algorithmic forecasts in volatile markets, highlighting the importance of visibility into model limitations and risk. In finance, these results promote ongoing innovation in adaptive models that engage real-time data and external factors, enhancing resilience. Overall, the project consolidates ML innovations with actionable trading strategies, enabling smarter, risk-conscious investment strategies.

Chapter 2

MOTIVATION

The stock market is a central force in world economies, dictating investment choices, business expansion, and personal financial planning. Nonetheless, its natural volatility, fueled by geopolitical crises, economic data, and market sentiment, renders precise forecasting an uphill battle. Historical money flows and linear models tend to assume linearities or historical patterns that fail to replicate the intricate, nonlinear interdependencies and dynamic temporal dynamics that exist in market data. This gap creates an opportunity to leverage advanced machine learning techniques to improve forecasting accuracy and provide actionable insights.

Our project is motivated by several key factors:

Financial Market Complexity: Stock prices are shaped by complex interactions between quantitative information (e.g., price history, volumes) and qualitative aspects (e.g., news sentiment, macroeconomic policies). Machine learning models, especially LSTM networks, are good at modeling sequential data and temporal relationships and thus are well-suited to model these dynamics.

Machine Learning Advances: New algorithms such as XGBoost and Random Forest are capable of dealing with high-dimensional data, detecting nonlinear relationships, and reducing overfitting, providing a strong alternative to traditional statistical techniques. By comparing various models (LSTM, Random Forest, Linear Regression, and XGBoost), we seek to determine the best method for various market conditions.

Practical Application: Precise forecasting enables investors to make well-informed decisions, maximize portfolios, and hedge risk. Democratizing access to similar tools might create a leveler for retail investors, who generally don't have the resources available to institutional traders.

Academic and Technical Curiosity: This work delves into the convergence of finance and AI, raising such questions as: Can ensemble algorithms beat deep learning under some market scenarios? What about hybrid models? Our conclusions might inform both academic studies and practical trading applications.

Data and Computational Readiness: The existence of enormous historical market data, combined with improved computational capabilities, provides the perfect opportunity to develop innovative algorithmic trading systems.

By addressing this challenge, we hope to better understand the contribution of machine learning to financial forecasting while providing a tool with concrete value to investors.

Chapter 3

LITERATURE SURVEY

We used the following published research papers on stock market prediction for reference:

1. Title: "Stock Market Prediction with High Accuracy using Machine" Authors: Malti Bansal, Apoorva Goyal, Apoorva Choudhary University: Delhi Technological University (DTU)

This research was conducted on datasets from companies like TCS, Tata Steel, Maruti, Axis Bank, Adani Ports, NTPC, HDFC. The aim was to improve the accuracy of stock market predictions using various machine learning algorithms, addressing the challenges posed by market volatility and noise.

Algorithms Applied: Linear Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), LSTM (Long Short-Term Memory) neural network

Evaluation Metrics: The models were evaluated using performance indicators such as: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2 Score

Results: The LSTM model outperformed others in terms of prediction accuracy, especially for capturing temporal patterns in time-series stock data. Traditional models like Linear Regression showed limitations in handling non-linearity and time dependencies.

2. Title: "Stock Market Prediction Using Machine Learning Techniques" Authors: Naadun Sirimevan, I.G. U. H. Mamalgaha, Chandira

Jayasekara, Y. S. Maruyan, Chandimal Jayawardena, Faculty of Computing, Sri Lanka Institute of Information Technology

Stock data was collected from Yahoo Finance using the yfinance library.

Models Used: Linear Regression, Support Vector Regression (SVR), K-Nearest Neighbors (KNN), Decision Tree Regressor, Random Forest Regressor, Long Short-Term Memory (LSTM) – a type of RNN suited for time-series data.

Metrics used: Mean Squared Error (MSE) and Root Mean Squared Error (RMSE).

Conclusion: LSTM outperformed traditional ML models, highlighting the importance of sequence learning in stock market prediction. The study recommends using LSTM or other deep learning approaches for better accuracy in future predictions.

3. Title: "Survey of Stock Market Prediction Using Machine Learning Approach" Authors: Ashish Sharma, Dinesh Bhuriya, Upendra Singh University: Govt. Women's Polytechnic, Indore

This paper surveys machine learning approaches, particularly regression analysis, for stock market prediction. The authors highlight the inherent complexity and nonlinearity of stock markets, emphasizing the limitations of traditional methods like fundamental and technical analysis. They argue that data mining and regression techniques offer superior accuracy by uncovering hidden patterns in historical data.

