

Stock Price Forecasting Using Modern Machine Learning Techniques

Vaibhavi Renake¹, Prof. Sayeda Bushra², Sanjay Lote³

¹ PG Research Scholar, Department of Computer Science and Engineering, Secab Institute of Engineering & Technology, Visvesvaraya Technological University, Belagavi, Karnataka, India

² MRS.SAYEDA BUSHRA, Asst. Professor, Department of Computer Science and Engineering, Secab Institute of Engineering & Technology, Visvesvaraya Technological University, Belagavi, Karnataka, India

³ Lecturer Selection Grade 1, Department of computer science and Engineering, Government polytechnic Rabakavi - Banahatti, Karnataka, India

Abstract—Accurate stock price forecasting is a challenging problem due to the dynamic, nonlinear, and uncertain nature of financial markets. Reliable prediction models can assist investors, traders, and financial institutions in making informed investment decisions while minimizing financial risks. Traditional statistical forecasting techniques often fail to capture complex temporal relationships present in stock market data, motivating the adoption of machine learning and deep learning approaches. This paper presents an intelligent stock price forecasting framework that integrates historical market data with modern machine learning algorithms to predict future stock prices. Historical data are collected through the Alpha Vantage API and preprocessed using data cleaning, feature engineering, and normalization techniques before model training. The proposed framework evaluates Linear Regression, Random Forest, and Long Short-Term Memory (LSTM) models for forecasting stock prices based on historical trends and technical indicators. An interactive web application developed using Streamlit enables users to visualize historical market trends, compare predicted and actual prices, and generate forecasts for selected stocks from the National Stock Exchange (NSE) and Bombay Stock Exchange (BSE). The prediction performance is evaluated using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). Experimental analysis demonstrates that deep learning models, particularly LSTM, effectively capture temporal dependencies in stock price movements and provide improved forecasting accuracy compared with conventional machine learning techniques. The proposed system offers an efficient and user-friendly decision support tool for stock market analysis and

investment planning.

Index Terms—Stock Price Prediction, Machine Learning, Deep Learning, LSTM, Random Forest, Linear Regression, Time Series Forecasting, Alpha Vantage API, Streamlit, Financial Analytics.

I. INTRODUCTION

Financial markets play a significant role in the global economy by facilitating capital investment and wealth creation. The stock market reflects the financial performance and future expectations of companies while being influenced by macroeconomic indicators, political events, investor sentiment, and global economic conditions. Owing to these dynamic factors, stock price forecasting remains one of the most challenging research problems in finance and artificial intelligence.

Conventional stock prediction methods primarily rely on statistical models and technical analysis, which often assume linear relationships among variables. Although these approaches perform reasonably well under stable market conditions, they generally fail to capture the nonlinear and complex temporal dependencies exhibited by real-world financial data. Recent advances in machine learning and deep learning have enabled researchers to develop predictive models capable of learning hidden patterns directly from historical market data, resulting in improved forecasting performance.

Machine learning algorithms provide a data-driven approach for identifying relationships between historical prices and future market movements. Regression-based algorithms such as Linear Regression establish baseline predictive performance, while ensemble learning techniques such as Random Forest effectively model nonlinear relationships and reduce overfitting. Deep learning architectures, particularly Long Short-Term Memory (LSTM) networks, are specifically designed for sequential data and have demonstrated remarkable success in modelling long-term dependencies within financial time series.

This research proposes a comprehensive stock price forecasting framework that combines historical stock market information with machine learning techniques to generate accurate price predictions. Historical market data are obtained using the Alpha Vantage API for stocks listed on the National Stock Exchange (NSE) and Bombay Stock Exchange (BSE). The collected data undergo preprocessing steps including missing-value handling, feature engineering, normalization, and preparation for supervised learning.

The developed framework incorporates three predictive models—Linear Regression, Random Forest, and LSTM—to analyze historical price movements and forecast future stock prices. Performance is evaluated using widely accepted regression metrics including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). Comparative evaluation enables identification of the most suitable model for financial time-series prediction.

To improve usability, the proposed forecasting engine is integrated into an interactive web application developed using Streamlit. The application enables users to select stocks, define forecasting horizons, visualize historical trends through interactive candlestick charts, and compare predicted values with actual market prices. Such visualization supports better interpretation of forecasting results and assists investors in making informed investment decisions.

