Journal of Research High School

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

Predicting Stock Market Movements: A Comparative Analysis of Machine Learning Models for Investment Strategies

Shaunak Thamke[1] *

[1] The Bronx High School of Science, Bronx, NY, USA
*Corresponding Author: Shaunakthamke@gmail.com

Advisor: Odysseas Drosis, od84@cornell.edu

Received January 26, 2025; Revised July 28, 2025; Accepted August 11, 2025

Abstract

This research paper addressed the critical challenges in stock market prediction that arise from the inherent volatility and complexity of financial markets. In the short term, stock prices are driven by a complex interplay of factors, ranging from supply and demand, market sentiment, news events, and economic indicators, rather than solely by a company's fundamental performance. The primary research question guiding the study was: "How can Artificial Intelligence (AI) driven models improve the accuracy of short-term stock market predictions for practical investment purposes?" To explore this, the study evaluated the effectiveness of four AI-driven models including Linear Regression, Decision Trees, Neural Networks, and Random Forest. The objective was to enhance investment strategy formulation and contribute to the broader field of quantitative finance. The methodology involved training these models on a five-year dataset from Dec 2019-2024 utilizing key libraries such as YahooFinance (YFinance), numpy, and Sci-Kit Learn and assessing their accuracy using Mean Squared Error (MSE) supplemented with Confidence Interval (CI) for prediction precision. A trading simulation was incorporated to analyze potential financial returns based on model predictions. For example, the simulation predicted that investing $20,000 in Google would yield a profit of $47,729 in five years. This research aims to provide a foundational, accessible tool for individuals with limited financial literacy empowering them to make informed, profit-generating investment decisions.

Keywords: Artificial Intelligence, Machine Learning, Linear Regression, Decision Trees, Neural Networks, Random Forest, Confidence Interval, Stock Market Prediction, Simulation

1. Introduction

The stock market is a vital part of the global economy which enables people to buy and sell company shares. With about 62% of Americans owning stocks (Gallup, 2024), market performance significantly impacts household savings. Despite its importance, the market is hard to predict due to high volatility and many influencing factors, such as earnings reports, trends, and news. Prices fluctuate based on supply and demand, and traditional methods often struggle to capture the complex, non-linear patterns in the data.

AI has recently become a popular tool for solving complex problems, and this study explores its potential in short-term stock market prediction. AI enables machines to mimic human intelligence by learning from data to recognize patterns and making informed decisions. This makes it useful for forecasting in fields like finance, healthcare, and weather. AI models use historical data to identify patterns and generate forecasts. Common models include Linear Regression, Decision Trees, Random Forests, Neural Networks, Long Short-Term Memory (LSTM) networks, support vector machines (SVMs), and recurrent neural networks (RNNs). These models are trained by feeding them past stock prices (and sometimes other indicators) so they can minimize error functions such as MSE. Once trained, they produce output much faster than a human could calculate by hand. However, just applying AI is not enough; it is

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

important to understand the existing body of research that has already been conducted, where it falls short, and how a new study can improve on it.

Previous studies show both the promise and the limits of AI for stock prediction. For example, Zheng et al. (n.d.) developed a model that reached about 70% accuracy. While this result sounds strong, the study mainly reports an accuracy number and does not explain which specific market conditions or external events might cause the remaining 30% of errors. Without examining those causes, it is hard for investors to trust the model during real-world shocks (like sudden news or economic changes). Patel et al. (2021) surveyed many AI papers and concluded there is significant room for further work. Their review shows growth in the field, but it does not offer a side-by-side evaluation of how different algorithms perform under the same data conditions. As a result, readers still do not know which model families are most stable or how they compare when measured consistently. Singh (2022) found that, on Nifty 50 index data from 1996–2021, Linear Regression and Artificial Neural Networks (ANN) performed similarly, while Random Forest and Decision Trees underperformed as the dataset size grew. In this research paper, the aim was to validate whether these models hold similar patterns for U.S. market data. Tupe et al. (2021) combined several algorithms such as Artificial Neural Network (ANN), Random Forest, Support Vector Machine (SVM), and LSTM to boost predictive accuracy. Their approach is valuable, yet their paper focuses on prediction metrics only. It does not translate those predictions into practical investment outcomes like expected dollar profit or loss. Across these studies, a common limitation is the lack of a clear bridge between model accuracy and actionable financial decisions.

