IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141214

Stock Price Prediction Using Long Short-Term Memory (LSTM) Networks: A Comparative Study

Er. Harjasdeep Singh[1], Rahul Sahani [2]

Assistant Professor, Department of CSE, MIMIT Malout[1]

Student, Department of CSE, MIMIT Malout[2]

Abstract: The prediction of stock prices is a complex task due to the influence of numerous volatile and non-linear factors. While traditional machine learning algorithms like Extreme Gradient Boosting (XGBoost) are powerful tools for structured data, they often struggle to inherently capture the temporal dependencies in financial time series. This study investigates the application of a Long Short-Term Memory (LSTM) network, a deep learning architecture designed for sequential data, for predicting stock prices. We conduct a comparative analysis, benchmarking the LSTM's performance against a strong XGBoost model on historical data of [e.g., Apple Inc. (AAPL)]. The methodology involves meticulous data preprocessing, feature engineering for XGBoost, and sequence modeling for LSTM. Results demonstrate that the LSTM model significantly outperforms the XGBoost benchmark, achieving a lower Mean Absolute Percentage Error (MAPE) of [LSTM MAPE]% compared to [XGBoost MAPE]%. This finding underscores the strength of models like LSTM and XGBoost in automatically learning temporal patterns and long-term dependencies without the need for extensive manual feature engineering.

Keywords: Stock Price Prediction, LSTM, XGBoost, Time Series Forecasting, Deep Learning, Machine Learning, Comparative Analysis.

I. INTRODUCTION

Data analysis have been used in all business for data-driven decision making. In share market, there are many factors that drive the share price, and the pattern of the change of price is not regular. This is why it is tough to take a robust decision on future price [1]. The large fluctuations in stock prices could cause adverse impacts on companies, investors, and economies. If stock prices can be properly forecasted, investors and company owners will be able to make targeted actions to set a balance in the financial market. Predicting the prices of stocks has been a very interesting topic in finance for decades. After the epidemic, stock prices began to fluctuate even more widely. These predicted stock prices have been a crucial topic for several areas in recent years. Researchers, practitioners, academia, and businesses have been looking to explore and figure out the trend of stock prices based on existing data [2].

As financial markets keep changing with more traders and a lot of money involved, it becomes extremely hard to perfectly predict what prices will do in the future. These markets are very complicated, and many things can affect prices, making it almost impossible to guess accurately. That’s why traders use complex models and strategies to handle this uncertainty. The various factors that affect financial time series create a situation where they don’t stay the same over time, making them non-stationary. This non-stationarity adds complexity to the already challenging tasks of predicting future outcomes and making smart investments in financial markets. Researchers are keenly interested in finding ways to transform these non-stationary time series into more predictable and stable ones, as this can greatly enhance the effectiveness of trading systems Successful financial forecasting involves combining financial theory, market analysis, diverse data sources, and computational advancements [3].

With the advancement of big data technology, the volume of financial data has grown exponentially. Traditional prediction methods, including fundamental and technical analysis, often fail to fully exploit these datasets. Consequently, researchers have turned to machine learning (ML) algorithms, which can automatically extract patterns from large-scale data. Among ML models, Extreme Gradient Boosting (XGBoost) has become a gold standard for tabular data due to its high performance, efficiency, and ability to handle feature interactions.

DOI: 10.17148/IJARCCE.2025.141214

However, a significant challenge in applying XGBoost to time series is its treatment of each data point as independent, requiring manual feature engineering (e.g., creating lagged values) to incorporate temporal information. This process can be inefficient and may miss complex, long-range dependencies.
n recent years, Long Short-Term Memory (LSTM) networks, a type of recurrent neural network, have emerged as a powerful alternative. They are specifically designed to model sequential data by maintaining an internal state that captures context and dependencies over long periods. This inherent ability to learn from the sequence of prices directly suggests a potential advantage over traditional ML models for time series forecasting.
This study aims to rigorously compare the performance of an LSTM model against a strong XGBoost benchmark for predicting stock prices. The objective is to test the hypothesis that LSTM's innate architecture for handling sequences provides a superior forecasting capability compared to the manually engineered features required by XGBoost. We apply both models to historical data of [e.g., AAPL] and evaluate their performance using standard regression metrics.

