STOCK MARKET PREDICTION USING MACHINE LEARNING

Ayush Nichal [1], Ishan Nalawade [2], Bhavesh Naranje [3], Chaitanya Kapulkar [4]

[1]*Department of Computer Engineering*

[2]*Department of Computer Engineering Zeal College of Engineering and Research Pune, India*

ABSTRACT:

Forecasting stock prices is a challenge in the stock market because there are no clearly defined prediction rules but some of the superior return rates exist on public markets. As stock prices are borne out of the students' room they tend to be volatile. While they are volatile, one can chart stock prices and other statistical means, which will help today's indicated investors choose wisely the company they want to spend their money on. In this study, we will look at consumer spending and which company you are most interested in spending on. It is also very important to mention that with this simple project idea using the dash library, we can more easily and dynamically plot financial data of the company you are interested in spending your money by learning about the provided tabular data from December Yu finance python library. In addition, using machine learning algorithms, we can look ahead and predict future stock prices. This is great for anyone who is on the beginning or intermediate journey in starting python or data science or an analytical exercise for anybody that is already in this field. With students in mind, it would be for the intermediate level thinkers who have tried both Python/ML before.

Keywords: Stock Market Software, Artificial Neural Network, Dash Python, yfinance, Forecasting

[1] INTRODUCTION

Today, the financial markets continue to be a significant part of the meaning of globalization and the stock markets have the greatest significance for economic activity. Therefore, to investors, analysts and policymakers, it is equally meaningful to get as accurate of a picture and forecast of the stock prices as they can. The project titled "Visualization and Trend Forecasting of Stocks" is aimed at designing a special web application to visualize stocks trends for the chosen company and to apply several major technical indicators, including Exponential Moving Average (EMA).

Visualization helps bring to light other aspects of the data that are not easily seen when you are working with the data at the raw tabular level. EMA is just one of many technical indicators the traders use to eliminate the noise (or fluctuating data) to determine trends. The ultimate aim is to predict future movements of the stock to help all the stakeholders with the best information for their possible action.

The present review of the literature will critically analyse the contemporaneous literature in stock visualisation, use of indicators and forecasting methods in an attempt to provide an understanding of the contemporary scholarship in this domain.

Elementary individual investors have taken advantage of money models to improve their understanding of the market and generated a profitable investment. There is ample data available for analysis and processing regarding the volatility of stock data. Investors make educated guesses based on data analysis. They read the news, review the company's history, purchase trends, and many other factors that go into speculation.

The mainstream theory is that stock prices are random and unpredictable. This begs to ask the question why top companies, such as Morgan Stanley and Citigroup hire quantitative analysts to develop predictive models. This paper hopes to employ Machine Learning Model, to suddenly create predictive analysis of stock prices.

[2] LITERATURE REVIEW

The prediction of stock market movements has been a long-standing challenge due to the volatile, nonlinear, and time-dependent nature of financial data. Traditional statistical methods, such as autoregressive and moving average models, often fail to capture these complexities effectively [3][4]. To overcome these limitations, recent research has increasingly focused on machine learning (ML) and deep learning (DL) techniques, including Support Vector Machines (SVM), Random Forest, k-Nearest Neighbors (k-NN), Gradient Boosting, Artificial Neural Networks (ANN), and Long Short-Term Memory (LSTM) networks [2][5][6][7][8].

Among these, LSTM models have consistently demonstrated superior performance in modeling temporal dependencies and capturing complex time-series patterns. For instance, Jabed (2024) reported that LSTM achieved the highest predictive accuracy for both domestic and international datasets, outperforming Prophet and Random Forest models in short-term forecasting tasks [1]. Similarly, studies comparing LSTM with Recurrent Neural Networks (RNN), Support Vector Regression (SVR), and other conventional models confirm its robustness and higher accuracy across diverse stock market datasets [3][4][8]. Prophet, however, has shown notable advantages in long-term forecasting by effectively modeling seasonality and holiday effects [1].

