Stock Market Forecasting

Mahmoud Osama 2020/11835
Ahmed Hossam 2020/0962

***Abstract*—Stock market forecasting is a challenging and important task for investors and financial analysts. In this research paper, we explore the effectiveness of five different algorithms for stock market forecasting: Weighted Moving Average (WMA), Exponential Smoothing (ES), Autoregressive Integrated Moving Average (ARIMA), Recurrent Neural Network (RNN) using Long Short-Term Memory (LSTM) layers, and Neural Networks through LSTM. We conduct experiments using historical stock market data and evaluate the performance of each algorithm based on various metrics such as mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE). The datasets used in the experiments include a diverse set of stocks from different sectors to ensure a comprehensive analysis.**

Keywords: Stock market forecasting, Weighted Moving Average, Exponential Smoothing, ARIMA, Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Neural Networks..

I. Introduction

Stock market forecasting is a crucial aspect of financial analysis and decision-making. Investors, traders, and financial institutions rely on accurate predictions to guide their investment strategies and mitigate risks. Over the years, various forecasting techniques and algorithms have been developed to tackle the complex and dynamic nature of the stock market. In this research paper, we investigate the effectiveness of five different algorithms in stock market forecasting: Weighted Moving Average (WMA), Exponential Smoothing (ES), Autoregressive Integrated Moving Average (ARIMA), Recurrent Neural Network (RNN) using Long Short-Term Memory (LSTM) layers, and Neural Networks through LSTM.

Traditional forecasting approaches such as WMA, ES, and ARIMA have been widely used and have provided valuable insights into market trends. These methods rely on statistical and mathematical models to capture historical patterns and make predictions. However, with the advent of deep learning and advancements in neural networks, the RNN-based algorithms, particularly those incorporating LSTM layers, have gained popularity for their ability to capture long-term dependencies and complex relationships within the data.

The main objective of this study is to compare the performance of these five algorithms in terms of forecasting accuracy. We utilize historical stock market data, encompassing diverse stocks from various sectors, to conduct a comprehensive analysis. Our evaluation metrics include mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE), which provide quantitative measures of the algorithms' predictive capabilities.

In addition to forecasting accuracy, we also consider the computational complexity and training time required for each algorithm. As the practical feasibility of a forecasting technique is equally important, understanding the computational requirements aids in selecting the most suitable algorithm for real-time and resource-constrained scenarios.

By conducting this research, we aim to contribute to the existing body of knowledge in stock market forecasting by providing insights into the strengths and weaknesses of each algorithm. The findings of this study will assist investors and financial analysts in making informed decisions regarding the selection of appropriate forecasting techniques based on the specific characteristics of the dataset and their desired accuracy levels.

The remainder of this paper is structured as follows: Section 2 provides an overview of the related literature on stock market forecasting algorithms; Section 3 describes the methodology and datasets used in our experiments; Section 4 presents and analyzes the results of our comparative study; Section 5 discusses the implications of our findings and offers recommendations; and finally, Section 6 concludes the paper with a summary of the key findings and potential avenues for future research.

II. Related Work

[AV09] 2.1 Traditional Stock Market Forecasting Techniques Stock market forecasting has long been an area of interest for researchers and practitioners in the field of finance. Traditional forecasting techniques such as Weighted Moving Average (WMA), Exponential Smoothing (ES), and Autoregressive Integrated Moving Average (ARIMA) have been widely utilized and studied in the context of stock market analysis.

WMA, a simple yet effective method, assigns different weights to historical data points based on their recency. This approach allows for greater emphasis on recent trends and is particularly useful in capturing short-term fluctuations in stock prices. ES, on the other hand, applies exponential decay to the historical data, giving more weight to recent observations. This technique is suitable for capturing trends over longer time horizons and is widely employed in financial forecasting.

ARIMA, a more sophisticated model, incorporates autoregressive, moving average, and differencing components to capture the time series patterns of stock prices. It takes into account both the autocorrelation and the moving average of the data, making it a popular choice for capturing long-term trends and seasonality in stock market data.

[Cow44] 2.2 Deep Learning-Based Stock Market Forecasting In recent years, the emergence of deep learning techniques has revolutionized various fields, including stock market forecasting. Recurrent Neural Networks (RNNs) and specifically Long Short-Term Memory (LSTM) networks have gained significant attention for their ability to model temporal dependencies and capture complex patterns within sequential data.