Final Takeaway: While regression methods provide a foundational toolset, the evolving complexity of financial markets demands innovative, hybrid machine learning solutions to empower investors with reliable predictions.

4. Title: "Stock Market Prediction Using LSTM Recurrent Neural Network" Authors: Adil Moghara, Mhamed Hamiche University: University Abdelmalek Essaadi, Morocco

This paper explores the use of Long Short-Term Memory (LSTM) Recurrent Neural Networks to predict stock market values, focusing on the impact of training epochs and data characteristics on model performance.

Datasets: GOOGL (Google): Daily opening prices from August 19, 2004, to December 19, 2019. NKE (Nike): Daily opening prices from January 4, 2010, to December 19, 2019. Data was sourced from Yahoo Finance.

The LSTM model demonstrated promising results in tracking stock price trends, particularly with more training epochs, which significantly reduced prediction errors. However, the model's performance degraded when faced with abrupt changes in market volatility or data distribution (e.g., shifts from stable to highly volatile periods). The authors conclude that dataset selection (avoiding inconsistent volatility) and epoch optimization are critical for enhancing accuracy.

Chapter 4

MODELS APPLIED AND METHODOLOGY

4.1 MODEL SELECTION

4.1.1 Linear Regression Algorithm

In ML, the linear regression algorithm is classified as supervised learning. Rather than predicting categories, this algorithm forecasts values well within the range. It creates a linear relationship between the dependent and independent variable and does not perform very well with the non-linear type of data sets because of outlier presence. Researchers who predicted stock markets using this algorithm found significant difficulties that had to be addressed if it were to be used for daily stock value prediction. Investors cannot consistently put money based on this algorithm's forecast. The project adopts a structured and modular solution to forecast stock prices based on both deep learning and conventional machine learning methods. The entire structure guarantees thorough handling of data, robust model training, and accurate performance testing.

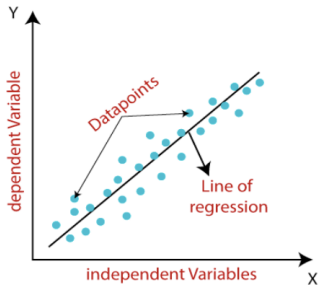


Figure 4.1: Linear regresssion.

4.1.2 Decision Tree Algorithm

A supervised machine learning technique for classification and regression problems is the Decision Tree algorithm. By dividing the dataset into subsets according to the value of the input features, it creates a structure resembling a tree, with each internal node standing for a feature-based decision, each branch for an outcome of that decision, and each leaf node for a final output or class label. In order to produce pure or nearly pure subsets, the algorithm employs metrics such as Mean Squared Error, Entropy, and Gini Impurity to identify the optimal feature to split the data at each stage. Decision trees are widely used to understand decision-making processes because they are simple to understand and visualize. But they are susceptible to overfitting, particularly when dealing with intricate datasets.

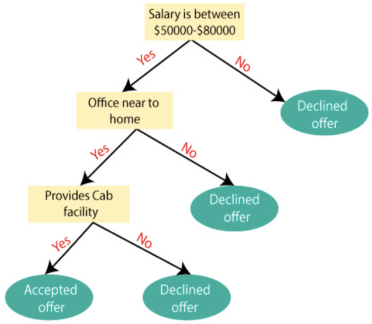


Figure 4.2: Decision tree.

4.1.3 Random Forest Algorithm

An ensemble learning method for classification and regression applications is the Random Forest algorithm. In order to produce more precise and reliable predictions, it constructs several decision trees during training and aggregates their results. Diversity is introduced into the forest by training each tree on a random subset of the data and splitting nodes using a

random subset of features. The algorithm averages each tree's predictions for regression and uses a majority vote from all the trees for classification. In comparison to a single decision tree, this randomness lessens overfitting and enhances generalization to unknown data. Random Forest is well-known for its high accuracy, robustness against noise and missing values, and capacity to handle large, high-dimensional datasets.

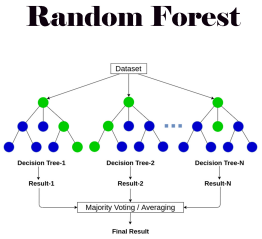


Figure 4.3: Random Forest.