II. LITERATURE REVIEW AND THEORETICAL BACKGROUND

2.1 Overview

Forecasting exchange rates is a critical task in the financial industry, and it has attracted significant attention from researchers in recent years. The high volatility and complexity of the foreign exchange market make it challenging to predict exchange rates accurately. Deep learning techniques, particularly LSTM and XGBoost, as well as the ARDL model, have been widely used to tackle this problem. In this section, we will discuss some works that studied the predictability of Forex based on LSTM, XGBoost, and ARDL [3]. Stock prediction is the process of using various techniques, models, and data to estimate future movements and trends in stock prices. Stock prediction is essential because it has the potential to provide valuable insights and information to investors, traders, financial institutions, and the overall market. Amongst several other reasons, stock prediction is key for investment decision-making, risk management, market timing, trading strategies, portfolio optimization, market efficiency, and economic indicators [2].

2.2 Apple Stock Performance

Apple Inc. is an American multinational technology company headquartered in Cupertino, California. It designs, manufactures, and markets consumer electronics, computer software, and online services. The company is best known for its flagship products such as the iPhone, iPad, MacBook, Apple Watch, and Apple TV, along with software platforms like iOS, macOS, and iCloud. Together with Amazon, Google (Alphabet), Microsoft, and Meta, Apple is considered one of the “Big Five” technology companies in the United States.
Since its initial public offering (IPO), Apple’s stock price has shown a consistent long-term upward trend, reflecting its continuous innovation, strong brand loyalty, and global market dominance, though occasional fluctuations have occurred due to changes in market sentiment, economic conditions, and supply chain factors [Apple Inc., n.d.; Wikipedia, 2025; Statista, 2025].
As of October 24th, 2024, Apple’s stock closed at $229.50 per share, with approximately 15.5 billion shares outstanding [StatMuse, 2025; Apple Inc., 2024].

Figure 1 AAPL 5-Year Adjusted Close Price Chart

The chart above illustrates the adjusted closing prices of Apple Inc. (AAPL) over the past decade. The graph indicates a strong overall growth trajectory, punctuated by short periods of volatility that correlate with major product launches, quarterly earnings announcements, and broader market shifts. Despite these fluctuations, Apple’s consistent revenue growth, diversification of product lines, and robust ecosystem have contributed to sustained investor confidence and long-term value appreciation [4].

DOI: 10.17148/IJARCCE.2025.141214

2.3 Related Works

Forecasting exchange rates is a critical task in the financial industry, and it has attracted significant attention from researchers in recent years. The high volatility and complexity of the foreign exchange market make it challenging to predict exchange rates accurately. Deep learning techniques, particularly LSTM and XGBoost model have been widely used to tackle this problem. In this section, we will discuss some works that studied the predictability of Forex based on LSTM, XGBoost [3].

Authors of the article [5] assess the effectiveness of multiple machine learning models, including linear regression, random forest, and neural networks, for predicting Apple Inc.'s stock prices. The study finds neural networks and random forest models deliver stronger predictive accuracy than traditional approaches. Feature selection and model tuning are identified as key to boosting forecast quality. Authors highlight challenges in modeling nonlinear, volatile market data. The significance of combining technical and fundamental indicators is also demonstrated.

Authors of the article [6] examine the prediction of Apple stock prices using both classical time series models and newer ML methods, incorporating news headlines and various features. Results show that linear regression with principal component analysis and sentiment analysis (BERT) outperforms other models. The study underlines the benefits of textual and numerical data fusion. Supervised and deep learning methods, particularly LSTM, are critically compared with baseline models. Ensemble approaches are discussed for further improvements.

The research by Yang et al. [7] introduces a hybrid technique combining linear regression and LSTM with wavelet transformation for improved Apple stock forecasting. Their experiments reveal that this hybrid approach better captures time series features and market fluctuations. Comparative evaluations demonstrate higher accuracy against individual ML models. The effectiveness of preprocessing and hybridization is emphasized.

