Kurdistan Journal of Applied Research (KJAR)
Print-ISSN: 2411-7684 | Electronic-ISSN: 2411-7706

Original Article

Check for updates

Enhanced AdaBoostM1 with Multilayer Perceptron for Stock Price Prediction

Rebwar Mala Nabi [a b *], Soran AB. Saeed [a], Habibollah Haron [c]

[a] Technical College of Informatics, Sulaimani Polytechnic University, Sulaymaniyah, Iraq.
[b] Kurdistan Technical Institute, Sulaymaniyah, Iraq.
[c] University of Technology Malaysia, Johor, Malaysia.

Submitted: 10 March 2023
Revised: 15 April 2023
Accepted: 15 May 2023

*** Corresponding Author**:
rebwar.nabi@spu.edu.iq

Keywords: Stock Market Price Prediction, AdaboostM1, Boosting Algorithms, Multilayer Perceptron, WEKA, Multiclass Classification.

How to cite this paper: R. M. Nabi, S. AB. Saeed, H. Haron, "Enhanced AdaBoostM1 with Multilayer Perceptron for Stock Price Prediction", KJAR, vol. 8, no. 1, pp. 60-72, Jun. 2023, doi: 10.24017/science.2023.1.7

Abstract: Stock market investment has gained significant popularity due to its potential for economic returns, prompting extensive research in financial time series forecasting. Among the predictive models, various adaptations of the AdaBoostM1 algorithm have been applied to stock market prediction, either by tuning parameters or experimenting with different base learners. However, the achieved accuracy often remains suboptimal. This study addresses these limitations by introducing an enhanced version of AdaBoostM1 (ADA), implemented on the Waikato Environment for Knowledge Analysis (WEKA) platform, to forecast stock prices using historical data. The proposed model, termed AdaBoost with Multilayer Perceptron (ADA-MLP), replaces the commonly used Decision stumps with a set of Multilayer Perceptron (MLP) models as weak learners. The experimental results demonstrate that ADA-MLP consistently outperformed the standard AdaBoostM1 algorithm, achieving an average classification accuracy of 100%, compared to 98.48% by AdaBoostM1—a relative improvement of 1.52%. Additionally, ADA-MLP demonstrated superior performance against other enhanced versions of AdaBoost presented in prior studies, achieving an average of 5.3% higher accuracy. Statistical significance testing using the paired t-test confirmed the reliability of these results, with p-values < 0.05. The experiments were conducted on the yahoo finance dataset from 25 years of historical data spanning from January 1995 to January 2020, comprising 6295 samples ensuring a robust and comprehensive evaluation. These findings highlight the potential of ADA-MLP to enhance financial forecasting and offer a reliable tool for stock market prediction. Future research could explore extending this approach to other financial instruments and larger datasets to further validate its effectiveness.

1. Introduction

Stock Market Prediction is recognized as a complicated non-linear dynamic system influenced by multiple factors [1-6]. In the past decades it is been identified that traditional analysis and forecasting methods have been insufficient to accurately expose the inherent pattern for the stock market. Therefore, big differences between expected and predicted results are noticed [7-9] . However, recently various machine learning methods have been applied to predict the future stock market prices more accurately and precisely such as Support Vector Machine, Decision Trees, Fuzzy and Neural Networks [10-13]. Furthermore, ensemble methods were also applied massively, in which Adaboost is considered as the most famous compare to other ensembles [9, 10].

Numerous studies have been published in the past several decades using Adaboost for stock price prediction [12-18]. Initially, the Adaboost was founded by Freund and Schapire [11]. They proposed the new boosting algorithm, which does not require any previous knowledge of the weak hypothesis nonetheless it adapts to the accuracy and will produce the weighted majority proposition. This exposes that there is a persistent development in the accuracy of the final hypothesis when any of the weak hypothesis is developed. Their final description is desperately different from the report of boosting by the previous studies [1, 15, 19-21].

Additionally, numerous enhancements and versions of ensembles have been proposed. Initially, Nazário *et al.* [21] introduced an ensemble of multi-layer feedforward networks for predicting Chinese stocks. Three constituent networks were trained utilizing training techniques including backpropagation and Adam. The ensemble was constructed with the bagging methodology. The findings indicate that Chinese markets are partially predictable, achieving satisfactory accuracy, precision, and recall. Additionally, Guoying and Ping [14] introduced the Adaboost integration algorithm, employing diverse prediction variables to forecast annual stock returns. They adjusted the weight and the weak learner parameter in response to the misclassifications that occurred.

Furthermore, Sun, *et al*. [23] proposed a novel financial distress prediction model based on the synthetic minority oversampling technique (SMOTE). Their model was combined with the Adaboost Support Vector Machine Ensemble Integrating with Time Weighting (ADASVM-TW). Their model was evaluated on a total of 2628 Chinese listed companies for ten years. Their ADASVM-TW achieved a classification accuracy (CA) of 91.22% on average. Equally, Riberio and Coelho [20] conducted intensive experiments using ensembles bagging (random forests - RF), boosting (gradient boosting machine - GBM and extreme gradient boosting machine - XGB), and stacking.

Similarly, Sun *et al.* [24] proposed several versions of Adaboost such as Adaboost with Long Short-Term Memory (AdaBoost-LSTM), AdaBoost with support vector regression (SVR), and Adaboost with Extreme Learning Machine (ELM). However, no details were provided regarding the versions of Adaboost.

Moreover, Wang and Bai [25] used GBM with an Artificial Neural Network (ANN) the boosting-ANN was construed by using ANN to build strong learners from three weak learners. The Boosting-ANN was evaluated on NASDAQ and S&P 500. Furthermore, their model was used as gradient boosting machine concept in the replacement of Adaboost.