Models like linear regression, decision trees, random forests, and neural networks are particularly relevant for short-term prediction because they can quickly capture current trends, patterns, and relationships in historical price data. Their ability to process real-time or high-frequency data makes them suitable for forecasting short-term price movements. In contrast, SVMs can struggle with large, noisy financial datasets, while RNNs often require extensive data and training time, making them less practical for fast, short-term predictions. LSTM models require large data and high computation, making them less ideal for short-term or real-time predictions.

As a result, this research aims to fill that gap by evaluating four accessible AI models - Linear Regression, Decision Trees, Neural Networks, and Random Forest. Instead of reporting accuracy alone, the study adds a simple trading simulation that converts predictions into estimated dollar returns. By doing so, it directly answers the question most investors care about: “If I invested today, how much might I gain after a certain period?” Another improvement is that the models are compared using the same dataset and evaluation metric (MSE) complemented by confidence interval analysis, giving a fair test of their strengths and weaknesses. When a model performs poorly, the analysis considers possible reasons, such as overfitting, lack of external features, or sensitivity to sudden price jumps.

The central research question is: How can AI-driven models improve the accuracy and usefulness of short-term stock market predictions for practical investment decisions? By connecting predictive performance to simulated profit, this study provides clearer guidance for everyday investors who may not be financially literate but want understandable tools. Overall, the goal is not only to predict future prices but also to show how those predictions can support smarter, more informed investment choices.

2. Materials and Methods

For this project, the SciKit-Learn library and five years of historical data from Yahoo Finance covering Google, General Motors, Microsoft, and J.P. Morgan were used to build the models. The Scikit-learn Library contained valuable information for creating a linear regression, neural network, decision tree, and random forest model. The historical data was split into a ratio of 67% to 33% for training and testing purposes. The split was found to be prudent because it provides enough data for the model to learn patterns (training set) while reserving a substantial portion to evaluate performance on unseen data (testing set), helping to avoid overfitting and assess generalization. Moreover, YFinance data was already normalized for splits/dividends etc.

A linear regression model is conceptually straightforward, easy to develop and computationally efficient. It assumes a linear relationship between the dependent and independent variables offering clarity and interpretability; hence it is not a black-box model. Its simplicity enables not only prediction but also powerful feature extraction. This

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

approach demands that all features are on the same scale, which is true in this case, as the prices of the same stocks remain similar and comparable day-to-day.

A linear regression model was used to predict the stock prices based on historical ticker data from the past five years. A linear regression model captures the linear dependence of data through a model as shown in Figure 1. This model is not likely to overfit the data. The model is also comparably simple to interpret. The generic equation for the linear model is shown as follows where y is a dependent variable dependent on independent variables $x_1, x_2, \ldots, x_n$ with their respective weights w_1, w_2,....,w_n and b is the constant:

$$y = w_1 * x_1 + w_2 * x_2 + \cdots + w_n * x_n + b$$

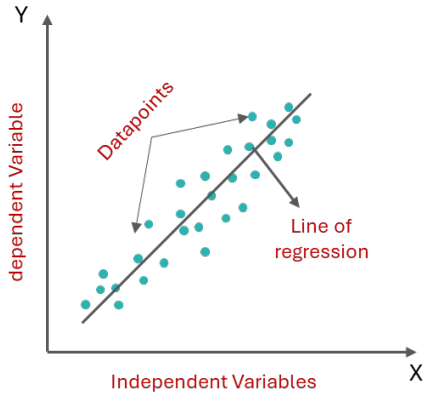

Figure 1. Linear regression graph.