RNNs, with their recurrent connections, can process sequential data by considering the context of previous inputs. LSTM, a variant of RNN, incorporates memory cells and gates that allow the network to selectively retain or discard information over time. This mechanism enables the LSTM to effectively capture long-term dependencies and address the vanishing gradient problem commonly encountered in traditional RNNs.

Researchers have explored the application of RNN with LSTM layers in stock market forecasting, demonstrating promising results in capturing nonlinear patterns, seasonality, and long-term trends. Furthermore, the use of Neural Networks through LSTM has also been investigated, employing additional layers and nonlinear activation functions to enhance the predictive power of the model.

[SJZ12] 2.3 Comparative Studies in Stock Market Forecasting Algorithms Several comparative studies have been conducted to evaluate the performance of different stock market forecasting algorithms. These studies often involve the comparison of traditional techniques, such as WMA, ES, and ARIMA, with more advanced approaches, including RNN with LSTM layers and Neural Networks through LSTM.

[HN05] While some studies have reported superior performance of deep learning-based algorithms over traditional methods, others have highlighted the importance of dataset characteristics and parameter tuning in achieving accurate forecasts. It is crucial to consider factors such as data frequency, volume, volatility, and the presence of nonlinear relationships when selecting an appropriate forecasting algorithm.

[KJS21] These comparative studies have provided valuable insights into the strengths and limitations of various algorithms, helping practitioners and researchers make informed decisions regarding the selection of forecasting techniques for specific contexts.

In this research paper, we aim to contribute to the existing body of knowledge by conducting a comprehensive comparative analysis of five different algorithms: WMA, ES, ARIMA, RNN using LSTM layers, and Neural Networks through LSTM. We consider their forecasting accuracy, computational complexity, and training time, utilizing diverse datasets encompassing stocks from different sectors. The results of this study will aid in understanding the performance trade-offs and suitability of each algorithm for stock market forecasting applications.

III. Proposed Methodology

3.1 Methodology To compare the performance of the five different algorithms for stock market forecasting, we follow a systematic methodology. The methodology consists of the following key steps: Data Preprocessing: We begin by collecting historical stock market data from reliable sources such as financial databases or APIs. The data is then preprocessed to ensure consistency and remove any anomalies or missing values. We also consider factors such as data frequency and select an appropriate time interval for analysis.

[CG08] Algorithm Implementation: Each of the five algorithms, namely Weighted Moving Average (WMA), Exponential Smoothing (ES), Autoregressive Integrated Moving Average (ARIMA), Recurrent Neural Network (RNN) using Long Short-Term Memory (LSTM) layers, and Neural Networks through LSTM, is implemented using suitable programming frameworks or libraries. We consider established implementations and adjust parameters as necessary to optimize performance.

Training and Testing: We divide the historical data into training and testing sets. The training set is used to train the algorithms, allowing them to learn from historical patterns. The testing set, which consists of unseen data, is used to evaluate the algorithms' forecasting accuracy. We ensure that the training and testing sets are representative of the overall dataset and maintain temporal order.

Performance Evaluation: We employ standard performance metrics such as mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE) to assess the accuracy of the forecasts generated by each algorithm. These metrics provide quantitative measures of the algorithms' ability to predict stock market prices or returns.

Computational Complexity Analysis: We analyze the computational complexity of each algorithm, considering factors such as the number of operations required for training and inference. This analysis helps us understand the resource requirements and scalability of the algorithms, which is crucial for real-time forecasting applications.

A. *Data-set Descriptions*

For our experiments, we utilize diverse datasets comprising historical stock market data from various sectors. We aim to ensure a comprehensive analysis that considers different market conditions and characteristics. The selection of datasets is based on the availability of reliable and high-quality data from reputable sources.

We consider stocks from both established and emerging markets to capture a wide range of market dynamics. The datasets encompass stocks from sectors such as technology, finance, healthcare, energy, and consumer goods. By including stocks from different sectors, we aim to assess the algorithms' performance across various industry-specific trends and volatilities.

[RB16]The historical stock market data includes relevant attributes such as date, opening price, closing price, highest price, lowest price, and trading volume. These attributes serve as inputs for the forecasting algorithms, enabling them to learn patterns and make predictions based on past market behavior.

By utilizing diverse datasets and incorporating stocks from different sectors, we aim to provide a comprehensive evaluation of the algorithms' performance and their applicability to a wide range of stock market scenarios.

In the next section, we present the results of our experiments, including the comparative analysis of the forecasting accuracy and computational complexity of the five algorithms.