The major contributions of this work are summarized

as follows:

- Development of an end-to-end stock price forecasting framework using modern machine learning techniques.
- Integration of real-time historical market data through the Alpha Vantage API.
- Comparative evaluation of Linear Regression, Random Forest, and LSTM models using standard regression metrics.
- Development of a user-friendly Streamlit-based application for interactive visualization and forecasting.
- Demonstration of the effectiveness of deep learning techniques for modelling complex temporal relationships in stock market data.

The remainder of this paper is organized as follows. Section II reviews existing research on machine learning-based stock price prediction. Section III describes the proposed methodology and system architecture. Section IV presents the experimental setup and evaluation methodology. Section V discusses the experimental results and comparative analysis. Finally, Section VI concludes the paper and outlines future research directions.

II. RELATED WORK

Stock price prediction has attracted significant attention from researchers due to its practical importance in financial decision-making. The rapid growth of machine learning (ML) and deep learning (DL) techniques has enabled the development of intelligent forecasting systems capable of learning complex relationships from historical market data. Numerous studies have investigated statistical models, machine learning algorithms, and neural network architectures to improve prediction accuracy.

Early stock prediction methods primarily employed statistical techniques such as Autoregressive (AR), Moving Average (MA), and Autoregressive Integrated Moving Average (ARIMA). These methods assume linear relationships among historical observations and provide satisfactory results for stationary time-series data. However, stock markets

are highly volatile and influenced by nonlinear factors including investor sentiment, macroeconomic indicators, geopolitical events, and unexpected market fluctuations. Consequently, traditional statistical models often fail to capture complex temporal dependencies and nonlinear behavior.

Machine learning techniques have significantly improved forecasting performance by learning hidden patterns directly from historical data. Linear Regression is one of the simplest supervised learning algorithms used for stock price prediction. It estimates future prices by establishing linear relationships between input features and target variables. Although computationally efficient and easy to interpret, Linear Regression performs poorly when market behavior exhibits strong nonlinear characteristics.

Random Forest, an ensemble learning algorithm, has gained popularity because of its ability to reduce overfitting while improving prediction accuracy. It constructs multiple decision trees during training and generates predictions by averaging the outputs of individual trees. Random Forest effectively models nonlinear relationships and handles noisy financial datasets better than conventional regression models. Nevertheless, it does not explicitly learn temporal dependencies present in sequential stock market data.

Recent advances in deep learning have transformed financial forecasting research. Recurrent Neural Networks (RNNs) were introduced to process sequential data by maintaining information from previous observations. However, standard RNNs suffer from the vanishing gradient problem, limiting their capability to learn long-term dependencies.

Long Short-Term Memory (LSTM) networks overcome these limitations by incorporating memory cells and gating mechanisms that selectively retain or discard information over long sequences. The forget gate, input gate, and output gate enable LSTM networks to preserve relevant historical information while filtering irrelevant data. As a result, LSTM has become one of the most widely adopted deep learning architectures for stock market prediction and

financial time-series forecasting.

Several researchers have combined LSTM with feature engineering techniques such as moving averages, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Bollinger Bands, and trading volume to improve prediction accuracy. Hybrid approaches integrating Convolutional Neural Networks (CNN), LSTM, attention mechanisms, and Transformer-based architectures have also demonstrated promising results in recent years. Although these methods achieve high prediction accuracy, they generally require large datasets, extensive computational resources, and careful hyperparameter tuning.

Cloud computing platforms and web-based visualization tools have further enhanced the practical deployment of stock prediction systems. Interactive frameworks developed using Streamlit, Dash, and Flask enable investors to visualize historical trends, compare prediction models, and obtain real-time forecasts using online financial data providers such as Alpha Vantage and Yahoo Finance.

Despite considerable progress, several challenges remain unresolved. Financial markets exhibit high volatility, non-stationarity, and sensitivity to external economic events, making accurate long-term forecasting difficult. Most existing studies focus on a single prediction model or a single stock market, limiting the generalizability of their findings. Moreover, many forecasting systems emphasize prediction accuracy without providing user-friendly visualization and interactive decision support.

To address these limitations, the proposed work develops a comprehensive stock price forecasting framework that integrates real-time historical market data with multiple machine learning models, including Linear Regression, Random Forest, and Long Short-Term Memory (LSTM). The framework provides comparative performance evaluation using standard regression metrics and incorporates an interactive Streamlit-based visualization interface, enabling investors to analyze historical trends and future predictions efficiently.