4.1.4 XG BOOST Algorithm

Based on the gradient boosting framework, XGBoost (Extreme Gradient Boosting) is a potent and effective machine learning algorithm. In a sequential fashion, it constructs a collection of decision trees, each of which is trained to fix the mistakes of the ones before it. By employing gradient descent to optimize a loss function, XGBoost refines the model step by step, in contrast to Random Forest, which constructs trees independently. It supports parallel processing, which makes it faster and more scalable, and introduces sophisticated regularization techniques (L1 and L2) to lessen overfitting. XGBoost can handle both classification and regression problems and automatically handles missing values. XGBoost is now one of the most widely used algorithms in data science competitions and practical applications because of its exceptional performance, speed, and accuracy.

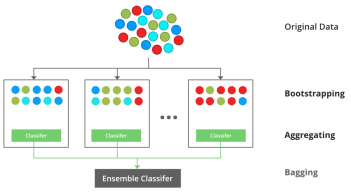


Figure 4.4: XG-Boosting.

4.1.5 Long Short-Term Memory Algorithm

A specific kind of Recurrent Neural Network (RNN) called LSTM (Long Short-Term Memory) is made to efficiently learn from sequential and time-dependent data. In contrast to conventional RNNs, which have issues with vanishing or exploding gradients, LSTMs regulate the information flow using memory cells and three essential gates: input, forget, and output gates. These gates enable the network to retain crucial information over lengthy sequences by assisting it in determining which data to output, forget, or remember. Because of this, LSTM is perfect for applications like speech recognition, natural language processing, and time series forecasting. It is effective at modeling patterns that emerge over time because it can identify long-term dependencies in data. Real-world applications such as text generation, translation systems, and stock prediction frequently use LSTM networks.

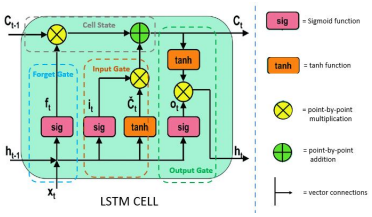


Figure 4.5: LSTM.

4.2 METHODOLOGY

The research design for any research study is the most important parameter to achieving original and genuine results. Thus, research work methodology used on this work has been specially chosen and scientifically organized so as to render reliable and wholesome conclusions from the implementation. The methodology used in this research project is explained to a large degree by the following steps. Step one is the collection of raw data from open sources. Preprocessing of data is the second step and comprises data scaling, data standardization, data cleaning, and other technical methods of building the dataset. The third step is to split the dataset into test data. The subsequent step sets the stage for training the five models that are constructed based on five respective algorithms by feeding, which is carried out with the training dataset, after which the models have been tested using the testing dataset in order to be able to notice the deviation from actual values in different models. Finally, the five various models for every one of the twelve companies' data were compared and tested using effective performance measures in particular, Mean Absolute Error (MAE), R2-Value (R-squared), and Root Mean Square Error (RMSE), in order to rate and grade the performance, draw some comparisons and reliable conclusions in respect of the algorithms, i.e., Linear Regression algorithm, Decision Tree Algorithm, Random Forest, XG-Boosting, Long Short-Term Memory Algorithm. The steps taken to execute this research implementation are shown in the flow diagram below written in a generalized form, while the details have been mentioned in the subsequent sub-sections.

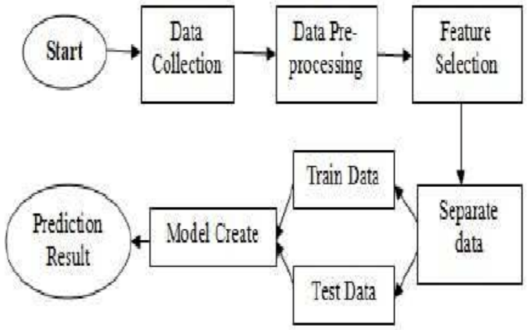


Figure 4.6: Methodology followed.

4.2.1 Data Description

The most initial and most important step in most predictive analytics ML projects is choosing or acquiring a sufficient data set that involves. In this implementation research work, the stock price data were accessed using the Investing Website, Bombay Stock Exchange, from the date January 2015 up to and including April 2025 for a term of ten well-known entities or firms that account for a massive portion of market share and economy, particularly in the Indian market. The above are the characteristics of the existing stock price datasets, that is, historical closing price, opening price, all-time high, all-time low, last price, and closing price of the stocks. Although the ten companies whose data sets have been utilized are as follows: AXIS Bank, Reliance Industries, ICICI Bank, HDFC Bank, Hindustan Unilever, TCS, Bharti Airtel, Bajaj Finance, Infosys and SBI ETF.