In this study the aim was to predict the next-day price using the previous three days' prices; accordingly, the model is

$$p_t = w_1 * p_{t-1} + w_2 * p_{t-2} + w_3 * p_{t-3} + b$$

where p_t is the predicted price (dependent variable) from the last three prices $p_{t-1}, p_{t-2}, p_{t-3}$ (independent variables) with their respective weights w_1, w_2, w_3, and b is the constant.

A neural network was employed as an alternative model to predict stock prices from historical data. Among the models considered, it is the most complex and computationally intensive, often functioning as a "black box" due to its lack of interpretability. Designed to mimic the human brain, it consists of multiple interconnected hidden layers of artificial neurons, as illustrated in Figure 2.

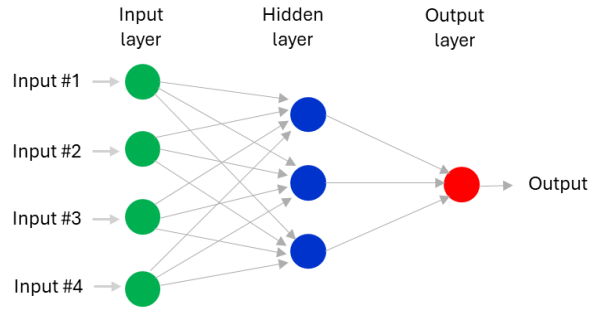

Figure 2. Architecture of the neural network model used for stock price prediction.

Neural networks are effective at capturing non-linear relationships that simpler models may miss. However, their complexity increases the risk of overfitting, especially as the number of parameters grows. Identifying the optimal network architecture, such as the number of hidden layers, is non-trivial and was determined through experimentation. While deeper networks can enhance predictive power, they also increase training time and reduce transparency making them less ideal for high-frequency trading. For example, a model may overfit by extrapolating a continuous rise in stock prices without accounting for potential downturns.

Another way to predict stock prices based on historical data is to use a decision tree model. A decision tree model operates like a flowchart, making predictions through a series of if-then-else decision rules as illustrated in Figure 3. A decision tree model is generally more interpretable than neural networks and even linear regression, as its hierarchical structure can be visualized and understood easily. It is less computationally demanding than neural networks but can become complex and prone to overfitting with deep trees. Compared to linear regression, decision trees handle non-linear relationships better but may lack the simplicity and efficiency of linear models. For this project, the

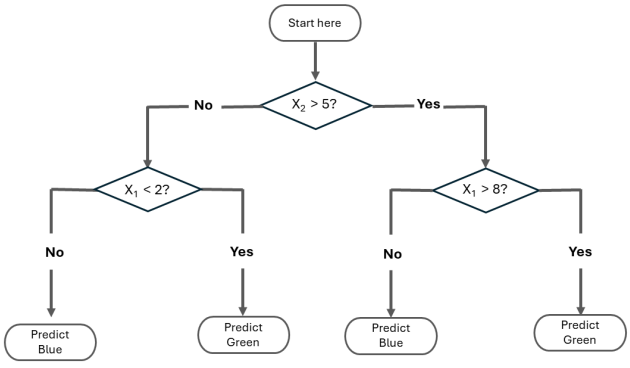

Figure 3. Decision tree model

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

optimal maximum depth of the decision tree was set to 8, limiting the tree to eight levels to prevent overfitting and improve computational efficiency. By following these decision paths, the model arrives at a final output.

The final model used to predict future stock prices was the Random Forest, an ensemble method that combines multiple decision trees to improve prediction accuracy and robustness. Unlike a single decision tree, Random Forest reduces overfitting by averaging the outputs of multiple trees, as illustrated in Figure 4. It also provides feature importance rankings, offering insights into which variables most influence predictions.

Although more complex and less interpretable than linear regression or a single decision tree, Random Forest generally achieves higher accuracy, particularly with non-linear data. It is also more computationally efficient and interpretable than neural networks.

The model's performance depends largely on two key hyperparameters: max depth and number of trees. Greater depth can capture complex patterns but increases the risk of overfitting, while too few trees may reduce accuracy and too many may add unnecessary computational cost. To strike a balance between accuracy and efficiency, several configurations were tested. The optimal max depth for this project was determined to be five.