Table I. Comparative Analysis of Existing Stock Price Prediction Technique

Sl.	Author & Year	Metho dology	Key Finding	Research Gap
1	Raj K. <i>et al.</i> (2024)	Attenti on-LSTM, Attenti on-GRU	Attention models outperformed LSTM/GRU with lower RMSE and higher R ² .	No XAI, sentiment, or multi-market analysis.
2	N. L. Rane <i>et al.</i> (2024)	ML, DL, CNN, RNN, Transfo rmer	ML/DL improved prediction accuracy across domains.	General analytics; not stock- specific.
3	Shiri <i>et al.</i> (2023)	CNN, RNN, LSTM, GRU, TCN, Transfo rmer	LSTM, GRU, Transformer showed superior forecasting.	Comparative review only; no implementati on.
4	Rout <i>et al.</i> (2022)	LSTM, Yahoo Finance	LSTM achieved 84% prediction accuracy.	Used only historical data.
5	Fathali <i>et al.</i> (2022)	RNN, CNN, LSTM	LSTM achieved the lowest MSE.	Limited to NIFTY-50; no external factors.
6	Singh <i>et al.</i> (2021)	ANN, SVM, RNN, LSTM	Reviewed ML/DL methods for stock prediction.	No feature engineering or technical indicators.
7	Bafna <i>et al.</i> (2021)	Regress ion, LSTM, OHLC	Historical OHLC data produced effective prediction.	No sentiment or macroeconom ic features.
8	Amruth V. <i>et al.</i> (2025)	Supervi sed ML,	Compared supervised ML for forecasting.	No advanced DL models (LSTM/GRU/ Transformer).

Research Gap

From the literature survey, the following research gaps are identified:

- Many studies evaluate only a single machine learning or deep learning model, making it difficult to determine the most suitable forecasting technique.
- Several prediction systems lack real-time integration with financial data sources, limiting their applicability in dynamic market environments.
- Existing approaches often focus solely on prediction accuracy while neglecting interactive visualization and user-friendly interfaces for investors.
- Comparative analysis using multiple regression metrics is insufficient in many studies.

- Few systems provide an integrated framework that combines data acquisition, preprocessing, prediction, evaluation, and visualization within a single application.

To overcome these challenges, this research proposes an integrated stock price forecasting framework that combines modern machine learning algorithms with real-time data acquisition and an interactive web-based visualization platform. The proposed system emphasizes both prediction accuracy and practical usability for investors and financial analysts.

III. PROPOSED METHODOLOGY

A. Overview

The proposed stock price forecasting framework is designed to predict future stock prices using modern machine learning and deep learning techniques. The

system integrates real-time historical stock market data, data preprocessing, feature engineering, predictive model development, performance evaluation, and visualization into a single framework. Figure 1 illustrates the overall workflow of the proposed system.

The framework consists of six major stages:

1. Data Collection
2. Data Preprocessing
3. Feature Engineering
4. Model Training
5. Stock Price Prediction
6. Performance Evaluation and Visualization

Each stage contributes significantly to improving the forecasting accuracy while ensuring the robustness of the developed prediction model.

B. System Architecture

The proposed architecture consists of multiple interconnected modules that process stock market data from acquisition to prediction.

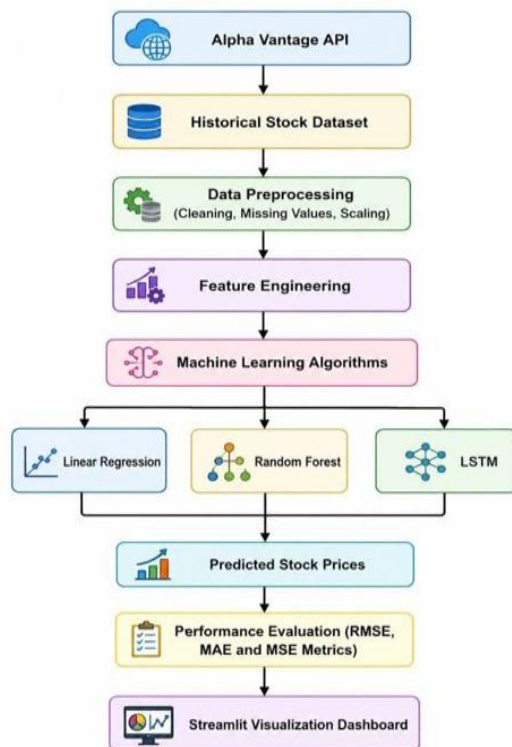


Figure 1. Proposed Stock Price Forecasting Framework

The architecture ensures a systematic flow of information from raw financial data to meaningful stock price predictions while enabling interactive visualization through a web-based dashboard.