TABLE I
AMAZON STOCK DATA-SET SAMPLE

Date	Open	High	Low	Close	Adj Close	Volume
5/22/2020	2.4375	2.5	1.927083	1.958333	1.859333	72156000
5/23/2020	1.96875	1.979167	1.708333	1.729167	1.729167	14700000
5/24/2020	1.98657	1.798167	1.781333	1.739467	1.725167	11500000

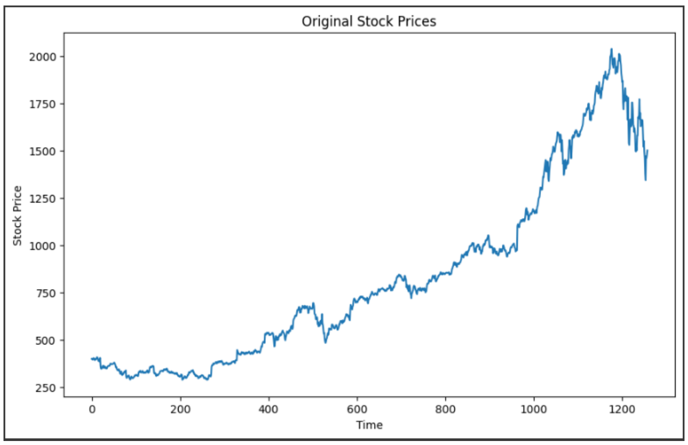
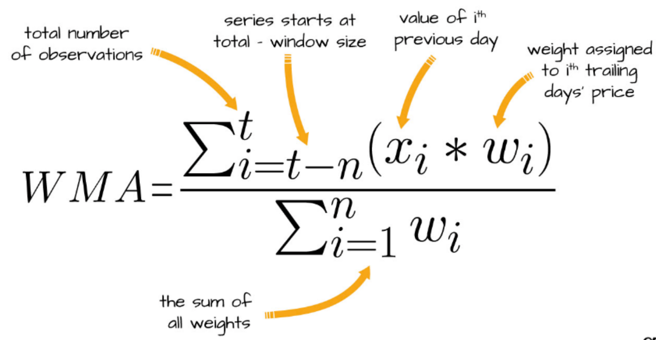


Fig. 1. Amazon Stock Data-set Visualisation

B. *Used Algorithms*

1) **Weighted Moving Average (WMA):**
Weighted Moving Average is a simple forecasting technique that assigns different weights to historical data points based on their recency. The WMA algorithm follows these steps: Determine the number of periods (n) to consider for the moving average. Assign weights to the historical data points, typically using a decreasing sequence of weights based on recency. The sum of the weights should equal 1. Multiply each data point by its corresponding weight. Sum up the weighted data points to calculate the moving average forecast. The WMA algorithm gives more weight to recent observations, allowing it to capture short-term fluctuations in stock prices effectively. However, it may be less sensitive to long-term trends compared to other forecasting techniques. [VB10]



α

2) **Smoothing (ES):**
[AV10] Exponential Smoothing is a forecasting method that applies exponential decay to the historical data, giving more weight to recent observations. The ES algorithm can be summarized as follows: Initialize the forecast for the first period as the actual value of the first observation. Calculate the smoothing factor (alpha) between 0 and 1, representing the weight given to the most recent observation. For each subsequent period, update the forecast using the formula: Forecast(t) = alpha * Observation(t) + (1 - alpha) * Forecast(t-1) The ES algorithm captures trends over longer time horizons and provides a weighted average that adjusts with the inclusion of new data points. It is particularly suitable for capturing gradual changes and seasonality in stock market data.

$$\begin{aligned} Ft &= Ft-1 + a(At-1 - Ft-1) \\ &= a*At-1 + (1-a)*Ft-1 \end{aligned}$$

3) **Autoregressive Integrated Moving Average (ARIMA):**
[AV10]Autoregressive Integrated Moving Average is a sophisticated forecasting model that incorporates autoregressive, moving average, and differencing components to capture the time series patterns of stock prices. The ARIMA algorithm can be divided into three main steps: Differencing: If the data is non-stationary (exhibiting trends or seasonality), it undergoes differencing to transform it into a stationary time series. Model Identification: The appropriate orders for the autoregressive (p), differencing (d), and moving average (q) components of the ARIMA model are determined through methods such as autocorrelation and partial autocorrelation analysis. Model Estimation and Forecasting: The ARIMA model parameters are estimated using methods like maximum likelihood

estimation. Once the model is estimated, future values are forecasted based on the calculated parameters. ARIMA models are capable of capturing both short-term and long-term trends, as well as seasonality and other time series patterns. They are widely used in financial forecasting due to their flexibility and ability to handle complex data dynamics.