In addition, Kang and Michalak [26] proposed an improved version of AdaboostM1 in Waikato Environment for Knowledge Analysis (WEKA) by changing the base learner from decision stump to J48 Tree algorithm with tuning weight threshold (P) and the number of iterations for boosting algorithm. Their evaluation method and metrics were not quite accurate however, based on their conclusion, the average error rate was reduced by 1.5%. Besides, Chang *et al*. [6] suggested an enhanced Adaboost algorithm for computational financial analysis, integrating the Adaboost algorithm with an Average True Range–Relative Strength Index strategy. The Hushen 300 index used as the dataset for benchmarking their model. Their model attained a classification accuracy of 95% at its optimal prediction outcome.

To summarize, based on the above literature and intensive systematic reviews by previous studies [1, 21, 26, 28], it can be identified that the stock price prediction has not reached the reliable level, where investors can invest without fear and hesitation as well as there is a big room for improvements in terms of classification accuracy. Therefore, the aim of this study is to propose an improved version of AdaboostM1, which was implemented in WEKA. The enhanced AdaBoostM1 algorithm is proposed by integrating the set of Multilayer Perceptron (MLP) predictors instead of using decision stumps. The improved AdaBoostM1 is named Adaboost with Multilayer perceptron (ADA-MLP).

2. Methods and Materials

2.1. Proposed System

In this paper, the proposed methodology has been built using Java, leveraging the WEKA Java library for machine learning. WEKA offers a comprehensive set of machine learning algorithms for tasks such as data preprocessing, feature selection, and classification. The system was fully implemented from the ground up to conduct experiments and assess the performance of the proposed enhanced AdaBoostM1 algorithm.

2.2. Dataset Collection

As shown in Table 1, 25 years of historical data, spanning from January 1995 to January 2020, were collected from Yahoo Finance (https://finance.yahoo.com/) for companies listed on the NASDAQ, including CMCSA, CSCO, AAPL, SBUX, LRCX, MCHP, MSFT, NTAP, QCOM, and SWSK. Additionally, the GSPC composite, representing the top 500 companies on the S&P 500, was selected for analysis. Each dataset consists of 6,295 daily records. After preprocessing for monthly predictions, a total of 3,270 monthly records obtained, with 65% allocated for training and 35% reserved for testing.

Table 1: Dataset Description for monthly prediction.

Dataset	Stock	Duration	Training No	Testing No	Total
CMCSA	NASDAQ	01/01/1995 to 01/01/2020	194	105	299
AAPL			194	105	299
CSCO			189	102	290
SBUX			194	105	299
LRCX			194	105	299
MHCP			194	105	299
MSFT			194	105	299
NTAP			188	101	289
QCOM			194	105	299
SWKS			194	105	299
GSPC	S&P 500		194	105	299
Total		25 Years	2126	1145	3270

2.3. Preprocessing

Preprocessing is a critical phase in both machine learning and data mining. In this study, a novel approach is introduced for preprocessing the collected data. The stock movement is processed by comparing the predicted and actual percentage changes per month to categorize the monthly data. To determine these monthly movements, the stock's price fluctuation is calculated as the difference between the monthly closing and opening prices, as outlined in Equation 1.

$$Difference = close\ price - open\ price \qquad (1)$$

where:

Closing price refers to the price at the final date of the month.
Open price refers to the price at the commencement of the month.

The percentage (%) change in stock price is determined using Equation 2:

$$Percentage\ Difference = (Difference * 100)/(Open\ price) \qquad (2)$$

The subsequent rules were implemented to generate classification classes in a multiclass classification scenario:

If the percentage difference exceeds 1, the class is deemed positive;
If the Percentage Difference is less than -1, then the class is classified as negative.
The class is otherwise neutral.

2.4. Feature Selection

In this study, Principal Components Analysis Evaluation used as feature selection algorithm to find the best features. It is worth mentioning that the WEKA's default configuration was implemented, which means no parameter configuration was changed because it was not within the scope of this study.

2.5. Improved AdaboostM1

As stated earlier, the significant goal of this study was to improve AdaboostM1 which is implemented in WEKA [10]. To achieve this, intensive experiments were conducted on the parameters of the AdaBoostM1 Algorithm. It is widely known that one of the core parameters for each ensemble is the base classifier; namely the machine learning algorithm is used to train and form the ensemble. Therefore, this study has reached the point that to use the base classifier parameter to improve the algorithm.

Figure 1 demonstrates the process of improving the AdaBoostM1 Algorithm by finding a more effective base learner comparing with its default classifier and competing with its rivals as well as previous studies.

MLP is a model of a feed-forward ANN. Sometimes, MLP is used ambiguously, loosely to belong to any type of feed-forward ANN; whereas, few of them are used strictly to mention networks, which consist of MLP. Furthermore, MLP is also called vanilla neural networks; that have a single hidden layer.