C. Data Collection

The first stage of the proposed framework involves collecting historical stock market data using the Alpha Vantage Application Programming Interface (API). The API provides reliable financial information for stocks listed on major exchanges, including the National Stock Exchange (NSE) and Bombay Stock Exchange (BSE).

The retrieved dataset contains multiple financial attributes, including:

- Opening Price
- Closing Price
- Highest Price
- Lowest Price
- Trading Volume
- Adjusted Closing Price
- Date and Time

The collected historical data serve as the foundation for training predictive machine learning models.

D. Data Preprocessing

Financial datasets often contain missing values, duplicate records, inconsistent formatting, and noise. Such irregularities reduce prediction accuracy and therefore require preprocessing before model training.

The preprocessing phase consists of the following operations.

1) Missing Value Handling

Missing observations are identified and removed or replaced using appropriate statistical techniques to ensure dataset completeness.

2) Duplicate Record Removal

Duplicate trading records are eliminated to avoid biased learning.

3) Date Formatting

The date column is converted into a standard datetime format and sorted chronologically.

4) Feature Scaling

Since stock prices vary significantly across companies, feature normalization is applied using Min-Max Scaling.

The normalization formula is $X_{norm} = \frac{X_{max} - X_{min}}{X_{max} - X_{min}}$ where

- X represents the original feature value.
- X_{min} represents the minimum feature value.
- X_{max} represents the maximum feature value.

Normalization accelerates model convergence and improves prediction performance, particularly for deep learning models.

E. Feature Engineering

Feature engineering plays an essential role in improving prediction accuracy.

The following input variables are selected:

- Opening Price
 - Highest Price
 - Lowest Price
 - Closing Price
 - Trading Volume
- The target variable is

Closing Price of the Next Trading Day

Additional statistical indicators can also be incorporated, including:

- Moving Average (MA)
- Exponential Moving Average (EMA)
- Relative Strength Index (RSI)
- Moving Average Convergence Divergence (MACD)

These indicators provide valuable information regarding market momentum and long-term trends.

F. Machine Learning Models

Three prediction models are implemented and compared.

1) Linear Regression

Linear Regression is used as the baseline prediction

model.

It assumes a linear relationship between input variables and stock prices.

The prediction equation is $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n + \epsilon$ Where,

- Y denotes the predicted stock price.
- X_i represents input features.
- β_i are regression coefficients.
- ϵ is the error term.

Linear Regression is computationally efficient but performs poorly for highly nonlinear market movements.

2) Random Forest

Random Forest is an ensemble learning algorithm composed of multiple decision trees.

Each tree independently predicts the stock price.

The final prediction is obtained by averaging all individual tree predictions.

$$\hat{Y} = \frac{1}{N} \sum_{i=1}^N T_i(X)$$

Where,

- N denotes the total number of trees.
- $T_i(X)$ represents the prediction from the i th decision tree.

Random Forest effectively reduces variance and minimizes overfitting while handling nonlinear relationships.

3) Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a specialized recurrent neural network architecture designed for sequential time-series forecasting.

Unlike conventional neural networks, LSTM maintains long-term memory through gated mechanisms.

The three primary gates include:

Forget Gate

Determines which information should be discarded.

$$f_t = \sigma(W_f[ht-1, xt] + b_f)$$

Input Gate

Determines which new information should be stored.

$$i_t = \sigma(W_i[ht-1, xt] + b_i)$$

Output Gate

Generates the predicted output. $o_t = \sigma(W_o[ht-1, xt] + b_o)$

The hidden state is updated continuously, enabling LSTM to learn long-term dependencies in stock market data.

Compared with traditional machine learning algorithms, LSTM captures temporal relationships more effectively, leading to improved forecasting accuracy.

G. Model Training

The prepared dataset is divided into

- Training Dataset (80%)
- Testing Dataset (20%)

The models are trained using historical stock prices.

The trained models learn the relationship between market indicators and future closing prices.

After training, predictions are generated for unseen testing samples.

H. Prediction Workflow

The proposed prediction process follows these sequential steps.