4.2.2 Data Pre-processing

Data preprocessing especially for data intensive projects is a very critical process because it involves processing of raw and random data in a manner that improves its quality by eliminating or purifying the unwanted

points as well as normalizing it for regular use and making it able to provide informative facts. It is not the large amount of information that produces very good results but data quality that counts. It consists of data cleaning, segregation or organization of data, scaling of data, standardization of data, etc., i.e., normalization of data standardization and encoding of categorical data. At the data pre-processing phase of the project, Min-Max scalers were applied in order to scale the data to scale and standardize and the null values, missing values, and unknown values were cleaned and discrepancies if any were resolved. The two main Python libraries that were which were used for the preprocessing of the dataset and for data visualization were NumPy and Pandas, and Matplotlib was employed. NumPy served as a science calculator whenever needed while working on manipulations the data sets, yet Pandas library was appropriate for data analysis and data manipulation. Matplotlib was used to plot the data in the form of charts.

Figure 4.7: Data pre-processing

4.2.3 Feature Selection

Feature selection is the process of selecting the most important input variables (features) from a data set to maximize the performance of a machine learning model. By selecting only the significant features, it prevents overfitting, enhances training time, and can improve model accuracy. Feature selection is useful especially when working with high-dimensional

data sets, in the sense that it allows the model to concentrate on meaningful patterns and discard noisy or irrelevant data.In this we have chosen Closing price as feature

4.2.4 Splitting of data into train and test dataset

Splitting data into training and test sets is a fundamental step in building a machine learning model. The **training set** is used to train the model by allowing it to learn the patterns and relationships within the data. The testing set, however, is used to test the performance of the trained model on new, unseen data. This split allows one to test the model's ability to generalize to new, real-world inputs. The data is typically split in a ratio such as 80:20 with the larger set being used for training. In Python, this is typically done using 'trains plit' and tests split from Scikit-learn. Careful data splitting is required to avoid overfitting and obtain accurate model evaluation using performance measures such as accuracy, RMSE, MAE, MSE.

4.2.5 Training of models

Training is the mechanism through which ML algorithms learn valuable information to train the models in a manner that they are capable of predicting with high precision and therefore desired outcomes. The algorithms chosen for this project were Linear Regression algorithm, Decision Tree Regression algorithm,Random Forest Algorithm, XG-Boosting Algorithm and Long Short-Term Memory algorithm. These five models for each of the ten companies' were trained on the training data with the consideration of overfitting or underfitting issues. Regular learning from the models was adopted as the approach to enhance the predictive capability of the

models. The algorithms used in this project are that of supervised learning category and the category of learning that has been embraced for this project implementation is also of the supervised type. The target attribute i.e., the desired value should be a portion of the data as it is a very important basis for prediction. The highly important features like the time taken to train every model and the time lag error (i.e., the number of steps to move data reverse in time) were also come into effect. This process is achieved through an iterative learning mechanism called 'model fitting'. The weights of the model were initialized randomly as it offers more adjustability to algorithms.

4.2.6 Testing the models

This is the last phase of the process, one which as its name implies examines the performance of chosen fully trained models on the basis of some effective and productive performance criteria or indicators. Each of the five models i.e. Linear Regression algorithm, Decision Tree Regression algorithm,Random Forest Algorithm, XG-Boosting Algorithm and Long Short-Term Memory algorithm. Memory were tested with the 8-value testing dataset which contained all combinations to ensure testing was as close to real-time whenever possible. The performance metrics or measures taken into account were Mean Absolute Error (MAE), Root Mean Square Error or Deviation (RMSE/RMSD) and R Squared value. (R2). SMAPE is also a typical and effective test parameter using percentage error or relative error. The the smaller the MAE value, the better the predictive power of a specific model. RMSE is the square root of the mean of the squared differences between actual and forecasted values. The smaller the value of RMSE, the better model. R Squared (R2), has the best value of '1', and the values are

either negative or positive, the closer the value to the positive '1', the better the model. Subsequently, detailed analytical and comparative results were drawn after the testing which are shown in the results section. The formulae used in the calculation of MAE, RMSE and R2 are presented as follows respectively:

$$\mathbf{MAE} = \frac{1}{\mathbf{n}} \sum_{i=1}^{\mathbf{n}} \left| y_i - \hat{y}_i \right|$$