Systematic reviews further reinforce the effectiveness of hybrid and deep learning approaches, which outperform traditional ML models, particularly when combined with robust preprocessing, feature engineering, and hyperparameter optimization [2][5][7]. Nevertheless, challenges such as overfitting, inconsistent data quality, and the inherent unpredictability of financial markets remain. To address these issues, scholars increasingly recommend the integration of ensemble methods, alternative data sources (e.g., news sentiment and social media trends), and explainable AI techniques to enhance both accuracy and interpretability [5][7].

Overall, the literature indicates that while no single model guarantees perfect prediction accuracy, deep learning—particularly LSTM—emerges as the most effective technique for short-term stock price forecasting. Furthermore, hybrid and ensemble methods hold considerable promise in improving long-term prediction performance and providing more robust insights into market trends.

[3] METHODOLOGY

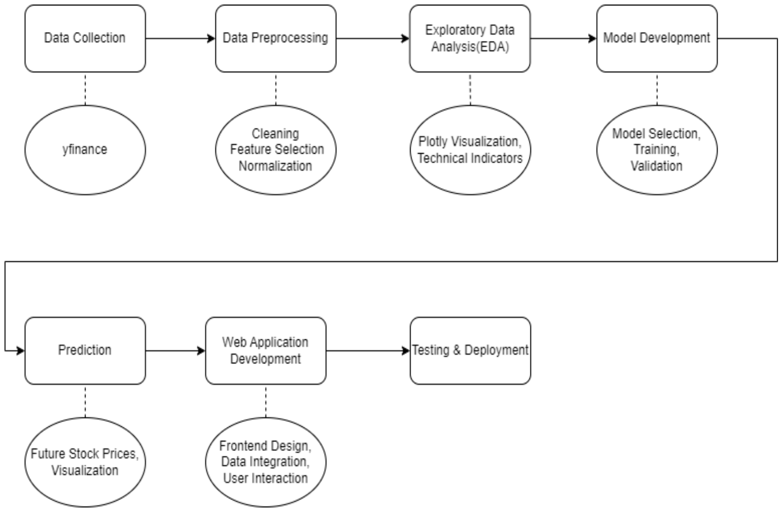

Figure: 1. Methodology diagram

3.1 Data Collection

In this project, the first step is Data Collection. Stock data will be collected via a library called yfinance: historical stock prices and volumes, as well as any other useful financial information about a selected company for a specified period. Preprocessing should generate enough information for proper analysis and prediction of trends, which is the next stage.

Dataset

Herein is a comprehensive dataset of historical stock price data organized and obtained from the yfinance source, which is a very trusted source of financial data. In this dataset, which is the one we will use for the purposes of analysis and forecasting, contains many of the essential features required in general stock analysis and price forecasting.

To improve the predictive modeling, technical indicators will be calculated and appended to the dataset. Examples of these include the Exponential Moving Average

(EMA) which gives more weight to recent prices allowing current trends to be determined more accurately.

The data, will be preprocessed very carefully to address instabilities in the data structures, such as missing values or outliers, and normalizing it, such that all features are weighted equally to the machine learning models. The timeframe for this dataset will be such that enough, data points can be included to capture historical trends to adequately predict future stock prices for the company of interest. This dataset lays the foundation for the entire project, thus driving both the visualization and predictive aspects of the application.

Section	**Description**
Dataset Overview	The dataset is composed of historical stock data collected from the yfinance library. It includes key financial metrics such as stock prices, volume, and additional technical indicators. This data is essential for analyzing trends and forecasting future stock prices.
Data Source	Data is fetched using the yfinance library, which provides reliable stock market data, including historical prices, volumes, and various financial metrics.
Data Collection Period	The dataset spans a specific period relevant to the selected company. The time frame is chosen to ensure adequate data for trend analysis and stock price prediction, usually covering several months to years of historical data.
Dataset Composition	- Date: Trading date (used as the index). - Open: Opening price of the stock. - High: Highest price during the day. - Low: Lowest price during the day. - Close: Closing price. - Adj Close: Adjusted closing price after corporate actions. - Volume: Total number of shares traded. - Technical Indicators: Exponential Moving Average (EMA) and other calculated indicators.
Features	- Quantitative Features: Open, High, Low, Close, Adjusted Close, Volume. - Derived Features: EMA and other technical indicators for trend analysis. - Date: Serving as the temporal index for time-series analysis.