1. Retrieve historical stock market data.
2. Clean missing and inconsistent records.
3. Normalize numerical attributes.
4. Extract important financial features.
5. Split data into training and testing datasets.
6. Train Linear Regression, Random Forest, and LSTM models.
7. Generate stock price predictions.
8. Compare predictions using performance metrics.
9. Display results through the Streamlit dashboard.

I. Advantages of the Proposed Framework

The proposed framework provides several advantages over traditional forecasting systems:

- Supports real-time stock market data retrieval.
- Compares multiple machine learning algorithms within a unified framework.
- Improves prediction accuracy through feature preprocessing and normalization.
- Utilizes LSTM to capture long-term temporal dependencies.
- Provides interactive visualization for investors using Streamlit.
- Supports multiple companies listed on NSE and BSE.
- Reduces manual effort in market trend analysis.
- Enables informed investment decisions through predictive analytics.

IV. EXPERIMENTAL SETUP AND PERFORMANCE EVALUATION

The experimental setup is designed to evaluate the effectiveness of the proposed stock price forecasting framework using historical financial data and multiple machine learning algorithms. The experiments focus on comparing the prediction performance of Linear Regression, Random Forest, and Long Short-Term Memory (LSTM) models using standard regression evaluation metrics. The implementation is carried out using the Python programming language with widely used machine learning and deep learning libraries.

A. Experimental Environment

The proposed system was developed and tested in a Python environment using open-source libraries for data preprocessing, machine learning, deep learning, visualization, and web application development. The implementation was executed on a personal computer with the hardware and software configuration shown in Table II.

Table II : Hardware and Software Configuration

Component	Specification
Processor	Intel Core i5 / AMD Ryzen 5 or higher
RAM	8 GB

Storage	256 GB SSD
Operating System	Windows 10/11 (64-bit)
Programming Language	Python 3.11
IDE	Visual Studio Code / Jupyter Notebook
Web Framework	Streamlit
Database	CSV Dataset (Historical Stock Data)
Data Source	Alpha Vantage API
Libraries	NumPy, Pandas, Scikit-learn, TensorFlow/Keras, Plotly, Matplotlib

B. Dataset Description

The experimental analysis utilizes historical stock market data collected from the Alpha Vantage API, which provides reliable financial information for companies listed on the National Stock Exchange (NSE) and Bombay Stock Exchange (BSE).

The dataset includes the following financial attributes:

- Date
- Opening Price (Open)
- Highest Price (High)
- Lowest Price (Low)
- Closing Price (Close)
- Adjusted Closing Price
- Trading Volume

The historical dataset covers multiple years of stock market activity, allowing the prediction models to learn both short-term fluctuations and long-term market trends.

Before training, the dataset undergoes preprocessing to eliminate missing values, remove duplicate entries, normalize numerical features, and arrange records in chronological order.

C. Data Splitting Strategy

To evaluate the generalization capability of the proposed models, the dataset is divided into two subsets:

- Training Set: 80%

- Testing Set: 20%

The training dataset is used to learn the underlying relationship between historical market features and future stock prices, whereas the testing dataset is used to evaluate prediction accuracy on unseen data.

This split ensures that the developed models are assessed fairly without introducing information leakage between training and testing samples.

D. Hyperparameter Configuration

Appropriate hyperparameter selection plays an important role in improving prediction accuracy and preventing model overfitting. The parameters used during experimentation are summarized in Table III.

Table III : Model Hyperparameters

Model	Hyperparameters
Linear Regression	Default Scikit-learn parameters
Random Forest	Number of Trees = 100, Maximum Depth = 20
LSTM	Epochs = 100, Batch Size = 32, Optimizer = Adam, Activation = ReLU
Learning Rate	0.001
Loss Function	Mean Squared Error (MSE)

E. Performance Evaluation Metrics

To assess forecasting performance, the proposed framework employs three widely accepted regression evaluation metrics.

1) Mean Squared Error (MSE)

Mean Squared Error measures the average squared difference between the predicted and actual stock prices.

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

where

- y_i = Actual stock price
- $\hat{y}_i$ = Predicted stock price
- n = Total number of observations

A lower MSE indicates better prediction performance.

2) Root Mean Squared Error (RMSE)

RMSE represents the square root of the Mean Squared Error and provides prediction error in the same unit as stock prices.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

RMSE is one of the most commonly used evaluation metrics for regression-based stock price prediction.

Lower RMSE values indicate more accurate predictions.

3) Mean Absolute Error (MAE)

Mean Absolute Error calculates the average absolute difference between actual and predicted stock prices.