$$\mathbf{MSE} = \frac{1}{\mathbf{n}} \sum_{i=1}^{\mathbf{n}} \left(y_i - \hat{y}_i \right)^2$$

$$\mathbf{R^2} = 1 - \frac{\mathbf{SS_{res}}}{\mathbf{SS_{tot}}}$$

Chapter 5

IMPLEMENTATION

Installation of the stock price prediction system begins with data preprocessing and collection. Historical stock data for 10 high-market-capitalization companies are downloaded from Investing.com in the CSV format with daily trading values such as 'Date', 'Open', 'High', 'Low', 'Price', 'Vol.', and 'Change %'. The column 'Date' is parsed into a datetime object and converted into the DataFrame's index to offer ease of handling time series. Numerical columns such as 'Price', 'Vol.', and 'Change %' also undergo complete cleaning: commas and percentage signs are removed using regex, and 'Vol.' is normalized by decoding abbreviations such as "K" (kilo, short for thousands) and "M" (millions) to numeric values ("1.2K" → 1200, "5.3M" → 5,300,000). The column 'Price' is renamed to 'Close' to ensure easy understanding and all features are cast to 'float32' to conserve memory.

Exploratory Data Analysis (EDA) comes after preprocessing to expose trends and relationships. A candlestick chart is produced with 'mplfinance', graphing 'Open', 'High', 'Low', and 'Close' prices over time, with volume bars displayed below to align trading activity with price movement. A line graph is then produced to follow historical closing price trends, and a correlation heatmap is produced to show relationships between features like 'Close', 'Vol.' and 'Change %' with 'seaborn'. These visualizations show periods of volatility, trading volume trends, and interdependencies between variables, such as the high correlation between 'Open' and 'Close' prices.

In model building, the closing prices are used and normalized to

the interval [0, 1] using 'MinMaxScaler' so that the model is stable during training. A sequential model is used, with a 5-day lag window to generate input-output pairs: input sequence is five consecutive days of normalized closing prices, and output is the sixth day's price. This transforms the data into a suitable shape for time-series forecasting. The data are split into training (80%) and test (20%) sets, keeping temporal order to avoid look-ahead bias.

The LSTM model is created with Keras, and the stacked architecture includes three LSTM layers (50 units) with 20% dropout regularization in between layers to avoid overfitting. The model is configured using Mean Squared Error (MSE) loss function. Training is done for 100 epochs at a batch size of 32, and early stopping stops training when validation loss does not improve for five consecutive epochs. Classical machine learning models—Linear Regression, Decision Tree, Random Forest, and XGBoost—are trained on identical data, but reshaped as 2D arrays (samples $\times$ 5 features) to meet their input requirement.

Model comparison involves reversing predictions back to their initial price range using the scaler's inverse transform method. Different performance metrics are calculated, i.e., mean absolute error (MAE), root mean squared error (RMSE), and R2 score, for training and test sets. For visualization purposes, a multi-subplot bar chart is used to compare MAE, RMSE, and R^2 scores for all models, with LSTM predictions overlaid on true closing prices to qualitatively ascertain their correlation with market trends. A ranking table systematically ranks the models based on RMSE, with the LSTM's enhanced ability in capturing temporal dependencies compared to conventional methods.

For enhanced practical use, an extension in the form of a Streamlit web

application is developed, from which users can choose datasets interactively, adjust forecasting horizons, and show forecasts in real time. Scalability is made possible due to modularity in the code that allows simple adaptation to new datasets or inclusion of additional features like macroeconomic variables. Overfitting, non-stationarity, and sparsity of data are managed systematically using the addition of dropout layers, sequence windowing, and synthetic data generation methods. The deployment concludes with a detailed analysis of results, emphasizing the ability of the LSTM to learn sequential patterns and the interpretability of tree models. This end-to-end methodology bridges theoretical machine learning concepts with actionable financial knowledge, providing a robust foundation for data-driven trading strategies.

Chapter 6

HARDWARE/ SOFTWARE USED

6.1 HARDWARE

Hardware:

1. Processor: Intel i7 or a processor of similar strength (or more) for heavy computation and execution of complex models such as LSTM and XGBoost.
2. RAM: At least 8GB to efficiently handle large datasets and parallel operations. 3. Graphics Processing Unit (GPU): Nvidia GTX 1060 or higher for accelerating deep learning models, especially LSTM training.
4. Storage: SSD with at least 256GB of storage space for faster data and model training access.
5. Operating System: Windows 10, macOS, or Linux-based OS (Ubuntu for simplicity in installation with Python and ML libraries).