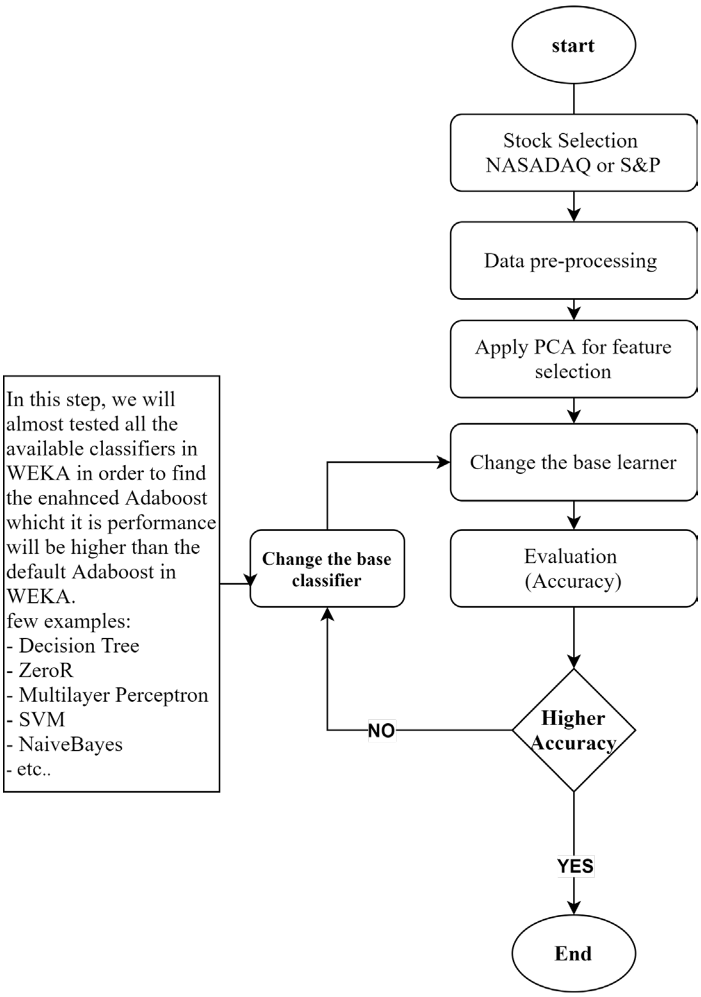

Figure. 1: Improving the AdaboostM1 algorithm flowchart.

In WEKA, MLP utilizes backpropagation to train the neural network for instance classification. The network can either be manually constructed or initialized using a basic heuristic approach. During the training process, network parameters can be tracked and adjusted as needed. The nodes within the network typically use sigmoid activation functions, except when dealing with numeric class outputs, in which case the output nodes function as un-thresholded linear units.

To integrate MLP into the AdaBoostM1 framework, the default base learner (decision stumps) was replaced with the MLP algorithm. This process involved configuring the AdaBoostM1 algorithm in WEKA to iterate over boosting rounds, training the MLP model as the weak learner in each iteration. While MLP was used in its default configuration for simplicity, further parameter tuning was conducted to optimize its performance. This included adjusting the learning rate, momentum, and the number of epochs to ensure convergence during the boosting process. Each iteration of AdaBoost assigned weights to instances based on the classification errors from the previous round, dynamically updating the MLP model. The iterative boosting process enabled the ensemble to correct misclassified instances more effectively, thereby enhancing overall accuracy. This integration demonstrates how iterative boosting and parameter optimization within WEKA were leveraged to exploit MLP's strengths in handling non-linear data patterns.

It is commonly known that when you do m-step future prediction. To make it clear, the iterative prediction blueprint is conducted in the current study, which is denoted as:

$$\hat{x}_{t+m} = f\left(x_t, x_{t-1}, \ldots, x_{t-(p-1)}\right) \quad (3)$$

where $\hat{x}$ is the prediction value, $\mathcal{X}_t$ is the real value in period t, and p indicates the lag orders. As we have mentioned earlier, the enhanced AdaBoostM1 algorithm is proposed by integrating the set of MLP predictors instead of using decision stumps. The enhanced AdaBoostM1 is named Adaboost with ADA-MLP, which will be proposed for stock market price prediction. The ADA-MLP flowchart is shown in figure 2, which is composed of three core phases.

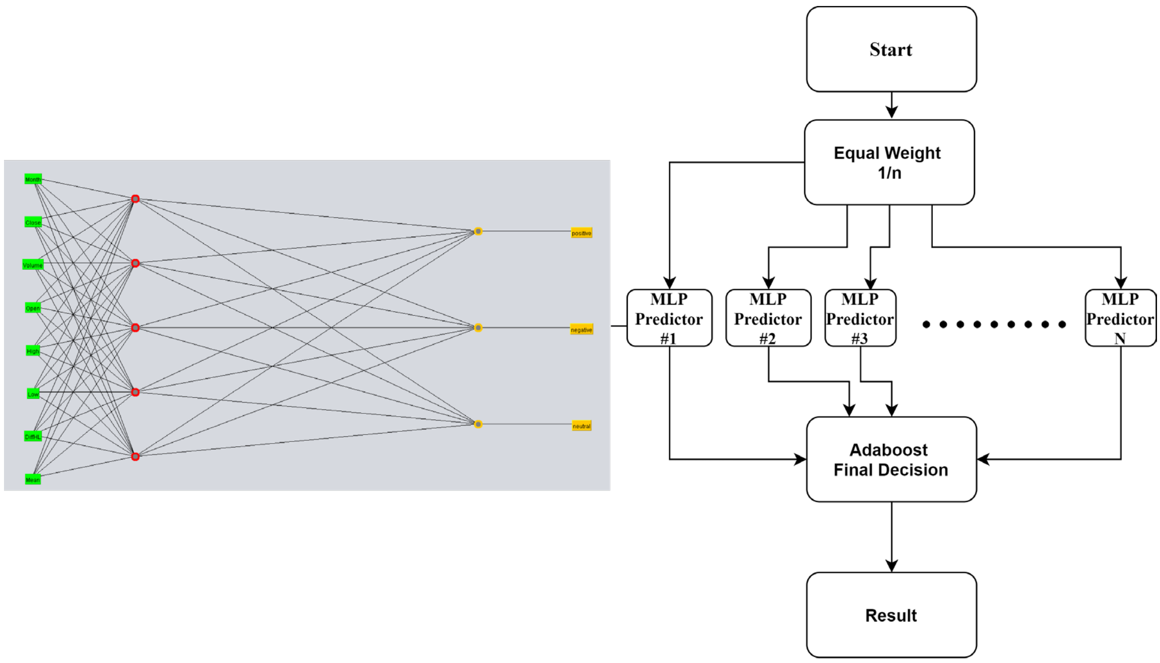

Figure 2: ADA-MLP flowchart.

- Step 1: weight's sampling $\{D_n^t\}$ of the training data $\{x_t\}_{t=1}^{T}$ are calculated by:

$$D_n^t = \frac{1}{N}, n = (1,2, \ldots, N; t = 1,2, \ldots, T) \quad (4)$$

where N is the number of MLP predictors, T is the number of training data.