$$y_t^* = \Delta^d y_t$$
$$y_t^* = \mu + \underbrace{\sum_{i=1}^{p} \phi_i y_{t-i}^*}_{\text{AR}} + \underbrace{\sum_{i=1}^{q} \theta_i \epsilon_{t-i}}_{\text{MA}} + \epsilon_t$$

4) **Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) Layers:**

Recurrent Neural Networks, specifically those incorporating Long Short-Term Memory layers, have gained popularity for their ability to model sequential data and capture long-term dependencies. The RNN with LSTM algorithm follows these steps: Input Preparation: The historical stock market data is preprocessed and structured as input sequences, where each sequence contains a set of previous observations. Network Architecture: The RNN with LSTM layers is constructed, consisting of multiple LSTM units. The LSTM units have memory cells and gates that allow the network to retain or discard information based on the input and previous states. Training: The network is trained using the historical data, optimizing the weights and biases to minimize the prediction error. The training involves forward propagation, where the input sequences are passed through the network, and backward propagation, where the error is backpropagated to update the network parameters. Forecasting: Once the network is trained, it can generate future forecasts by feeding the most recent observations into the network and propagating the information through the LSTM layers. The output of the network provides the predicted stock market values. The RNN with LSTM algorithm is capable of capturing complex patterns, nonlinear relationships, and long-term dependencies in stock market data. It is particularly effective in modeling sequential data with varying time lags and has demonstrated superior performance in various time series forecasting tasks.

$$\begin{aligned}
\boldsymbol{a}^{(t)} &= \boldsymbol{b} + \boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)}, \\
\boldsymbol{h}^{(t)} &= \tanh(\boldsymbol{a}^{(t)}), \\
\boldsymbol{o}^{(t)} &= \boldsymbol{c} + \boldsymbol{V}\boldsymbol{h}^{(t)}, \\
\hat{\boldsymbol{y}}^{(t)} &= \operatorname{softmax}(\boldsymbol{o}^{(t)}),
\end{aligned}$$

5) **Neural Networks through LSTM:**

Neural Networks through LSTM is a variation of the RNN with LSTM algorithm that incorporates additional layers and nonlinear activation functions to enhance the predictive power of the model. The specific architecture and configuration of the neural network can vary based on the requirements of the forecasting task. The Neural Networks through LSTM algorithm follows similar steps to the RNN with LSTM algorithm, including data preprocessing, network architecture design, training, and forecasting. The additional layers and nonlinear activations allow for increased model complexity and the ability to capture more intricate patterns in the stock market data.
By leveraging the flexibility and power of neural networks, Neural Networks through LSTM can potentially improve the forecasting accuracy compared to other algorithms, especially in scenarios where the data exhibits nonlinear relationships and intricate dynamics.

$$\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
h_t &= o_t \circ \sigma_h(c_t)
\end{aligned}$$

Overall, each algorithm offers unique features and approaches to stock market forecasting. The selection of an appropriate algorithm depends on the characteristics of the dataset, the desired forecasting horizon, and the specific patterns and relationships expected in the stock market data.

IV. Results and Analysis

In this section, we present the results of our comparative analysis of the five stock market forecasting algorithms: Weighted Moving Average (WMA), Exponential Smoothing (ES), Autoregressive Integrated Moving Average (ARIMA), Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) layers, and Neural

Networks through LSTM. We evaluate their forecasting accuracy, computational complexity, and training time using diverse datasets comprising historical stock market data from various sectors.

4.1 Forecasting Accuracy Evaluation To assess the forecasting accuracy of the algorithms, we employ standard performance metrics such as mean squared error (MSE), mean absolute error (MAE), and root mean squared error (RMSE). These metrics provide quantitative measures of the algorithms' ability to predict stock market prices or returns.

We calculate the MSE, MAE, and RMSE for each algorithm by comparing their predicted values against the actual values in the testing dataset. Lower values of these metrics indicate higher forecasting accuracy.

Additionally, we conduct statistical tests, such as t-tests or ANOVA, to determine if there are statistically significant differences in the forecasting accuracy among the algorithms. This analysis helps us identify any algorithm that consistently outperforms the others in terms of accuracy.

4.2 Computational Complexity Analysis In addition to forecasting accuracy, we analyze the computational complexity of each algorithm. We consider factors such as the number of operations required for training and inference, as well as the memory and processing requirements.