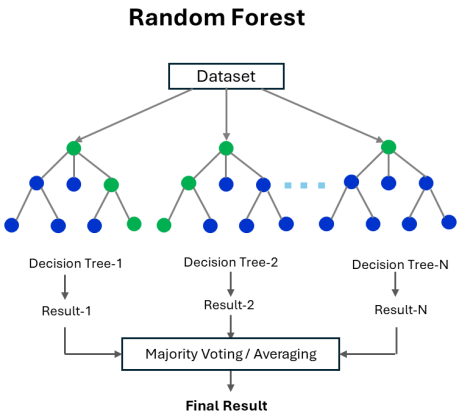

Figure 4. Random Forest model.

For model evaluation, the dataset was divided into training and testing subsets. This split allowed the model to learn patterns from the training data and then be assessed on its performance using the testing data. The purpose of this approach is to determine how well the model performs on the data it has not seen, ensuring its effectiveness beyond the training set.

MSE was used to test all models. While absolute error could have been an option, MSE was chosen because it minimizes the impact of occasional large errors, resulting in a more balanced or reasonably good evaluation even during bad days. MSE measures a model's performance by calculating the squared differences between predicted and actual values. These squared errors are then averaged to give the final MSE value. A lower MSE indicates a better-performing model.

The formula for MSE used in this research is shown below.

$$\frac{1}{n}\sum_{1}^{n}(Y_i - \hat{Y}_i)^2$$

where
Y_i is the real price
$\hat{Y}_i$ is the predicted price
n is the number of samples

Confidence Interval (CI) was used to find if the differences between the models are significant or not. CI shows the range where the true value is likely to fall, based on sample data. Wider CIs indicate more uncertainty, narrower ones more precision. Overlapping CIs suggest no significant difference, while non-overlapping ones imply a likely difference.

$$CI = K * \frac{std(scores)}{\sqrt{n}}$$

where
K is the Number for confidence (1.96 for 95% confidence)
$std(scores)$ is the standard deviation of MSE scores
n is the number of MSE scores

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

Linear Regression, being the best-performing model based on its lowest MSE, was used to develop a trading simulation to assess stock profitability. The model forecasts daily stock prices and informs trading decisions based on user input for initial capital and stock holdings. If a price increase is predicted and funds are available, one stock is bought; if a decrease is expected and stocks are held, one is sold. Otherwise, the portfolio remains unchanged. The simulation estimates the final portfolio value over a specified period, indicating potential profitability.

3. Results

The analysis evaluated the MSE for both training and testing datasets across Linear Regression, Neural Network, Decision Tree (with a maximum depth of 8), and Random Forest (with a maximum depth of 5) models applied to four companies including Google (GOOG), General Motors (GM), Microsoft (MSFT), and J.P. Morgan (JPM). It should be noted that while stocks from diverse industries including Technology, Banking and Automobiles were considered, the models can be tested on any stocks from any industries traded in the US.