- Step 2: The MLP predictor F_n is trained by the training data, which are formed according to the sampling weights D_n^t.
- Step 3: The prediction error $\{e_n'\}$ and ensemble weights $\{W_n\}$ of the MLP predictor F_n are produced by:

$$e_n^t = \frac{|x_i - \hat{x}_i|}{x_i}, (n = 1,2,\dots,N; t = 1,2,\dots,T)$$
$$D_{n+1}^t = \frac{D_n^t \beta_n^t}{\sum_{t=1}^{T} D_n^t \beta_n^t} \quad (5)$$

Update the weight's sampling D_{n+1}^t of the training data $\{x_t\}_{t=1}^{T}$ as follows:

$$D_{n+1}^t = \frac{D_n^t \beta_n^t}{\sum_{t=1}^{T} D_n^t \beta_n^t} \quad (6)$$

where $\beta_n^t = \exp(e_n^t)$ is the update amount of training samples X_t.

- Step 4: re-do the phases two to four until all the MLP predictors are produced.
- Step 5: The last prediction value will be calculated by combining the prediction results of all the MLP predictors with ensemble weights.

3. Results

To evaluate the ADA-MLP's performance, we have conducted intensive experiments on NASDAQ and S&P 500 stock datasets. In this experiment, the CA, Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), KAPPA statistics, F-Measures, Precision, and Recall are calculated for each dataset separately. Table 2 shows the CA results of the ADA-MLP. As it can be observed that ADA-MLP achieves very promising results. On averages, the ADA-MLP reaches 100% CA. Furthermore, the ADA-MLP has a MAPE of 0.67% and RMSE of 0.0095. Regarding the rest of the metrics the ADA-MLP achieves one on average for KAPPA statistics, F-Measures, precision and recall. It can be easily noted that the ADA-MLP enjoys fantastic results. In the discussion section the ADA-MLP will be compared and benchmarked with the original AdaboostM1 as well as the previous studies in the same domain.

Table 2: ADA-MLP prediction result.

Algorithm	Stock	Dataset	CA%	MAPE%	RMSE	KAPPA	F-Measures	Precision	Recall
ADA-MLP	NASDAQ	CMCSA	100	0.73	0.01	1	1	1	1
		AAPL	100	0.84	0.01	1	1	1	1
		CSCO	100	0.74	0.01	1	1	1	1
		SBUX	100	0.75	0.02	1	1	1	1
		LRCX	100	0.7	0.01	1	1	1	1
		MHCP	100	0.75	0.01	1	1	1	1
		MSFT	100	0.68	0.01	1	1	1	1
		NTAP	100	0.71	0.01	1	1	1	1
		QCOM	100	0.79	0.01	1	1	1	1
		SWKS	100	0.64	0.01	1	1	1	1
	S&P	GSPC	100	0.07	0.01	1	1	1	1
	Average		**100**	**0.67**	**0.0095**	**1**	**1**	**1**	**1**

4. Discussion

This section provides an in-depth discussion and benchmark analysis of the results obtained in this study.

4.1. Benchmark with WEKA

Figure 3 illustrates the CA comparisons between the ADA-MLP and the original ADA algorithms across all datasets for both the NASDAQ and S&P 500 indices. It is important to note that the same procedures and methodologies were employed in conducting these experiments.

As observed in Figure 3, the ADA-MLP consistently outperforms the original ADA algorithm, achieving an average CA of 100%, compared to ADA's CA of 98.48%. This represents an improvement of 1.52%, which is a statistically significant enhancement. Moreover, for specific datasets such as CSCO, LYRX, and NTAP, the CA increases were even more pronounced, with improvements of 6.94%, 3.85%, and 2.98%, respectively. However, there were instances where the CA results remained the same for certain datasets, such as AAPL and SBUX. Overall, these results indicate that ADA-MLP generally provides superior performance compared to the ADA algorithm.

In addition to the primary comparison, the study also evaluated the performance of both ADA and ADA-MLP using various feature selection algorithms available in WEKA. Figure 4 further highlights the overall CA results for both ADA and ADA-MLP when applied to the AAPL dataset, focusing on different feature selection algorithms. Similar to the previous experiment, the CA significantly improved with ADA-MLP when tested with these feature selection methods. The average CA for ADA-MLP was 76.80%, representing an impressive 13.94% increase over ADA, which recorded an average CA of only 62.86%. Notably, specific feature selection techniques, such as CHI, RFE, and IG, contributed to a dramatic accuracy increase of 20.19%.

These findings underscore the efficacy of the ADA-MLP algorithm and its ability to leverage feature selection techniques to enhance predictive performance in financial data classification.

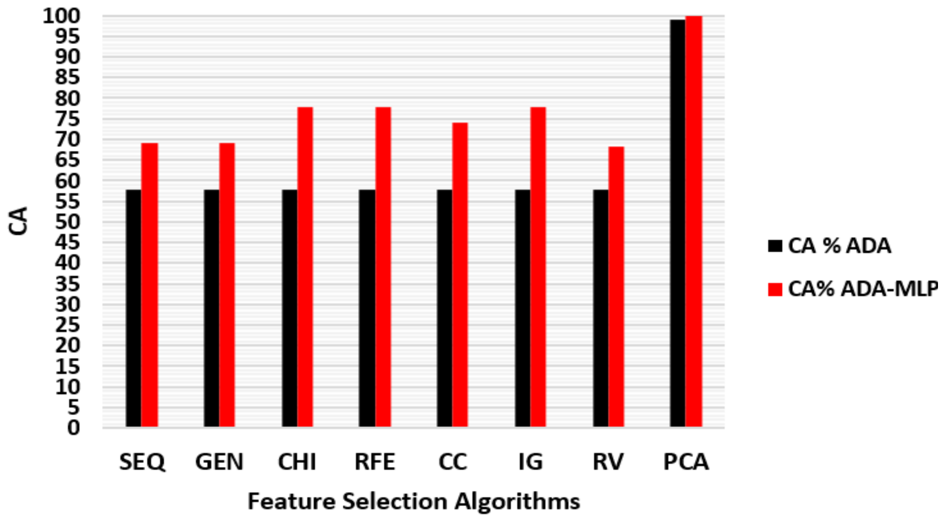

Figure 3: CA comparison between ADA-MLP and ADA with various feature selection algorithm.