Data Cleaning	The dataset is cleaned to remove any missing values, outliers, or inconsistencies that might affect the accuracy of trend analysis or machine learning models.
Feature Selection	Features such as Open, High, Low, Close, and Volume are selected for the machine learning models, along with calculated indicators like the Exponential Moving Average (EMA) to improve trend analysis.
Normalization	The data is normalized to bring all features onto a similar scale, ensuring that no feature dominates the machine learning models and that all contribute equally to the predictions.
Timeframe	The dataset includes data for a specified period for the selected company, ensuring a sufficient number of data points for training and testing machine learning models, as well as for visualizing long-term trends.
Dataset Size	The dataset's size is defined by the chosen period, typically ranging from hundreds to thousands of rows, each representing a single trading day and its corresponding financial metrics.
Target Variable	The primary target variable for prediction is the Close price, which represents the stock’s value at the end of the trading day, offering a crucial indicator for forecasting future stock prices.

Table 1

3.2 Data Preprocessing

Subsequently, the Data Preprocessing stage prepares data for analysis. Cleaning the raw data consists of addressing inconsistencies, including missing values or outliers. Regarding feature selection, the important features include Open, High, Low, Close, and Volume. Additional features that are calculated include the Exponential Moving Average (EMA) and then added to the dataset. Normalization is conducted to ensure uniformity among the data and to improve machine learning model performance.

3.3 Exploratory Data Analysis(EDA)

Exploratory Data Analysis (EDA) comes after preprocessing. The first visualization using Plotly will show the trend, patterns, and anomalies in the stock data. We see the application of technical indicators such as the EMA. These indicators are utilized to establish the trend by smoothing the noise into the prices that reveal the better view of the stock.

3.4 Model Development

Model Development: In this section, we will test various machine learning models, such as Linear Regression and LSTM, in terms of their potential predictive power for future stock prices. To train the models, we will use historical stock data but to ensure accuracy and prevent overfitting, approximately 20% of the data will be used for validation.

This algorithm utilizes a Long Short Term Memory (LSTM) neural network for stock price predictions based on its historical closing prices. LSTM is a unique type of recurrent neural network (RNN) that can learn long-term dependencies in sequential data and is therefore especially applicable to time series forecasting such as stock price prediction.

Why LSTM?
Time-series considerations: Because stock prices change in a sequence, LSTM is able to extract underlying processes in data more effectively than a standard feed-forward neural network can. Long-term dependencies: Because of its internal structure, LSTM is capable of remembering dependencies back in time. Stock price forecasting models must account for long-term dependencies where past prices can influence price changes much further in the future.

Preventing vanishing gradients: The design of the LSTM, particularly the gate structures to control the flow of information, allows the model to avoid training problems such as vanishing gradients, which is key to enabling the LSTM architecture to learn longer term trends in the data.

Model input: The model considers the closing ('Close') price to be the input model-specific feature.

Data:
The historical stock information specific to any given company is retrieved using the yfinance package.

The yfinance data contains:
Open, High, Low, Close price, Volume, Adjusted close price. In the context of this study however, the model will only consider df['Close'], which is then scaled prior to being passed to the LSTM model.

Model architecture
Layers:
The model consists of two LSTM layers, both with 50 units (neurons). The first LSTM layer returns the full sequence 2 4 4 9 15 20 28 38 43 50 53 59 (i.e. return_sequences=True) to the

second LSTM layer, which returns after running through all time steps only the last output in the sequence to the next layer. In this case, there are two Dense layers, in which the final output layer produces one output, the stock price prediction.

Optimizer and Loss:
The model uses the Adam optimizer that combines benefits of Momentum and RMSProp optimizers and minimizes the mean squared error (MSE), commonly used for regression problems (e.g., stock price prediction).

Processing flow
Training:
The model is trained on the historical data for 60 days to estimate the price of day 61. The model is trained for a single epoch with a batch size of 1.

Prediction:
Using the trained model to predict stock prices on the testing set (which is the remaining 20% of data). The predicted prices were then scaled back from the MinMaxScaler to the original price range.