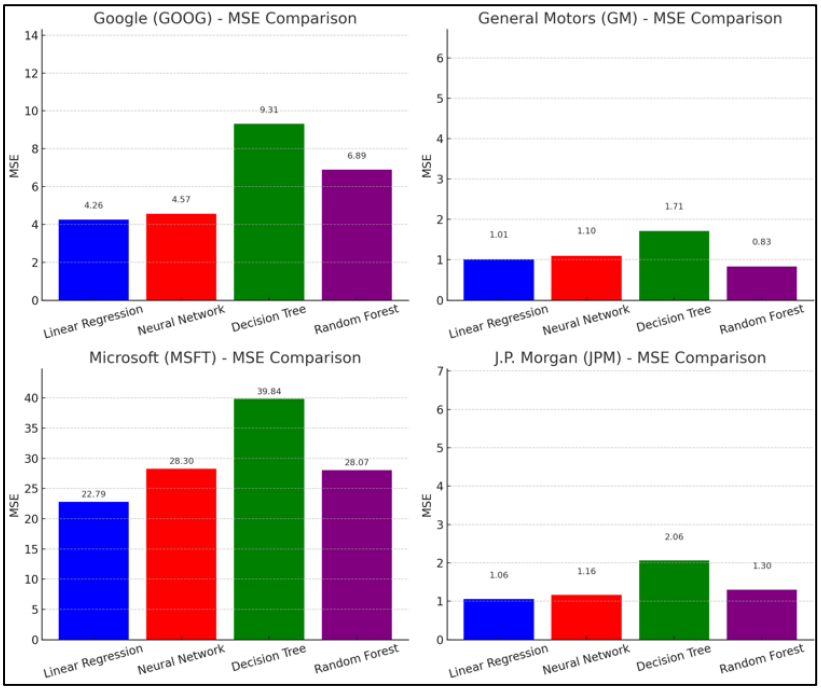

Figure 5. MSE Comparison of training data.

As shown in Figure 5, Linear Regression achieved the lowest MSE on the testing data for all stocks except GM, where Random Forest performed slightly better. This indicates its strong generalization capability and resistance to overfitting. Conversely, the decision tree exhibited the highest MSE on testing data, likely due to overfitting the training data and failing to generalize well to data not seen before. Random Forest outperformed in one case (GM) and could be a good back-up choice, particularly when dealing with less linear patterns. Neural Networks, while better than Decision Trees, are not as robust as Linear Regression or Random Forest in this dataset. GM and JPM have relatively low MSEs across all models, suggesting easier predictability, whereas MSFT and GOOG exhibit higher MSEs, which may imply greater price volatility.

From the results, it was concluded that the linear model consistently gave the best performance with the best (lowest MSE) for all stocks except General Motors (GM), where Random Forest performed better. Decision Tree had the highest MSE in all cases, suggesting it to be the least accurate model among the four. The best and worst model performances were evident in the scatter plot shown in Figure 6, which compared actual vs. predicted values over five years of testing. The Decision Tree model displayed greater dispersion, indicating lower accuracy compared to the more tightly clustered predictions of the Linear Regression model.

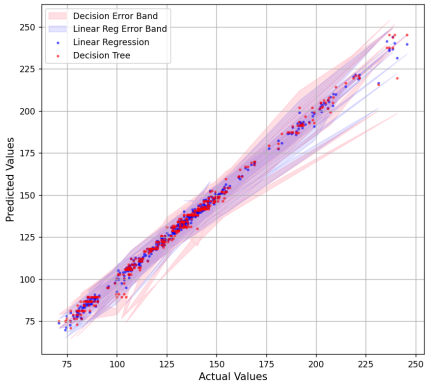

Figure 6. Scatter plot of Actual vs Predicted values for the worst and best performing models on testing data.

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

Testing the models on the training data yielded different results. As shown in Figure 7, contrary to the testing data results, Decision Tree consistently achieved the lowest MSE on the training data across all stocks, indicating excellent training performance. The wide range of MSE for decision tree models for training and testing highlighted the potential overfitting problem. Neural Network and Random Forest showed moderate MSE values, generally performing better than Linear Regression in some cases. Linear Regression had relatively high MSE values compared to other models, suggesting it did not fit the training data as closely.