The ADA-MLP model performed well across various stock types, particularly excelling with stocks that showed moderate and stable volatility. This strength likely stems from the MLP's ability to identify complex non-linear patterns in historical data, which are less pronounced in highly volatile markets. However, the model's accuracy dipped slightly during periods of extreme market fluctuations or high volatility. This is understandable, as predicting rapid and unpredictable market changes that deviate from historical patterns is inherently challenging. These results suggest that further refinement is needed, such as integrating external factors like macroeconomic data or sentiment analysis, to improve performance under high-volatility conditions.

On the practical side, using MLP as the base learner in the AdaBoost framework comes with some challenges. Training the ensemble requires substantial computational resources, especially for large datasets or when many boosting iterations are involved. The demand for memory and processing power can pose significant constraints for practitioners with limited resources. Additionally, the iterative boosting process, coupled with the backpropagation algorithm of the MLP, results in longer training times compared to simpler base learners like decision-stumps.

Despite these hurdles, ADA-MLP's superior accuracy and robustness make it a powerful tool for stock price forecasting. To address the computational demands, future research could focus on more efficient neural network designs, leveraging distributed computing, or developing hybrid models that combine MLP with simpler classifiers for specific use cases.

4.2. ADA-MLP Benchmark with Previous Studies

This section presents a benchmarking analysis of ADA-MLP against various iterations of the AdaBoost algorithm as documented in previous studies. A notable challenge in this comparative analysis was the lack of uniformity in performance evaluation metrics across the studies. Additionally, researchers utilized different stock datasets for their evaluations, making direct comparisons difficult [29]. Nevertheless, it is widely acknowledged that the inherent nature and structure of stocks are largely similar; thus, the choice of stocks often reflects researchers' preferences or geographical focus.

In the initial benchmark, Guoying and Ping [14] introduced an integrated AdaBoost algorithm that employed various prediction variables to forecast annual stock returns. They modified weights and weak learner parameters in response to misclassification errors.

As illustrated in Table 3, the overall accuracy for the advanced AdaBoost algorithm was significantly lower than that of ADA-MLP, with a notable difference of 18% in favor of ADA-MLP. While the best result for the advanced AdaBoost was highlighted, it is important to note that this model achieved only 54.8% and 47.2% accuracy in certain cases with different classes. The authors attributed these suboptimal results to the inadequacy of the stock datasets and the poor performance of weak learners when reliant on a single factor. In contrast, ADA-MLP demonstrated substantial improvement by transitioning from decision stumps to a MLP as the base learner.

Table 3: ADA-MLP comparison with [14].

Model	Stock	Average CA%
Adv. ADA	A-Share	between 66% and 82%
ADA-MLP	NAS. and S&P	100

In a similar vein, Sun *et al.* [24] developed the AdaBoost-LSTM model, which integrated AdaBoost with other algorithms such as MLP, SVR, LSTM, and ELM for financial time series forecasting. However, specific details regarding the AdaBoost-MLP model were not provided, and it was not selected as their primary method. Figure 4 illustrates a comparative analysis of the results between their models and ADA-MLP, highlighting the relative performance of these approaches.

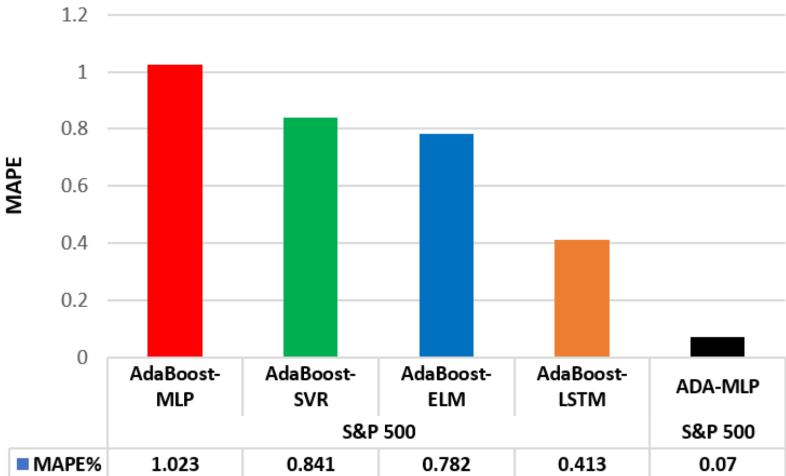

Figure 4: ADA-MLP Benchmark comparison using MAPE with [24].

As highlighted in Figure 4, the enhanced AdaBoost model (ADA-MLP) significantly outperforms all previous iterations of AdaBoost. While the best-performing approach from the literature, AdaBoost-LSTM, achieved a MAPE of 0.413%, the ADA-MLP recorded a remarkable MAPE of 0.07%. This indicates that ADA-MLP outperforms the best approach by a substantial margin of 0.343%. Furthermore, ADA-MLP consistently exceeds the performance of all other versions of AdaBoost.

Table 4 presents a comparison of ADA-MLP with the AdaBoost-GA-PWSVM model with Chullamonthon and Tangamchit [30]. This study's model demonstrates superior performance, as it outstrips the AdaBoost-GA-PWSVM approach. In selecting their best results from 36 datasets, the authors reported a 100% accuracy for two datasets and 96.5% for two others. In contrast, ADA-MLP achieved 100% accuracy on multiple datasets, including CMCSA, LYRX, and GSPC.