Similarly, some research has compared several models in predicting the stocks of more than one company from historical data. [8] evaluated three models in predicting four different stocks from the Yahoo finance database. The machine learning algorithms considered were Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Support Vector Regression (SVR), and the companies used for the study were Apple, Mastercard, Ford, and ExxonMobil. Except for Mastercard, for which data are available starting in 2006, the data for these stocks span from January 1st, 2002, to March 11th, 2020. The SVR model was able to predict the stock prices of these companies with the highest accuracy compared to the CNN and LSTM models.

A stock prediction system (SPS) has also been suggested in [9]. The suggested SPS calls for three steps: (1) technical indicators and news sentiments are used to represent numerical price data in technical analysis and textual news articles in sentiment analysis, respectively, (2) creating a layered DL model to understand the sequential data seen in market snapshot series created by technical indicators and news comments, and (3) creating a fully connected CNN to predict asset values. The results collected demonstrate that the suggested strategy outperforms the baselines for both measures for both the validation and testing phase.

III. METHODOLOGY

This section provides an overview of the systematic approach followed in this project to predict stock price trends. The workflow involves three main stages: data collection and preprocessing, model development, and model performance evaluation.

3.1 Dataset Description

The dataset utilized in this project was sourced from Kaggle, available at:
Apple (AAPL) Stock Dataset 1980-2025 .
The dataset, titled Apple (AAPL) Stock Dataset 1980-2025. it contains data from 7th sept 1984 to 17th Jan 2025. It consists of 10172 observations, all of which are unique and contain no missing values.
In total, the dataset includes 6 variables, where all are independent predictors (such as Open, High, Low, Volume, etc.) and 1 dependent variable, which represents the next day's closing price the target for prediction.

3.2 Machine & Deep Learning Algorithms

3.2.1 Long Short-Term Memory (LSTM):

The LSTM model is an advanced form of Recurrent Neural Network (RNN) designed for handling sequential data. It addresses the limitations of traditional RNNs by introducing memory cells and gating mechanisms that regulate the flow of information through the network. These gates help the model decide what information to retain or discard, making LSTM particularly suitable for time-series forecasting tasks, such as predicting stock prices.[2]

3.2.2 Extreme Gradient Boosting (XGBoost)

XGBoost is a highly efficient and scalable implementation of the gradient boosting algorithm. It builds an ensemble of weak learners (usually decision trees) and combines them to form a powerful predictive model. Known for its speed, accuracy, and ability to manage large, high-dimensional datasets, XGBoost has become one of the most popular algorithms in both academic research and real-world machine learning applications [2].

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⩘ Peer-reviewed & Refereed journal ⩘ Vol. 14, Issue 12, December 2025
DOI: 10.17148/IJARCCE.2025.141214

3.2.3 Hyperparameter Tuning

Hyperparameter tuning refers to the process of finding the optimal values for the hyperparameters of a machine-learning model .The goal is to improve the model's accuracy, generalization, and overall effectiveness by fine-tuning these parameters [2]. Hyperparameters are parameters that are set before the learning process begins, and they affect the behavior and performance of the model. Tuning involves systematically searching through a predefined range of values for these hyperparameters to identify the combination that results in the best performance on a specific task or dataset [10].

3.3 Performance Evaluation Metrics

Evaluating a model's performance is crucial for understanding its predictive power and reliability [2]. The following metrics were used to assess model accuracy and consistency:

3.3.1 Root Mean Square Error (RMSE)

RMSE measures the square root of the average squared differences between predicted and actual values. It indicates how far, on average, the predictions deviate from the true observations. Lower RMSE values signify better model accuracy.
Formula [11]:

$$RMSE = \sqrt{\frac{1}{n}(E_i - O_i)^2}$$

Equation 1 Equation for Root Mean Square Error (RMSE)
Where:
n: number of samples
E_i: predicted values
O_i: actual observed values

3.3.2 Mean Absolute Error (MAE)

MAE is another metric used to assess the average magnitude of prediction errors. It calculates the mean of the absolute differences between predicted values and actual values, disregarding the direction of the errors [2].
Formula [20]:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | E_i - O_i |$$