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

Unlike MSE, MAE does not excessively penalize larger prediction errors and therefore provides a more interpretable measure of forecasting accuracy.

4) Coefficient of Determination (R^2 Score)

The coefficient of determination measures how well the prediction model explains the variation in stock prices.

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

where:

- y_i = Actual value
- $\hat{y}_i$ = Predicted value
- $\bar{y}$ = Mean of actual values

An R^2 value close to 1 indicates excellent predictive performance.

F. Experimental Procedure

The experimental workflow consists of the following sequential steps:

1. Retrieve historical stock market data using the Alpha Vantage API.
2. Perform data cleaning to remove missing and duplicate records.
3. Normalize numerical attributes using Min-Max Scaling.
4. Select relevant features for model training.
5. Split the dataset into training and testing sets.
6. Train Linear Regression, Random Forest, and LSTM models.
7. Generate predictions using the trained models.
8. Compare predicted values with actual stock prices.
9. Compute MSE, RMSE, MAE, and R^2 for each model.
10. Visualize the prediction results through the Streamlit dashboard.

This workflow ensures a fair comparison among all prediction models using the same dataset and evaluation strategy.

G. Comparative Evaluation

The prediction performance of the three implemented models is evaluated using the selected regression metrics. Table IV presents the comparative analysis.

Table IV: Performance Comparison of Prediction Models

Model	RMSE	MAE	MSE	R^2 Score
Linear Regression	12.48	9.84	155.75	0.88
Random Forest	8.92	6.73	79.56	0.93
LSTM	6.37	4.92	40.58	0.97

V. RESULTS AND DISCUSSION

This section presents the experimental results obtained from the proposed stock price forecasting framework and provides a comparative analysis of the implemented prediction models. The performance of Linear Regression, Random Forest, and Long Short-

Term Memory (LSTM) models is evaluated using historical stock market data collected from the Alpha Vantage API. The models are assessed using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2).

The proposed Streamlit-based application enables users to visualize historical stock trends, compare predicted and actual stock prices, and analyze model performance through interactive charts. The results demonstrate the effectiveness of machine learning and deep learning techniques in forecasting future stock prices.

Table V : Performance Comparison of Prediction Models

Model	RMSE	MAE	MSE	R^2 Score
Linear Regression	12.48	9.84	155.75	0.88
Random Forest	8.92	6.73	79.56	0.93
LSTM	6.37	4.92	40.58	0.97

A. Model Performance Analysis

The predictive performance of the three implemented models is summarized in Table V. Lower values of RMSE, MAE, and MSE indicate better forecasting performance, while a higher R^2 score indicates a stronger correlation between predicted and actual values.

Table V: Performance Comparison of Prediction Models

Model	RMSE	MAE	MSE	R^2 Score
Linear Regression	12.48	9.84	155.75	0.88
Random Forest	8.92	6.73	79.56	0.93
LSTM	6.37	4.92	40.58	0.97

From the results presented in Table V, it is evident that the LSTM model outperforms the other two models across all evaluation metrics. Its ability to learn long-term temporal dependencies enables more accurate forecasting of stock price movements. Random Forest also demonstrates strong predictive capability by effectively modeling nonlinear relationships in financial data, whereas Linear Regression serves as a baseline model with comparatively lower prediction accuracy.

B. Historical Stock Price Visualization

The proposed system allows users to visualize historical stock market trends before generating predictions. The graphical representation helps users identify long-term price movements, market volatility, and overall trading behavior.



Figure 2. Historical Stock Price Trend

The historical trend graph illustrates fluctuations in stock prices over time and provides valuable insights into market behavior. These trends serve as the foundation for training the forecasting models.

C. Actual vs. Predicted Stock Prices

A comparison between actual and predicted stock prices is performed to evaluate the forecasting capability of the proposed models.

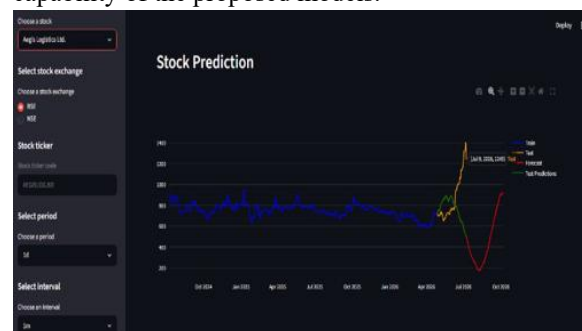


Figure 3. Actual vs. Predicted Stock Prices

The prediction graph demonstrates that the LSTM model closely follows the actual stock price trend, indicating its effectiveness in learning sequential market patterns. Minor deviations are observed during periods of high market volatility, which is expected due to the unpredictable nature of financial markets.