Moreover, ADA-MLP was evaluated on stock data from both the NASDAQ and S&P 500, across 11 datasets, achieving an impressive average accuracy of 100%. Although the authors of the AdaBoost-GA-PWSVM model did not calculate the average accuracy across all datasets, their reported results suggest an average accuracy range of 80-85%. Consequently, ADA-MLP surpasses the AdaBoost-GA-PWSVM model by approximately 15%.

Table 4: ADA-MLP benchmark with [30].

Model	Stock	Dataset	CA%
AdaBoost-GA-PWSVM	NASDAQ and SZSE	CBRL	100
		CERN	100
		AMAT	96.5
		BAX	96.5
ADA-MLP	NASDAQ and S&P 500	CMCSA	100

Another comparative analysis, as shown in Table 5, ADA-MLP outperformed the Boosting-ANN model in both MAPE and RMSE metrics. ADA-MLP recorded an average MAPE of 0.673% and RMSE of 0.0095, while the Boosting-ANN achieved a MAPE of 3.05% and RMSE of 1.35. These results highlight the limitations of their Boosting model, which focused on optimizing weak learners and weights. In contrast, the superior performance of ADA-MLP can be attributed to the strategic change in the base learner to an MLP.

Table 5: ADA-MLP benchmark with [25].

Model	Stock	MAPE%	RMSE
Boosting-ANN	NASDAQ and S&P	3.05	1.35
ADA-MLP	NASDAQ and S&P	0.673	0.0095

Additionally, Sun *et al.* [31] introduced a novel financial distress prediction model that combined the SMOTE with an AdaBoost-Support Vector Machine ensemble integrated with time weighting

(ADASVM-TW). Their model, evaluated on 2,628 Chinese listed companies over a decade, achieved an average CA of 91.22%. In comparison, ADA-MLP achieved a CA of 100%, representing an improvement of 8.78%.

Furthermore, Ribeiro and Coelho [20] conducted extensive experiments utilizing ensemble methods such as bagging (Random Forests), boosting (Gradient Boosting Machine and Extreme Gradient Boosting), and stacking. As shown in Table 6, the ADA-MLP model significantly outperformed all other methods evaluated in their study.

Table 6: ADA-MLP benchmark with [20].

Model		Stock	MAPE%
Lit.	XGB	Commodities	0.9787
	STACK		0.9855
	GBM		1.0867
	RF		1.1549
ADA-MLP		NAS and S&P	0.673

The results in Table 6 clearly demonstrate that ADA-MLP surpasses all ensemble methods tested in[20], achieving a MAPE of 0.673%, which is significantly lower than the best-performing model (XGB) with a MAPE of 0.9787%. Additionally, Kang and Michalak [26] proposed an enhanced version of AdaBoostM1 in WEKA, utilizing the J48 tree algorithm as the base learner and tuning the weight threshold and iteration count for the boosting algorithm. Although their evaluation methodology had certain limitations, they reported an average error rate of 0.9%, while the ADA-MLP achieved an error rate of 0. Likewise, as with previous benchmarks, ADA-MLP outperformed their model.

Lastly, in table 7, the ADA-MLP benchmarked against several prior studies. As illustrated, ADA-MLP outperformed all eight studies cited [7, 15,17,18, 30-33]. Although many of these studies did not provide average classification accuracy, in this study the best results for specific stocks or datasets were selected. Notably, our approach achieved an average classification accuracy of 100%, a benchmark not reached by any of the referenced studies.

In conclusion, the benchmarks and comparative analyses presented in this section illustrate that ADA-MLP consistently outperforms its competitors, both within the WEKA framework and in previous studies.

Table 7: Benchmark ADA-MLP with the list of previous studies.

Stock or Dataset	CA%

No: 1 [15] **Model:** Adaboost-diff-Versions

S&P 500	Discrete Ada (Expo. Loss) 66.57
S&P 500	Gentle Ada (Expo. Loss) 54.43
S&P 500	Real Ada (Expo. Loss) 68.4
S&P 500	Discrete Ada (Logis. Loss) 59.88
S&P 500	Gentle Ada (Logis. Loss) 56.22
S&P 500	Real Adat (Logis. Loss) 61.20
Description: Utilize functional forms of diverse classifiers to forecast financial time series data and assess the efficacy of various classifiers.	

No: 2 [17] **Model:** Adaboost

FX trading	67.3
Description: use AdaBoost algorithm to optimize	weight of technical indicators

No: 3 [32] **Model:** Adaboost

Electricity	16.67
Fixed Line Telecommunication	100
Automobile and Parts	55.13
Financial Services	89.58
Chemicals	78.88
Description: applied and compared salient machine learning algorithms to predict stock exchange volume	

No: 4 [33] **Model:** Advanced AdaBoost Algorithm

Shanghai A-share Stock	57.25
Description: establish a multi-factor stock selection model based on actor stock selection model based on AdaBoost algorithm and optimize the actual characteristics of stock selecting process	

No: 5 [34] **Model:** SVRvoting

Shanghai A-Share Stock	90.3 Best result chosen
Description: Based on three classififers (SVM, RF, ADA)	

No: 6 [35] **Model:** eXtreme Gradient Boosting (XGBoost)

Apple Inc.	87.99
Yahoo Inc.	99.91
Description: XGBoost is essentially an ensemble of weak classifier decision trees that primarily aims to train fresh decision trees to rectify the faults made by preceding trees.	

No: 7 [7] **Model:** GBDT (Gradient Boosting Decision Tree) and DE (Differential Evolution)

China stock market	88.5 Best result selected
Description: The suggested method utilizes GBDT for training, with the starting parameters of GBDT optimized by DE. Ultimately, out-of-sample data are identified using the trained GBDT–DE model, and its performance is evaluated.	