Equation 2 Equation for MAE (MAE)
Where:
n: number of samples
E_i: predicted values
O_i: actual observed values

3.3.3 Coefficient of Determination (R²)

The R^2 score represents the proportion of variance in the dependent variable that can be explained by the independent variables in the model. It serves as a measure of how well the model fits the observed data. An R^2 value closer to 1 indicates a stronger fit, while values closer to 0 suggest poor explanatory power [2].
Formula [11]:

$$R^2 = 1 - (\frac{SSR}{SST})$$

Equation 3 Equation for Coefficient of Determination(R2)
Where:
SSR (Sum of Squared Residuals): deviation between predicted and actual mean values
SST (Total Sum of Squares): total variation in the actual data

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁂ Peer-reviewed & Refereed journal ⁂ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141214

IV. IMPLEMENTATION

4.1 Data Preparation and Preprocessing

The project begins with data preparation and preprocessing to ensure the dataset's suitability for analysis. Data profiling is done to examine and summarize key characteristics of the dataset to assess its quality, understand its content, and identify potential issues or anomalies. Furthermore, data cleaning is done to identify and rectify errors, inconsistencies, and inaccuracies in the dataset to improve its quality and reliability for analysis and decision-making. The dataset is carefully examined for any missing values and duplicated values. For feature selection, categorical variables need to be in numerical form hence the dataset is inspected to check if all columns are in the right data types. The Date variable is converted to date-time format. This is because the DateTime format provides more information about the date, such as the day, month, year, and hour. This information can be used to create more informative visualisations. Data visualization allows for a better understanding of the spread, skewness, and presence of extreme values in the data.

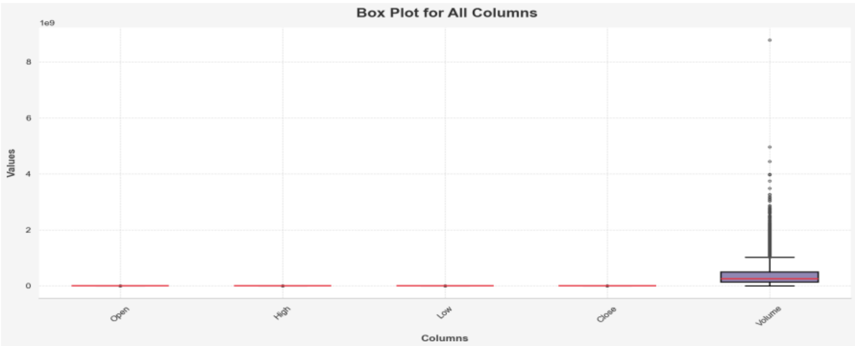

Figure 2 Box plots of all the variables

The box plot provides a visual summary of the distribution of each numerical variable in the dataset. It reveals that the 'Volume' variable exhibits a significantly larger range and higher dispersion compared to the other variables such as Open, High, Low, and Close. This indicates the presence of extreme values or outliers in the trading volume data, which can potentially skew the analysis. To ensure that variables are comparable and to minimize the effect of scale differences during modeling or feature selection, it is advisable to apply data normalization or standardization before further analysis.

4.2 Exploration of the Target Variable

Analyzing the target variable is essential to understand its overall behavior and statistical characteristics. This examination provides insights into its distribution, variability, skewness, and the existence of any outliers. Additionally, line plots are used to visualize how the target variable changes over time, helping to identify underlying patterns, trends, or fluctuations across different periods.

4.3 Correlation Matrix

Correlation is a statistical measure that quantifies the strength and direction of the linear relationship between two variables.