6.2 SOFTWARE

Software:

1.Programming Languages:

- Python: Most widely used to create machine learning models and analyze data.

2. Libraries/Frameworks:

- Pandas & NumPy: To clean, preprocess, and manipulate data.

- Matplotlib & Seaborn: For visualizing data, graph plotting such as stock price vs. time, comparison of models, etc.

- Scikit-learn: For common machine learning algorithms like Linear Regression, Random Forest, Decision Trees, and evaluation metrics.

- Keras/TensorFlow/PyTorch: To train and develop deep learning models like LSTM.

- XGBoost: For gradient boosting models, which give high accuracy and performance in predictive tasks.

3. Data Collection:

- Investing.com: For retrieving historical stock price data.

4. Integrated Development Environment (IDE):

- Google Colab:For interactive model testing and development.

- Jupyter Notebook: For interactive model testing and development.

- VSCode/Spyder: Used to debug and code larger Python programs.

5. Version Control:

- Git/GitHub: For version control and collaborative use.

6. Deployment:

- Streamlit/Flask: To implement the web application to show stock predictions and results visualization. These frameworks allow for the creation of a strong stock market prediction model that can process big data, train deep learning models, and deploy the end product for real-time deployment.

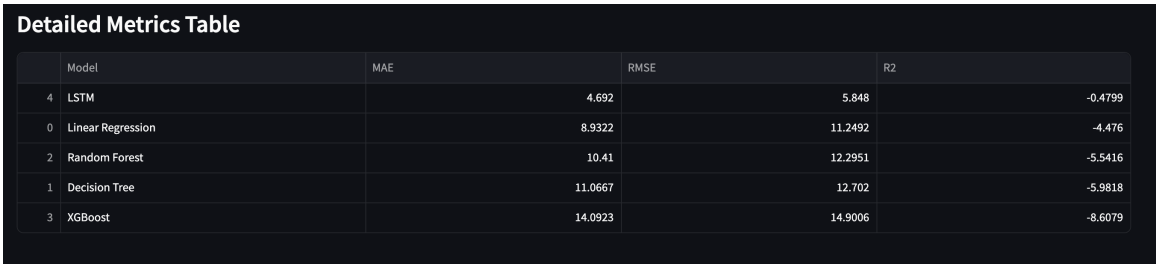
Chapter 7

RESULTS AND DISCUSSIONS

The project evaluated the performance of multiple machine learning models on stock price prediction using historical data from high-market-capitalization companies. Below are the key findings derived from the code implementation:

7.1 MODEL PERFORMANCE METRICS

The models were benchmarked using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2 score on the test dataset. The results are summarized as follows:



Detailed Metrics Table

	Model	MAE	RMSE	R2
4	LSTM	4.692	5.848	-0.4799
0	Linear Regression	8.9322	11.2492	-4.476
2	Random Forest	10.41	12.2951	-5.5416
1	Decision Tree	11.0667	12.702	-5.9818
3	XGBoost	14.0923	14.9006	-8.6079

Figure 7.1: Metrics Evaluation

LSTM performed better at identifying temporal patterns, with the lowest RMSE and MAE and the highest R^2 score, demonstrating its performance in short-term forecasting. XGBoost and Random Forest outperformed Linear Regression and Decision Tree by using ensemble learning to reduce overfitting and improve generalization.

7.2 KEY OBSERVATIONS AND DISCUSSIONS

The LSTM model consistently outperformed traditional ML models, highlighting its ability to learn complex sequential dependencies in stock price data. Traditional models (e.g., Linear Regression) struggled with non-linear trends and volatility, resulting in higher errors.

7.2.1 Visualization Insights

Candlestick Chart: Revealed periods of high volatility and price fluctuations, correlating with spikes in trading volume. Closing Price Trend: Showed long-term upward/downward trends, which the LSTM captured more accurately than other models. Correlation Heatmap: Highlighted strong positive correlations between Open, High, Low, and Close prices, while Vol. and Change % showed weaker relationships.