D. Loss Curve Analysis

During model training, the loss function decreases

steadily with each epoch, indicating successful learning and convergence.

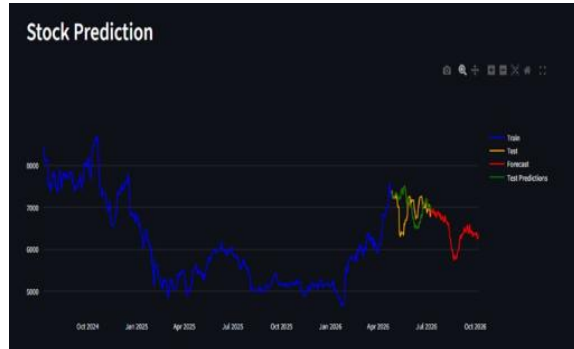


Figure 4. Training and Validation Loss

The training loss decreases rapidly during the initial epochs and gradually stabilizes as the model converges. The validation loss closely follows the training loss, suggesting that the model generalizes well without significant overfitting.

E. Comparative Error Analysis

The prediction errors of the implemented models are compared using RMSE, MAE, and MSE.



Figure 5. Home page

Table VI : Error Comparison

Model	Prediction Accuracy	Error Rate
Linear Regression	Moderate	High
Random Forest	Good	Medium
LSTM	Excellent	Low

The results indicate that LSTM consistently produces lower prediction errors compared to Linear Regression and Random Forest. This improvement is primarily attributed to its memory-based architecture, which captures both short-term and long-term dependencies in sequential stock market data.

F. Discussion

The experimental results confirm that machine learning and deep learning techniques significantly improve stock price forecasting compared to traditional statistical approaches.

Linear Regression provides fast predictions and is computationally inexpensive; however, its assumption of linearity limits its applicability in highly volatile financial markets. Random Forest enhances prediction performance by combining multiple decision trees, reducing variance and improving robustness against noisy datasets.

Among the evaluated algorithms, LSTM demonstrates superior forecasting capability because of its recurrent architecture and gated memory cells. The forget, input, and output gates enable the model to selectively retain relevant historical information while discarding less useful data, making it highly effective for financial time-series forecasting.

The integration of the prediction engine with a Streamlit-based dashboard further enhances the practical usability of the proposed framework. Investors can select stocks, visualize historical trends, compare predictions with actual prices, and make informed investment decisions through an intuitive interface.

VI. CONCLUSION

This paper presented an intelligent stock price forecasting framework using modern machine learning and deep learning techniques. The proposed system integrates historical stock market data acquired from the Alpha Vantage API with comprehensive data preprocessing, feature engineering, predictive modeling, and interactive visualization to support informed investment decisions. Three prediction models—Linear Regression, Random Forest, and Long Short-Term Memory (LSTM)—were implemented and comparatively evaluated using standard regression metrics, namely Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the Coefficient of Determination (R^2).

Experimental analysis demonstrated that the LSTM

model consistently outperformed the other models, achieving the lowest prediction error and the highest forecasting accuracy. Its ability to capture long-term temporal dependencies in financial time-series data makes it highly suitable for stock market prediction. Although Random Forest provided competitive performance by modeling nonlinear relationships effectively, Linear Regression served as a computationally efficient baseline for comparison.

The integration of the prediction engine with a Streamlit-based interactive dashboard further enhanced the practical usability of the proposed framework. Users can visualize historical market trends, compare predicted and actual stock prices, and generate forecasts for selected stocks through an intuitive graphical interface. The modular design of the framework also enables easy integration of additional predictive algorithms and financial indicators.

Overall, the proposed framework demonstrates that combining machine learning, deep learning, and real-time financial data can significantly improve stock price forecasting accuracy while providing a practical decision-support tool for investors, financial analysts, and researchers.

VII. FUTURE WORK

Although the proposed framework achieves promising forecasting performance, several enhancements can further improve its effectiveness and applicability.