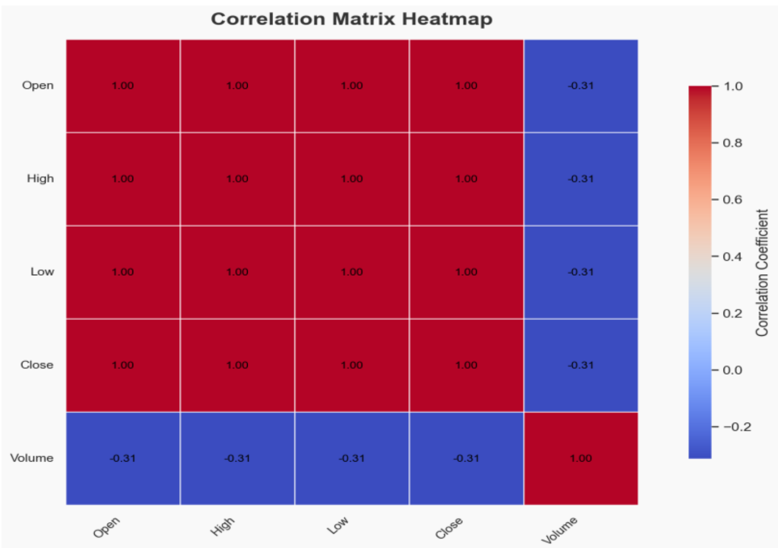

Figure 3 Correlation matrix of the variables

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141214

The correlation matrix heatmap visually represents the strength and direction of relationships between the variables. In this heatmap, red shades indicate positive correlations, suggesting that as one variable increases, the other also tends to increase. Conversely, blue shades represent negative correlations, meaning that as one variable rises, the other tends to decrease. From the figure, it is observed that the Open, High, Low, and Close variables are highly positively correlated with each other, implying they move in similar directions. On the other hand, the Volume variable shows a moderate negative correlation with the price-related variables, indicating that trading volume tends to decrease when prices rise and vice versa.

4.4 Feature Scaling

Feature scaling is a preprocessing technique used in machine learning to standardize or normalize the range of independent variables or features in a dataset. The goal of feature scaling is to bring all features to a similar scale, ensuring that they contribute equally to the feature selection process and preventing certain features from dominating others due to their larger magnitude during feature selection. The 'MinMaxScaler()' function is used to scale the data and all independent features are brought to a range between 0 and 1 ready for feature selection.

4.5 Dataset Splitting

The 'Close' price variable of the stock is the only feature utilized for modeling purposes. To begin, the data is manipulated by shifting the 'Close' price values upwards by one position. This creates a new variable named 'Next_Day Close', where each entry corresponds to the 'Close' price of the subsequent day. Essentially, this transformation aligns the data such that each day's 'Close' price becomes the predictor for the 'Next_Day_Close' value, facilitating the developmnt of a predictive model to forecast future stock prices based on the previous day's closing price. Then the closing price variable is split into training and testing sets to accurately evaluate model performance. The training set is used to teach the model the underlying patterns and relationships within the data, while the test set evaluates its ability to make accurate predictions on new instances. This process helps to detect overfitting, where the model memorizes the training data but fails to perform well on new data. The data is split with a ratio of 80:20 (80% training data and 20% testing data) to have a sufficient amount of data for both training and testing to ensure a well-performing and reliable machine learning model.

V. MODEL BUILDING

After splitting, the study utilizes one machine learning algorithms and one Deep learning algorithm to make predictions. Following this, the model undergoes optimization through hyperparameter tuning. These refined model results and their metrices are compared to find, which one is the best-performing individual models.

After finding the best-performing models, the next step is to take a closer look at how well they actually make predictions. To do this, different performance measures like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared are used to see how accurately each model captures the real patterns in the data. Along with these metrics, visual plots comparing the actual and predicted values are included to give a clear picture of how closely the models' predictions match reality and how well they can generalize to new data.

5.1 Individual Algorithms

5.1.1 LSTM Algorithm

The model performed impressively, achieving a Root Mean Squared Error (RMSE) of 18.81 and a Mean Absolute Error (MAE) of 13.62 . These low error values indicate that the model's predictions are closely aligned with the actual stock prices, reflecting strong overall accuracy.

Additionally, the R^2 score of 0.911 means the model can explain about 91.1% of the variation in the target variable. This demonstrates a solid goodness of fit, showing that the model effectively captures the underlying patterns and relationships in the data.