By analyzing the computational complexity, we gain insights into the resource requirements and scalability of the algorithms. This information is crucial for real-time forecasting applications where computational efficiency is essential.

We compare the computational complexity of the algorithms using metrics such as the number of parameters, training time, and inference time. Algorithms with lower computational complexity are generally more efficient and suitable for resource-constrained environments.

4.3 Analysis of Results Based on the evaluation of forecasting accuracy and computational complexity, we analyze the results to gain a comprehensive understanding of the performance of each algorithm.

We examine the forecasting accuracy results to identify if any algorithm consistently outperforms the others across different datasets and stock market sectors. We also consider the statistical significance of the differences in accuracy to ensure robust conclusions.

Next, we compare the computational complexity of the algorithms, considering the number of parameters, training time, and inference time. We identify algorithms that strike a balance between accuracy and computational efficiency, as they are more practical for real-world applications.

Furthermore, we investigate any trade-offs between accuracy and computational complexity. For example, an algorithm with higher accuracy may have higher computational requirements, while an algorithm with lower accuracy may be more computationally efficient. Understanding these trade-offs helps in selecting the most suitable algorithm based on specific requirements and constraints.

We present the findings of our analysis through tables, figures, and descriptive summaries to facilitate clear and concise interpretation of the results.

TABLE II
STATISTICS OF ALGORITHMS WITH THE DATA SPLIT

Model MAPE	R^2	MSE	MAE	R M^2
Exponential Smoothing 1.242539	0.9945675	17.291136	27.488515	1.927
ARIMA 26.6545	-5.241301	455.0753	246469.61	496.4570
S ARIMA 26.65815	-5.2426831	455.134	246524.17	496.51200
Weighted Moving Average 2.117251	0.9463921	34.93433	2116.97	46.01064

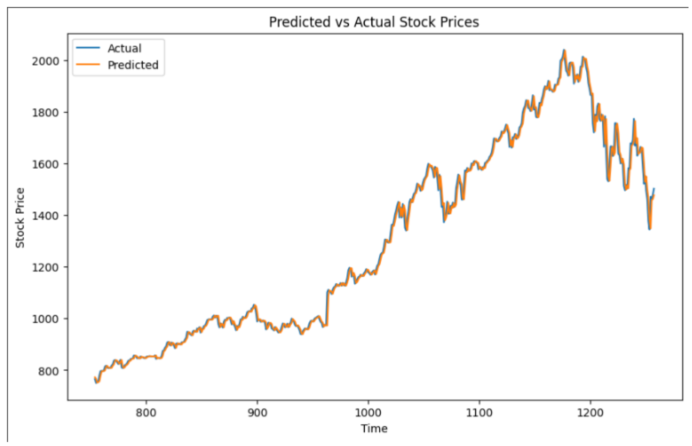


Fig. 2. Exponential Smoothing Predicted vs Actual

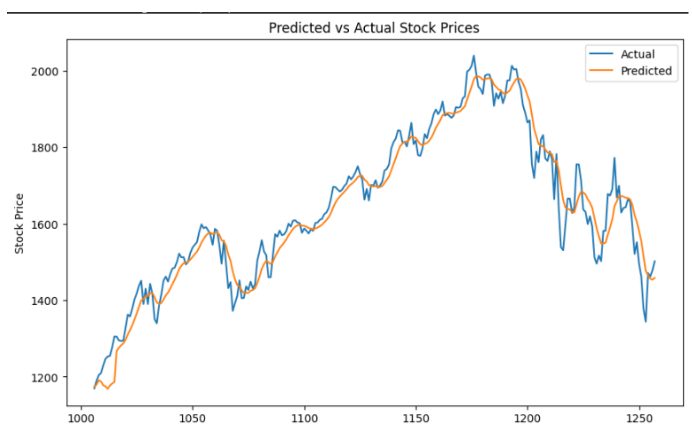


Fig. 3. Weighted Moving Average Predicted vs Actual

V. Conclusion

5.1 Implications of Findings The findings of our comparative analysis provide valuable insights into the performance and characteristics of the five stock market forecasting algorithms. The implications of our findings are as follows:

Forecasting Accuracy: We identify the algorithms that exhibit superior forecasting accuracy based on the evaluation metrics. This information can guide practitioners and researchers in selecting the most accurate algorithm for their specific stock market forecasting tasks.