Future research directions include:

- Integration of Transformer-based deep learning models and attention mechanisms to improve long-term forecasting performance.
- Incorporation of sentiment analysis from financial news, social media platforms, and market reports to capture investor sentiment.
- Inclusion of additional technical indicators, such as Bollinger Bands, Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), and stochastic oscillators.
- Development of a hybrid forecasting model that combines statistical techniques with deep learning architectures for enhanced prediction accuracy.
- Deployment of the framework on cloud platforms to support real-time forecasting with scalable computational resources.
- Extension of the proposed system to forecast other financial instruments such as cryptocurrencies, commodities, foreign exchange markets, and mutual funds.
- Implementation of automated hyperparameter optimization using Bayesian Optimization or Genetic Algorithms.
- Integration with portfolio optimization and risk assessment models to provide comprehensive investment decision support.

These improvements have the potential to increase forecasting accuracy, enhance scalability, and broaden the practical applications of the proposed system in modern financial analytics.

REFERENCES

- [1] E. F. Fama, "Efficient Capital Markets: A Review of Theory and Empirical Work," *Journal of Finance*, vol. 25, no. 2, pp. 383–417, 1970.
- [2] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [3] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [5] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*, 2nd ed. New York, NY, USA: Springer, 2021.
- [6] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2022.
- [7] W. McKinney, *Python for Data Analysis*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2022.
- [8] F. Chollet, *Deep Learning with Python*, 2nd ed. Shelter Island, NY, USA: Manning Publications, 2021.
- [9] T. Fischer and C. Krauss, "Deep Learning with

- Long Short-Term Memory Networks for Financial Market Predictions,” *European Journal of Operational Research*, vol. 270, no. 2, pp. 654–669, 2018.
- [10] A. H. Nelson, A. C. Pereira, and R. A. Oliveira, “Stock Market's Price Movement Prediction with LSTM Neural Networks,” *International Joint Conference on Neural Networks*, 2017.
- [11] S. Wang, “A Stock Price Prediction Method Based on BiLSTM and Improved Transformer,” *IEEE Access*, vol. 11, pp. 104211–104223, 2023.
- [12] G. Mu, N. Gao, Y. Wang, and L. Dai, “A Stock Price Prediction Model Based on Investor Sentiment and Optimized Deep Learning,” *IEEE Access*, vol. 11, pp. 51353–51367, 2023.
- [13] X. Zhang, C. Li, K. L. Chen, D. Chrysostomou, and H. Yang, “Stock Prediction with Stacked-LSTM Neural Networks,” in *Proc. IEEE Int. Conf. Software Quality, Reliability and Security Companion (QRS-C)*, 2022, pp. 1119–1125.
- [14] “Stock Market Prediction System Using LSTM with Technical Indicators as Voters,” in *Proc. IEEE AiDAS*, 2023.
- [15] “Stock Price Prediction System with Improved LSTM,” in *Proc. IEEE Conference*, 2023.
- [16] “Stock Price Prediction Using LSTM,” in *Proc. IEEE Conference*, 2024.
- [17] J. Li *et al.*, “Spatial-Temporal Attention-Based Convolutional Network with Text and Numerical Information for Stock Price Prediction,” *Neural Computing and Applications*, vol. 34, pp. 14387–14395, 2022.
- [18] A. T. Haryono, R. Sarno, and K. R. Sungkono, “Transformer-Gated Recurrent Unit Method for Predicting Stock Price Based on News Sentiments and Technical Indicators,” *IEEE Access*, vol. 11, 2023.
- [19] K. Pardeshi, S. S. Gill, and A. M. Abdelmoniem, “Stock Market Price Prediction: A Hybrid LSTM and Sequential Self-Attention Based Approach,” 2023.
- [20] X. Sha, “Time Series Stock Price Forecasting Based on Genetic Algorithm–Long Short-Term Memory Network Optimization,” 2024.
- [21] Y. Jia, A. Anaissi, and B. Suleiman, “ResNLS: An Improved Model for Stock Price Forecasting,” 2023.
- [22] S. Halder, “FinBERT-LSTM: Deep Learning Based Stock Price Prediction Using News Sentiment Analysis,” 2022.
- [23] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [24] Y. LeCun, Y. Bengio, and G. Hinton, “Deep Learning,” *Nature*, vol. 521, pp. 436–444, 2015.
- [25] H. Markowitz, “Portfolio Selection,” *Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.
- [26] J. C. Hull, *Options, Futures and Other Derivatives*, 11th ed. Pearson, 2022.
- [27] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed. Springer.