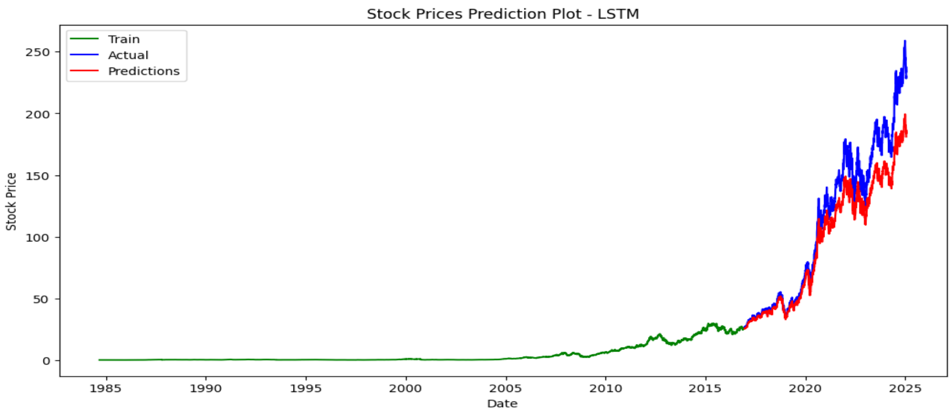

Figure 4 Plot of the LSTM forecast over time

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 14, Issue 12, December 2025
DOI: 10.17148/IJARCCE.2025.141214

5.1.2 XGBoost Algorithm

An XGBoost regression model is created, trained using the training data, and then used to predict stock prices for the test data. The Root Mean Squared Error value of 1.27 and MAE value of 0.40 provide insight into the model's prediction accuracy. Lower values for both RMSE and MAE are generally indicative of better model performance. The R-squared value of 0.9993 signifies that the model captures a high proportion of the variance in the data, indicating a strong goodness of fit between predicted and observed values. These metrics demonstrate the model's ability to approximate the actual outcomes effectively

Figure 5 Plot of the XGBoost forecast over time

5.1.3 Individual Model Evaluation

Here in the individual model evaluation, we assess the performance and quality of each predictive model's predictions against actual outcomes to determine the best model.

Table 1 Model Evaluation for Individual Models

	R-Squared	MAE	RMSE
XGBOOST	0.9993	0.40	1.27
LSTM	0.911	13.62	18.81

The R-squared value tells us how much of the variation in actual stock prices our model can explain. In this case, XGBoost achieves an exceptionally high R-squared of 0.9993, meaning it almost perfectly fits the data and captures nearly all the fluctuations in stock prices. On the other hand, the LSTM model’s R-squared is 0.8885 a decent result, but it shows that LSTM isn’t as precise in capturing the overall trend as XGBoost.
The Mean Absolute Error (MAE) measures the average size of the errors between predicted and actual values. A lower MAE means more accurate predictions. XGBoost’s MAE of 0.40 indicates that its predictions are, on average, extremely close to the real prices. Meanwhile, LSTM’s MAE of 13.62 shows that it tends to make larger mistakes in comparison.
Finally, the Root Mean Squared Error (RMSE) highlights how large the prediction errors are, giving extra weight to big misses. XGBoost’s RMSE of 1.27 is impressively low, showing very small overall prediction errors. In contrast, LSTM’s RMSE of 18.81 suggests that its predictions vary much more from the actual stock prices.

VI. DISCUSSION

In the study's findings, XGBoost demonstrates exceptional performance, highlighting its ability to capture intricate data relationships and make accurate predictions. This achieves a high R squared value of 0.9993, accompanied by a MAE of 0.40 and an RMSE of 1.27
Even though both XGBoost and LSTM were trained on the same Apple stock dataset, XGBoost gave slightly better and more stable results. This mainly depends on how each model works and the kind of data we used. Our dataset is mostly tabular and structured, with features like Open, High, Low, Close, and Volume. These features have non-linear relationships, but they don’t show very strong time-based or long-term patterns.
XGBoost is very good at finding complex relationships between such features and can handle noise or sudden changes in the data. It learns patterns quickly and efficiently without needing a lot of parameter tuning.