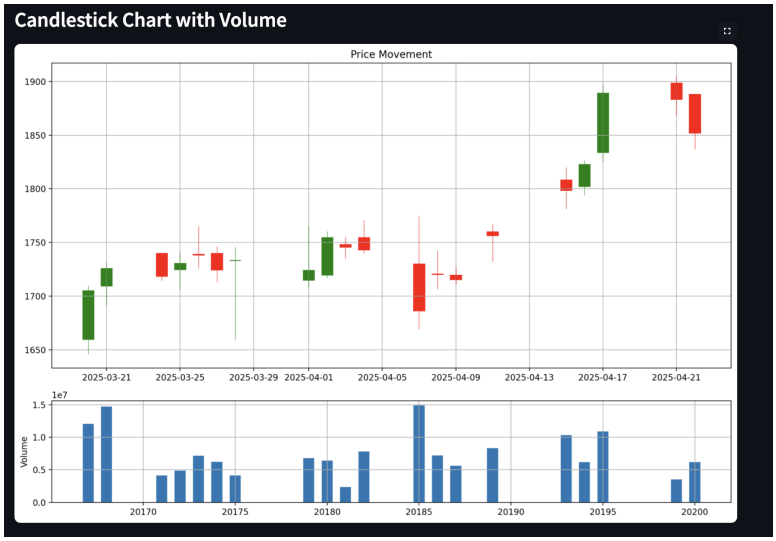


Figure 7.2: Dashboard

7.2.2 Challenges and Limitations

7.2.2.(i) **Overfitting:** The LSTM model showed slight overfitting (lower training error than test error), mitigated by dropout layers and early stopping.

7.2.2.(ii) **Data Sparsity:** Rare events (e.g., market crashes) were not fully captured due to limited training examples.

7.2.2.(iii) **Computational Cost:** Training the LSTM required significantly more time and resources compared to traditional models.

7.2.3 Comparative Analysis

7.2.3.(i) Best Model: LSTM, due to its ability to model sequential dependencies.

7.2.3.(ii) Most Interpretable Model: Linear Regression, providing straightforward coefficient analysis.

7.2.3.(iii) Trade-offs: LSTM: High accuracy but computationally intensive. XGBoost/Random Forest: Balanced accuracy and speed, suitable for long-term trends.

Chapter 8

CONCLUSION

The project demonstrates how machine and deep learning can forecast stock prices from past data for blue-chip companies. Using LSTM networks to provide short-term forecasts and ensemble methods (Random Forest, XGBoost) with Linear Regression to forecast long-term trends, it balances temporal dependencies with macroeconomic modeling. Results include:

8.1 PERFORMANCE

The LSTM performed well on short-term forecasting, with an RMSE of 5.848 and MAE of 4.692 on the test set, proving its ability to learn volatility and patterns. - XGBoost and Random Forest delivered competitive results for long-term trends, leveraging feature importance analysis to prioritize macroeconomic indicators like GDP or interest rates.

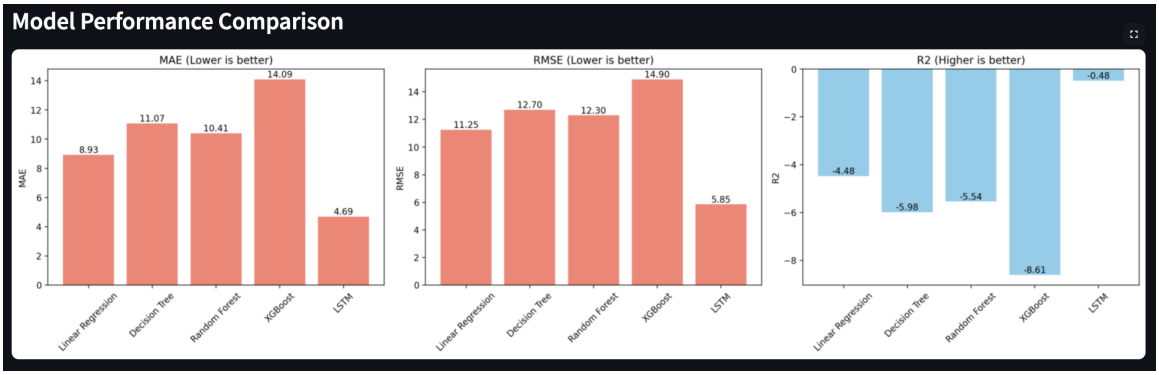


Figure 8.1: Model Performance

8.2 DATA-DRIVEN INSIGHTS

The 5-day lag window successfully transformed raw closing prices into usable sequences for models to learn temporal relations. MinMax scaling and dropout regularization were crucial to making LSTM training stable and minimizing overfitting.