Evaluation:
The root mean square error (RMSE) is used to validate the performance of the model. RMSE implies how close the estimated price is to the actual price.

Visualization:
The prices from the training set, actual prices from the validation (test) set, and predicted prices are plotted together on top of each other in a graphical format to visually assess how well the model performed.

3.5 Prediction

Following the completion of the model, a prediction phase can be initiated. The model selected in the previous stage will use the dates specified by the user and generate predictions of future closing price. The code will generate predictions, and finally, the code will visualize the predicted prices and historical prices next to each other in order to compare predicted and actual prices, allowing the user to evaluate the "quality" of the model's predictions.

3.6 Web Application Development

Development of Web Applications: This is the last part of the project. The front end of this application utilizes Dash HTML components for display, slapping CSS on for styling. The back end gets live data tthrough yfinance, and tosses the data in directly to the model, for making the predictions. It allows the user to input a company name and dates of interest, and then interactively display the visualized historical data and predicted future prices.

[4] RESULT

Model Results –

Date	Close	Predictions
2021-03-11	121.959999	121.983498
2021-03-12	121.029999	121.991493
...	...	...
2024-09-13	222.500000	218.733170
2024-09-16	216.320007	218.830887
2024-09-17	216.789993	218.082382
2024-09-18	220.690002	217.190598
2024-09-19	228.869995	216.901672

Table 2

[6] CONCLUSION

From the use of LSTMs on this project, it has been learned that deep learning can assist with increasing accuracy in stock price prediction. With the aid of historical data and data mining the system provides the investor with relevant information on which way the stock market is headed. However, and like any other stock prices prediction methods, this method has its disadvantages due to the number of outside factors that can affect the high variability of stock prices.

Improvements in the future would be to incorporate a non-numeric factor, improve the stability of the model and, possibly, implement real-time systems which would help with accuracy and generalization of stock price predictions. Overall, the usefulness of stock market visualization and predictability is strong because it can serve as a good proxy for helping to deliver guidance in making choices that deal with uncertainty in stock markets, if subsequent development of the model can be realized to incorporate wider dataset and features.

REFERENCES

[1] Jabed, "Feasibility of Forecasting Stock Prices Using Machine Learning Algorithms," 2024.

[2] "Machine Learning Based Stock Market Trend Prediction: A Study of Stock Price Forecasting Techniques," *ScienceDirect*, 2023.

[3] "Research on Stock Price Prediction Method Based on Deep Learning," in *Proc. 2nd Int. Conf. Information Technology and Computer Application (ITCA)*, 2020, pp. 69–73.

[4] G. Bathla, "Stock Price Prediction using LSTM and SVR," in *Proc. 6th Int. Conf. Parallel, Distributed and Grid Computing (PDGC)*, 2020, pp. 211–214.

[5] Md. Mobin Akhtar Abu Sarwar Zamani,"Stock Market Prediction Using Statistical Data and Machine Learning Algorithms," *Procedia Computer Science*, vol. 216, pp. 96–102, 2023.

[6] M. M. Akhtar et al., "Stock Market Prediction Based on Statistical Data Using Machine Learning Algorithms," *Journal of King Saud University – Science*, vol. 34, no. 4, p. 101940, 2022

[7] Md. M. Akhtar and A. S. Zamani, "Stock Market Prediction Using Statistical Data and Machine Learning Algorithms," *Procedia Computer Science*

[8] A. Subasi, F. Amir, K. Bagedo, A. Shams, and A. Sarirete, "Stock Market Prediction Using Machine Learning," *Procedia Computer Science*, vol. 194, pp. 173–179, 2021.

Author[s] brief Introduction

- **Ishan Nalawade** is a final-year undergraduate student in the Department of Computer Engineering at Zeal College of Engineering and Research, Pune.

- **Bhavesh Naranje** is pursuing a B.E. in Computer Engineering at Zeal College of Engineering and Research, Pune.

- **Ayush Nichal** is an undergraduate student in the Department of Computer Engineering at Zeal College of Engineering and Research, Pune.

- **Chaitanya Kapulkar** is a final-year student in Computer Engineering at Zeal College of Engineering and Research, Pune.