No: 8 [18] **Model:** Replicable implementation of an adaptive boosting (AdaBoost)

DDA data	82%
Description: Replicable execution of an adaptive boosting (AdaBoost) model that forecasts the probability of DDA titles being activated for acquisition.	

5. Conclusions

This study proposed an enhanced version of AdaBoostM1 for stock market price prediction using historical data to capture monthly trends. The improved model, ADA-MLP, replaces the traditional decision stump used in WEKA with MLP as the base learner, effectively integrating weak learners to improve performance. ADA-MLP demonstrated significant superiority over the original AdaBoostM1, achieving a 1.52% higher accuracy on average. Furthermore, it outperformed other modified versions of AdaBoost reported in the literature.

Building on these findings, multiple avenues for future research and practical applications are proposed. First, ADA-MLP could be tested with daily and weekly data to evaluate its predictive power across different timeframes, including intraday predictions for short-term trading strategies. Second, the methodology could be scaled to larger datasets, such as those spanning 50 years instead of 10, to examine its robustness over extended periods. Third, further performance enhancements could be achieved by optimizing hyperparameters or incorporating additional external features, such as macroeconomic indicators or sentiment analysis. Finally, ADA-MLP could be adapted for different types of financial markets, including commodities, currencies, and cryptocurrencies, to broaden its applicability.

For real-world implementation, ADA-MLP may be integrated into the WEKA project as a novel classifier, offering practitioners a powerful tool for stock market analysis. This integration could facilitate its adoption by researchers and financial analysts, paving the way for further advancements in financial forecasting.

Author contributions: Rebwar Mala Nabi: Conceptualization, Investigation, Methodology, Project administration, Writing – original draft. **Soran AB. Saeed:** Writing – review & editing. **Habibollah Haron:** Writing – review & editing.

Data availability: Data will be available upon reasonable request.

Conflicts of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding: The authors did not receive support from any organization for the submitted work.

References

[1] O. Bustos and A. Pomares-Quimbaya, "Stock market movement forecast: A Systematic review," *Expert Syst. Appl.*, vol. 156, p. 113464, Oct. 2020. doi: 10.1016/J.ESWA.2020.113464.

[2] V. Bolón-Canedo and A. Alonso-Betanzos, "Ensembles for feature selection: A review and future trends," *Inf. Fusion*, vol. 52, pp. 1–12, 2019. doi: 10.1016/j.inffus.2018.11.008.

[3] D. Kumar, P. K. Sarangi, and R. Verma, "A systematic review of stock market prediction using machine learning and statistical techniques," *Mater. Today Proc.*, Jan. 2021. doi: 10.1016/J.MATPR.2020.11.399.

[4] A. Thakkar and K. Chaudhari, "Applicability of genetic algorithms for stock market prediction: A systematic survey of the last decade," *Comput. Sci. Rev.*, vol. 53, p. 100652, Aug. 2024. doi: 10.1016/J.COSREV.2024.100652.

[5] K. Bisht and S. Kumar, "Hesitant fuzzy set based computational method for financial time series forecasting," *Granul. Comput.*, vol. 4, no. 4, pp. 655–669, Oct. 2019. doi: 10.1007/s41066-018-00144-4.

[6] V. Chang, T. Li, and Z. Zeng, "Towards an improved Adaboost algorithmic method for computational financial analysis," *J. Parallel. Distrib. Comput.*, vol. 134, pp. 219–232, Dec. 2019. doi: 10.1016/j.jpdc.2019.07.014.

[7] S. Deng, C. Wang, M. Wang, and Z. Sun, "A gradient boosting decision tree approach for insider trading identification: An empirical model evaluation of China stock market," *Appl. Soft Comput.*, vol. 83, p. 105652, 2019. doi: 10.1016/j.asoc.2019.105652.

[8] E. Egrioglu, U. Yolcu, and E. Bas, "Intuitionistic high-order fuzzy time series forecasting method based on pi-sigma artificial neural networks trained by artificial bee colony," *Granul. Comput.*, vol. 4, no. 4, pp. 639–654, Oct. 2019. doi: 10.1007/s41066-018-00143-5.

[9] M. Rashidpoor Toochaei and F. Moeini, "Evaluating the performance of ensemble classifiers in stock returns prediction using effective features," *Expert. Syst. Appl.*, vol. 213, p. 119186, Mar. 2023. doi: 10.1016/J.ESWA.2022.119186.

[10] E. Frank, M. A. Hall, and I. H. Witten "WEKA Workbench Online Appendix for 'Data mining: Practical machine learning tools and techniques,'" 4th Ed. 2016., Morgan Kaufmann, Burlington.

[11] Y. Freund, R. E. Schapire, and others, "Experiments with a new boosting algorithm," in *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning.* 1996, pp. 148–156.

[12] Y. Sun, S. Mutalib, N. Omar, and M. Huang, "Research on the application of LSTM-SA-AdaBoost hybrid model in stock forecasting," *Proceedings - 2023 5th International Conference on Applied Machine Learning, ICAML 2023*, pp. 310–315, 2023, doi: 10.1109/ICAML60083.2023.00064.

[13] Z. Dai, X. Zhang, and T. Li, "Forecasting stock return volatility in data-rich environment: A new powerful predictor," *N. Am. J. Econ. Financ.*, vol. 64, p. 101845, Jan. 2023. doi: 10.1016/J.NAJEF.2022.101845.

[14] Z. Guoying and C. Ping, "Forecast of yearly stock returns based on adaboost integration algorithm," in *Proceedings - 2nd IEEE International Conference on Smart Cloud, Smart Cloud 2017*, Institute of Electrical and Electronics Engineers Inc., Nov. 2017, pp. 263–267. doi: 10.1109/SmartCloud.2017.49.