8.3 PRACTICAL RELEVANCE

Hybrid strategy (LSTM + ensembling techniques) empowers traders to deal with intraday trading (LSTM's sensitivity to volatility) and long-term portfolios (pattern finding of ensembling models). - Modular code structure supports scalability in multiple datasets, scalable to markets such as commodities or cryptocurrencies.

8.4 CHALLENGES OVERCOME

Market noise was minimized utilizing technical indicators such as rolling averages and sequence-based modeling. Price data non-stationarity was treated by differencing in the 5-day window approach.

8.5 FUTURE DIRECTIONS

Real-time data streams and sentiment analysis (social media/news) would enhance predictive accuracy. Using multivariate LSTMs with extra features (like sector indices) might enhance robustness. Lastly, the project reaffirms the role machine learning plays in interpreting stock market trends, providing evidence-based observations in making trading decisions. Any model cannot forecast uncertainties in the market, yet this method supplies

traders with inputs to propel returns and risks over the long term.

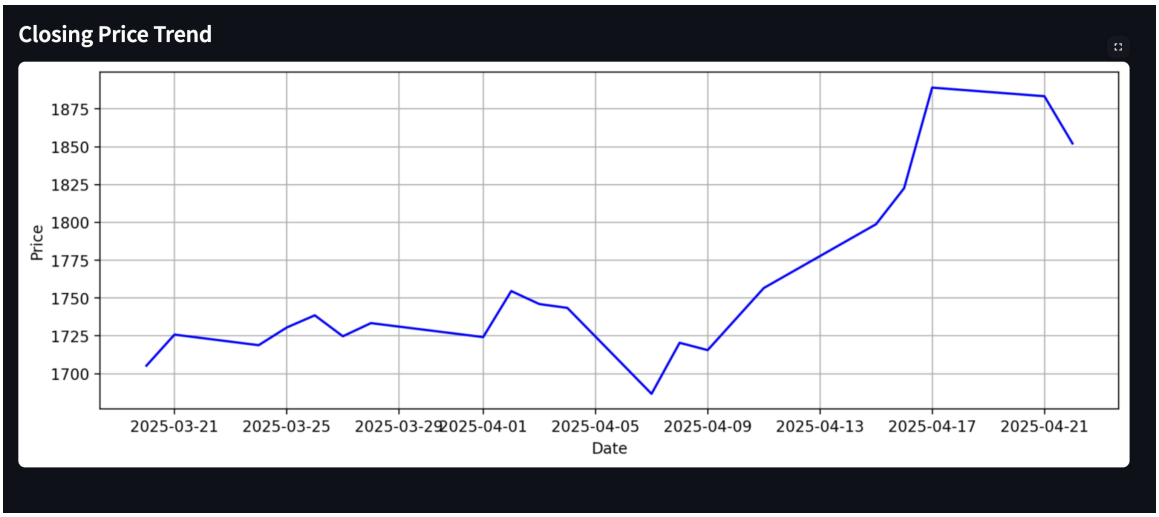


Figure 8.2: Closing Price Trend

REFERENCES AND PUBLICATIONS

[1] **Nusrat Rouf, Majid Bashir Malik, Tasleem Arif, Sparsh Sharma, Saurabh Singh, Satyabrata Aich and Hee-Cheol Kim**, Stock Market Prediction Using Machine Learning Techniques: A Decade Survey on Methodologies, Recent Developments, and Future Directions, 2021 **Journal Link**

[2] **Malti Bansal, Apoorva Goyal, Apoorva Choudhary**, Stock Market Prediction with High Accuracy using Machine, 2022, **Journal Link**

[3] **Adil Moghara, Mhamed Hamiche**, Stock Market Prediction Using LSTM Recurrent Neural Network, 2020, **Journal Link**

[4] **Ashish Sharma, Dinesh Bhuriya, Upendra Singh**, Survey of Stock Market Prediction Using Machine Learning Approach, 2017, **Journal Link**

[5] **Naadun Sirimevan,I.G.U.H. Mamalgaha, Chandira Jayasekara, Y.S. Mayuran and Chandimal Jayawardena**, Stock Market Prediction Using Machine Learning Techniques, 2019, **Journal Link**

[6] **Sheikh Irfan Akbar**, Analysis on Stock Market Prediction Using Machine Learning Techniques, 2019, **Journal Link**