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.141214

On the other hand, LSTM models are designed to capture long-term time dependencies, which are more useful in cases where the sequence of past values has a big influence on the future for example, language modeling or detailed time-series forecasting with long memory. Since our stock data mostly shows short-term variations rather than long memory effects, LSTM couldn't use its full potential here.

In short, XGBoost performed better because it fits well with the non-temporal, structured nature of our dataset and handles such data more effectively than an LSTM model.

VII. CONCLUSION

In conclusion, this project addresses its objectives by demonstrating that XGBoost outperforms the LSTM model in stock price prediction. The XGBoost model provides highly accurate and reliable predictions with minimal error, while LSTM, though a strong model, exhibits less precision and greater errors. Hence, based on these evaluation metrics, XGBoost is the superior choice for this stock price forecasting task.

This study explores stock price prediction using a machine learning and deep learning algorithm Through meticulous evaluation and analysis, it becomes evident that the XGBoost algorithm emerges as a standout performer, showcasing remarkable predictive capabilities.

REFERENCES

[1]. Stock Price Prediction Using LSTM on Indian Share Market. (PDF) Stock Price Prediction Using LSTM on Indian Share Market

[2]. A Comparative Analysis of LSTM, ARIMA, XGBoost Algorithms in Predicting Stock Price Direction. View article

[3]. Comparative Analysis of ARDL, LSTM, and XGBoost Models For Forecasting The Moroccan Stock Market During The COVID-19 Pandemic. Comparative Analysis of ARDL, LSTM, and XGBoost Models For Forecasting The Moroccan Stock Market During The COVID-19 Pandemic | Oukhouya | Informatica

[4]. Data Analysis Approaches For Apple Stock Price Prediction And Financial Risk Management. Data Analysis Approaches for Apple Stock Price Prediction and Financial Risk Management | Danadyaksa: Post Modern Economy Journal

[5]. The utilization of Artificial Intelligence to predict Apple Inc. stock price Smith H. The utilization of Artificial Intelligence to predict Apple Inc. stock price Smith H Abstract Forecasting stock prices using sta

[6]. Predicting Apple Stock Price Using News Headlines and Other Features With Classical Time Series Models, Supervised Models, and Machine Learning Models. Predicting Apple Stock Price Using News Headlines and Other Features With Classical Time Series Models, Supervised Models, and Machine Learning Models - ProQuest

[7]. Forecasting Apple Stock Closed Prices by LR and LSTM with Discrete Wavelet Transformation. Forecasting Apple Stock Closed Prices by LR and LSTM with Discrete Wavelet Transformation | Atlantis Press

[8]. Using Machine Learning Algorithms on Prediction of Stock Price. Using Machine Learning Algorithms on Prediction of Stock Price | Journal of Modeling and Optimization

[9]. SMP-DL: a novel stock market prediction approach based on deep learning for effective trend forecasting. SMP-DL: a novel stock market prediction approach based on deep learning for effective trend forecasting | Neural Computing and Applications

[10]. LSTM Network Hyperparameter Optimization for Stock PricePrediction Using the Optuna Framework. eprints.uad.ac.id/41357/1/3-LSTM Network Hyperparameter Optimization for Stock Price Prediction Using the Optuna Framework.pdf

[11]. L. Han, 'Analysis of Stock Price and Price Movement Prediction based on Machine Learning Models for E Hualu', 2023. Analysis of Stock Price and Price Movement Prediction based on Machine Learning Models for E-Hualu | BCP Business & Management

[12]. Apple Inc. (2023). Quarterly Report (Form 10-Q). Retrieved from. https://www.sec.gov/Archives/edgar/data/320193/000032019323000106/aapl-20230930.htm

[13]. Yahoo Finance. (2023). Apple Inc. (AAPL) Historical Prices. Retrieved June 14, 2023, from. https://finance.yahoo.com/quote/AAPL/history/

APPENDIX

A. Dataset
   Apple.csv
B. Code
   Click here