[15] M. Park, M. L. Lee, and J. Lee, "Predicting stock market indices using classification tools," *Asian Econ. Financ. Rev.*, vol. 9, no. 2, pp. 243–256, 2019. doi: 10.18488/journal.aefr.2019.92.243.256.

[16] Y. Guo, J. Guo, B. Sun, J. Bai, and Y. Chen, "A new decomposition ensemble model for stock price forecasting based on system clustering and particle swarm optimization," *Appl. Soft Comput.*, vol. 130, p. 109726, Nov. 2022. doi: 10.1016/J.ASOC.2022.109726.

[17] Y. Wu, J. Mao, and W. Li, "Predication of futures market by using boosting algorithm," in *2018 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Mar. 2018, pp. 1–4. doi: 10.1109/WiSPNET.2018.8538586.

[18] K. W. Walker and Z. Jiang, "Application of adaptive boosting (AdaBoost) in demand-driven acquisition (DDA) prediction: A machine-learning approach," *J. Acad. Librariansh.*, vol. 45, no. 3, pp. 203–212, May 2019. doi: 10.1016/j.acalib.2019.02.013.

[19] R. M. Nabi, S. Ab. M. Saeed, and H. Harron, "A Novel approach for stock price prediction using gradient boosting machine with feature engineering (GBM-wFE)," *Kurdistan Journal of Applied Research*, vol. 5, no. 1, pp. 28–48, Apr. 2020. doi: 10.24017/SCIENCE.2020.1.3.

[20] M. H. D. M. Ribeiro and L. dos S. Coelho, "Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series," *Appl. Soft Comput.*, vol. 86, p. 105837, 2020. doi: 10.1016/j.asoc.2019.105837.

[21] Y. H.-F. Seng Jia-Lang, "The association between stock price volatility and financial news – a sentiment analysis approach," *Kybernetes*, vol. 46, no. 8, pp. 1341–1365, Jan. 2017. doi: 10.1108/K-11-2016-0307.

[22] R. T. F. Nazário, J. L. e Silva, V. A. Sobreiro, and H. Kimura, "A literature review of technical analysis on stock markets," *Q. Rev. Econ. Financ.*, vol. 66, pp. 115–126, 2017. doi: 10.1016/j.qref.2017.01.014.

[23] J. Sun, H. Li, H. Fujita, B. Fu, and W. Ai, "Class-imbalanced dynamic financial distress prediction based on Adaboost-SVM ensemble combined with SMOTE and time weighting," *Inf. Fusion*, vol. 54, pp. 128–144, Feb. 2020. doi: 10.1016/j.inffus.2019.07.006.

[24] S. Sun, Y. Wei, and S. Wang, "AdaBoost-LSTM ensemble learning for financial time series forecasting," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018. doi: 10.1007/978-3-319-93713-7_55.

[25] C. Wang and X. Bai, "Boosting Learning Algorithm for Stock Price Forecasting," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 322, p. 52053, Mar. 2018. doi: 10.1088/1757-899x/322/5/052053.

[26] K. Kang and J. Michalak, "Enhanced version of AdaBoostM1 with J48 tree learning method," 2018, [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1802/1802.03522.pdf.

[27] A. S. Wafi, H. Hassan, and A. Mabrouk, "Fundamental analysis models in financial markets – review study," *Procedia Econ. Financ*, vol. 30, pp. 939–947, 2015. doi: 10.1016/S2212-5671(15)01344-1.

[28] R. M. Nabi, S. A. M. Saeed, H. Bin Harron, and H. Fujita, "Ultimate prediction of stock market price movement," *J. Comput. Sci.*, vol. 15, no. 12, pp. 1795–1808, Dec. 2019. doi: 10.3844/JCSSP.2019.1795.1808.

[29] I. K. Nti, A. F. Adekoya and B. A. Weyori "A systematic review of fundamental and technical analysis of stock market predictions," *Artif. Intell. Rev.*, Aug. 2019. doi: 10.1007/s10462-019-09754-z.

[30] P. Chullamonthon and P. Tangamchit, "Ensemble of supervised and unsupervised deep neural networks for stock price manipulation detection," *Expert Syst. Appl.*, vol. 220, p. 119698, Jun. 2023. doi: 10.1016/J.ESWA.2023.119698.

[31] J. Sun, H. Li, H. Fujita, B. Fu, and W. Ai, "Class-imbalanced dynamic financial distress prediction based on Adaboost-SVM ensemble combined with SMOTE and time weighting," *Inf. Fusion*, vol. 54, pp. 128–144, Feb. 2020. doi: 10.1016/J.INFFUS.2019.07.006.

[32] M. A. Ghazanfar, Y. Alahmari Saad Ali amd Fahad Aldhafiri, M. Mustaqeem Anam amd Maqsood, and M. Awais Azam, "Using machine learning classifiers to predict stock exchange index," *Int. J. Mach. Learn.*, vol. 7, no. 2, pp. 24–29, Jan. 2017. doi: 10.18178/ijmlc.2017.7.2.614.

[33] S. Yutong and H. Zhao, “Stock selection model based on advanced AdaBoost algorithm,” in *2015 7th International Conference on Modelling, Identification and Control (ICMIC)*, Dec. 2015, pp. 1–7. doi: 10.1109/ICMIC.2015.7409386.
[34] Z. Sun, A. Harit, A. I. Cristea, J. Wang, and P. Lio, “MONEY: Ensemble learning for stock price movement prediction via a convolutional network with adversarial hypergraph model,” *AI Open*, vol. 4, pp. 165–174, Jan. 2023. doi: 10.1016/J.AIOPEN.2023.10.002.
[35] S. Dey, Y. Kumar, S. Saha, and S. Basak, “Forecasting to Classification: Predicting the direction of stock market price using Xtreme Gradient Boosting Forecasting to Classification: Predicting the direction of stock market price using Xtreme Gradient Boosting,” no. October, pp. 1–10, 2016. doi: 10.13140/RG.2.2.15294.48968.