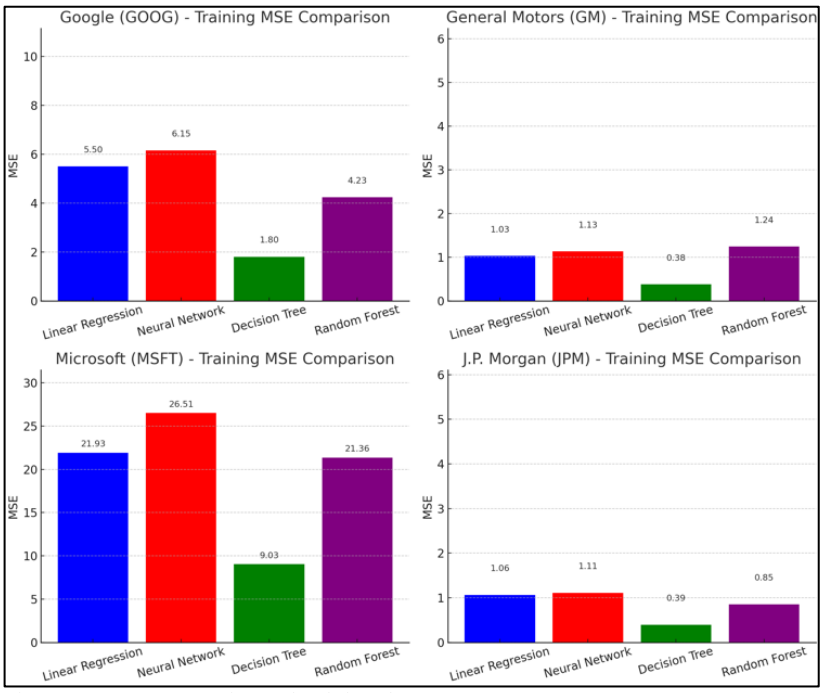

Figure 7. MSE Comparison of training data.

Since the models showed differences in performance for the testing and training data, further analysis based on 95% confidence level was performed on full data. All models were further compared using confidence intervals, as shown in Figure 8. Linear Regression performed consistently well, with low variance confidence intervals indicating high precision. Decision Tree was the weakest performer across all stocks, reflected by its wider dispersion. Random Forest showed improvement over Decision Tree but still underperformed compared to Linear Regression and Neural Network. Neural Network closely competed with Linear Regression, offering comparable performance.

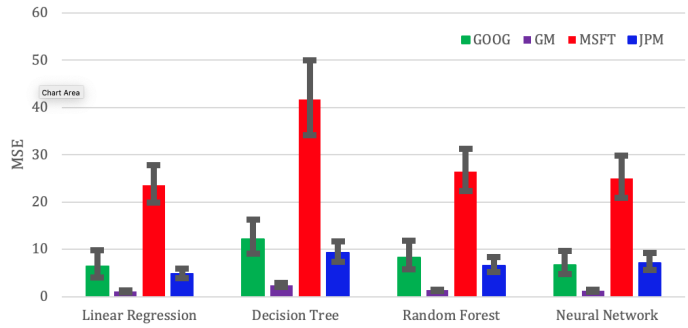

Figure 8. Model CI Comparison with 95% Confidence

Table 3 summarizes the outcomes of a trading bot simulation with varying initial investment amounts ($10,000, $20,000, $50,000, and $100,000). The bot trades in increments of one stock, simulating profits or losses for four companies: Google (GOOG), General Motors (GM), Microsoft (MSFT), and J.P. Morgan (JPM). JPM offers the highest returns across all investment levels, followed by MSFT, GOOG, and GM. Stocks like GOOG and GM show slowing returns as the initial investment increases, while MSFT and JPM maintain higher proportional growth. Investors should prioritize stocks like JPM and MSFT when using this trading bot strategy, as these provide the highest percentage returns.

Table 3 -Trader Simulation, with Increasing or Decreasing stock quantity being traded by an increment of 1.

Trading Bot - 1 stock	$10000	$20000	$50000	$100000
Google (GOOG)	$12034 (20%)	$37839 (89%)	$51540 (3%)	$101468 (1%)
General Motors (GM)	$10779 (8%)	$22010 (10%)	$51947 (4%)	$101241 (1%)
Microsoft (MSFT)	$12578 (26%)	$25685 (28%)	$56948 (14%)	$115505 (16%)
J.P. Morgan (JPM)	$18425 (84%)	$24022 (20%)	$56626 (13%)	$191164 (91%)

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

Trading in increments of 2-stocks in general results in higher absolute returns compared to 1-stock increments, but the percentage growth patterns remain similar as outlined in Table 4. For example, MSFT and JPM remain top performers, while GM continues to show limited profitability. Smaller portfolios benefit most from this strategy, while larger portfolios might require adjustments to maintain profitability. This simulation highlights the effectiveness of the trading bot for high-volatility stocks and smaller initial investments, emphasizing the need for tailored strategies based on stock type and portfolio size.