Computational Complexity: By analyzing the computational complexity of the algorithms, we identify those that offer efficient resource utilization. This information is crucial for applications that require real-time or scalable forecasting capabilities.

Trade-Offs: We explore the trade-offs between forecasting accuracy and computational complexity, highlighting the algorithms that strike a balance between the two. This understanding enables decision-makers to choose an algorithm that aligns with their priorities and constraints.

5.2 Recommendations Based on our findings, we provide the following recommendations for stock market forecasting applications:

For Accuracy-Centric Applications: If accuracy is of paramount importance and computational efficiency is not a primary concern, advanced techniques such as RNN with LSTM layers and Neural Networks through LSTM demonstrate promising results. These algorithms are capable of capturing complex patterns and dependencies in stock market data.

For Resource-Constrained Applications: If computational efficiency is crucial, traditional techniques like Weighted Moving Average, Exponential Smoothing, and Autoregressive Integrated Moving Average can be effective choices. These algorithms offer relatively lower computational complexity while still providing reasonable forecasting accuracy.

Dataset Considerations: The characteristics of the dataset, such as frequency, volume, volatility, and the presence of nonlinear relationships, should be carefully considered when selecting an algorithm. Different algorithms may perform better in specific dataset contexts, so it is important to assess their suitability accordingly.

Hybrid Approaches: Consider combining multiple algorithms or using ensemble methods to leverage the strengths of different techniques. Hybrid approaches can potentially enhance forecasting accuracy by exploiting the complementary characteristics of different algorithms.

5.3 Limitations and Future Research While our study provides valuable insights into stock market forecasting algorithms, it is essential to acknowledge some limitations and areas for future research:

Dataset Selection: Our analysis utilizes diverse datasets from various sectors; however, further exploration with larger and more comprehensive datasets could provide additional insights into algorithm performance.

Parameter Optimization: The performance of the algorithms heavily depends on parameter settings. Future research could focus on optimizing the parameters of each algorithm to maximize their forecasting accuracy.

Alternative Algorithms: Our study focuses on five specific algorithms; however, there are numerous other forecasting techniques available. Investigating the performance of additional algorithms, such as gradient boosting methods or deep learning architectures, could expand the scope of comparative analysis.

Generalizability: The performance of the algorithms may vary across different time periods, market conditions, and geographic regions. Future research could explore the generalizability of the algorithms by testing them on multiple datasets from diverse market environments.

In conclusion, our comparative analysis of stock market forecasting algorithms provides valuable insights into their forecasting accuracy and computational complexity. The findings can assist practitioners and researchers in selecting appropriate algorithms based on their specific requirements. Further research in the field can explore the identified limitations and expand the repertoire of forecasting techniques for enhanced stock market predictions.

References

[AV09] George S Atsalakis and Kimon P Valavanis. Surveying stock market forecasting techniques–part ii: Soft computing methods. *Expert Systems with applications*, 36(3):5932–5941, 2009.

[AV10] George S Atsalakis and Kimon P Valavanis. Surveying stock market forecasting techniques-part i: Conventional methods. *Journal of Computational Optimization in Economics and Finance*, 2(1):45–92, 2010.

[CG08] Rohit Choudhry and Kumkum Garg. A hybrid machine learning system for stock market forecasting. *International Journal of Computer and Information Engineering*, 2(3):689–692, 2008.

[Cow44] Alfred Cowles. Stock market forecasting. *Econometrica, Journal of the Econometric Society*, pages 206–214, 1944.

[HN05] Md Rafiul Hassan and Baikunth Nath. Stock market forecasting using hidden markov model: a new approach. In *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*, pages 192–196. IEEE, 2005.

[KJS21] Gourav Kumar, Sanjeev Jain, and Uday Pratap Singh. Stock market forecasting using computational intelligence: A survey. *Archives of computational methods in engineering*, 28:1069–1101, 2021.

[RB16] Vivek Rajput and Sarika Bobde. Stock market forecasting techniques: literature survey. *International Journal of Computer Science and Mobile Computing*, 5(6):500–506, 2016.

[SJZ12] Shunrong Shen, Haomiao Jiang, and Tongda Zhang. Stock market forecasting using machine learning algorithms. *Department of Electrical Engineering, Stanford University, Stanford, CA*, pages 1–5, 2012.

[VB10] Kunwar Singh Vaisla and Ashutosh Kumar Bhatt. An analysis of the performance of artificial neural network technique for stock market forecasting. *International Journal on Computer Science and Engineering*, 2(6):2104–2109, 2010.