Table 4 -Trader Simulation, with Increasing or Decreasing stock quantity being traded by an increment of 2.

Trading Bot - 2 stocks	$10000	$20000	$50000	$100000
Google (GOOG)	$14069 (41%)	$47729 (139%)	$53080 (6%)	$102936 (3%)
General Motors (GM)	$11559 (16%)	$24021 (20%)	$53894 (8%)	$102482 (2%)
Microsoft (MSFT)	$15156 (52%)	$31370 (57%)	$63897 (28%)	$131011 (31%)
J.P. Morgan (JPM)	$17912 (79%)	$28044 (40%)	$63253 (27%)	$118775 (19%)

These results highlight the trade-offs between model complexity and generalization. While decision trees excel on training data, simpler models like linear regression may be better suited for real-world applications where testing performance is critical. Further fine-tuning of more complex models like the Neural Network could potentially bridge the gap between underfitting and overfitting.

4. Discussion

Based on the results, linear regression was selected for the trading simulation due to its lowest MSE on testing data and tightly bounded confidence intervals. The simulation predicted profitable outcomes over a five-year period, even when trading only one stock per day. Increasing the trade volume to two stocks per day yielded even higher returns. The highest profit was observed with a $20,000 investment in Google, while the lowest was with a $100,000 investment in General Motors. Notably, an investment of $100,000 in Microsoft was projected to grow to $131,011 after five years.

For future research, several key questions can be explored to extend this study. These include: How might the outcome of a presidential election influence stock price predictions using AI models? In what ways could geopolitical events, such as the onset of war, alter predictive outcomes? How can natural disasters impact the accuracy of AI-driven stock forecasts? To what extent can political statements or social media activity (e.g., tweets) affect stock market predictions using AI?

Although the trading bot cannot predict exact market movements, it offers valuable insights into identifying patterns and trends. While no tool can guarantee success, the model aids investors in spotting profitable opportunities and managing risk. With careful application, it can enhance investment decision-making.

5. Conclusion

This study examined the use of machine learning models to predict stock market trends and simulate trading strategies using historical data. Among the four models tested, Linear Regression, Decision Trees, Neural Networks, and Random Forest, Linear Regression proved to be the most effective based on testing data, achieving the lowest MSE and tightly bounded confidence intervals. As a result, it was used in the simulation to guide trading decisions.

The trading bot showed consistent profitability across investment scenarios. Strategies involving larger trade volumes demonstrated even greater profit potential, illustrating the scalability of the approach.

While the model's reliance on historical data limits its ability to account for external shocks, incorporating variables such as political events, global crises, and social sentiment could improve its robustness.

Though it does not guarantee precision, the trading bot serves as a practical tool for investors by highlighting trends and supporting risk-aware decisions. With continued refinement, this AI-driven approach can offer more reliable and impactful investment guidance.

Vol. 2025 (11) 896 – 903
ISSN 2688-3651 [online]

References

Patel, A., et al. (2021). Prediction of stock market using artificial intelligence. *Proceedings of the 4th International Conference on Advances in Science & Technology (ICAST2021)*. https://ssrn.com/abstract=3871022

Singh, G., (2022). Machine Learning Models in Stock Market Prediction. Cornell University. https://arxiv.org/abs/2202.09359

Tupe-Waghmare, P., (2021). Prediction of stocks and stock price using artificial intelligence: A bibliometric study using Scopus database. *Library Philosophy and Practice (e-journal). 5369.* https://digitalcommons.unl.edu/libphilprac/5369

Zheng, A., & Jin, J. (n.d.). Using AI to make predictions on stock market. Stanford University. https://cs229.stanford.edu/proj2017/final-reports/5212256.pdf

Zou, J., et al. (2022). Stock Market Prediction via Deep Learning Techniques: A Survey. Cornell University. https://arxiv.org/abs/2212.12717