

Stock Market Decision Support Modeling with Tree-Based Adaboost Ensemble Machine Learning Models

Ernest Kwame Ampomah, Zhiguang Qin, Gabriel Nyame and Francis Effirm Botchey

School of Information & Software Engineering, University of Electronic Science and Technology of China, China

Email: ampomahke@gmail.com, qinzg@uestc.edu.cn, kwakuasane1972@gmail.com, botcheyfrancis@gmail.com

Keywords: AdaBoost, machine learning, stock market, features, tree-based ensemble models

Received: May 10, 2020

Forecasting stock market behavior has received tremendous attention from investors and researchers for a very long time due to its potential profitability. Predicting stock market behavior is regarded as one of the extremely challenging applications of time series forecasting. While there is divided opinion on the efficiency of markets, numerous empirical studies which are widely accepted have shown that the stock market is predictable to some extent. Statistical based methods and machine learning models are used to forecast and analyze the stock market. Machine learning (ML) models typically perform better than those of statistical and econometric models. In addition, performance of ensemble ML models is typically superior to those of individual ML models. In this paper, we study and compare the efficiency of tree-based ensemble ML models (namely, Bagging classifier, Random Forest (RF), Extra trees classifier (ET), AdaBoost of Bagging (ADA_of_BAG), AdaBoost of RandomForest (ADA_of_RF), and AdaBoost of ExtraTrees (ADA_of_ET)). Stock data randomly collected from three different stock exchanges were used for the study. Forty technical indicators were computed and used as input features. The data set was split into training and test sets. The performance of the models was evaluated with the test set using accuracy, precision, recall, F1-score, specificity and AUC metrics. Kendall W test of concordance was used to rank the performance of the different models. The experimental results indicated that AdaBoost of Bagging (ADA_of_BAG) model was the highest performer among the tree-based ensemble models studied. Also, boosting of the bagging ensemble models improved the performance of the bagging ensemble models.

Povzetek: Z Adaboost algoritmi na osnovi dreves je analizirano dogajanje na borzah.

1 Introduction

Forecasting stock market behavior has received tremendous attention from investors, and researchers for a very long time due to its potential profitability (Bacchetta, et al, 2009; Campbell & Hamao, 1992; Granger & Morgenstern, 1970; Lin, et al, 2009; Rajashree & Pradipta, 2016; Weng et, al, 2018). It offers investors the opportunity to be proactive and take decisions which are knowledge-driven in order to gain good returns on their investments with less risk. Predicting stock market behaviour is regarded as one of the extremely challenging applications of time series forecasting. The stock market is affected by factors, such as economic policies, government decrees, political situations, psychology of investors, and so on (Tan, et al, 2007). These factors make the market very dynamic, nonlinear and complex, nonparametric, and chaotic nature (Abu-Mostafa & Atiya, 1996). While there is divided opinion on the efficiency of markets, numerous empirical studies which are widely accepted have shown that the stock market is predictable to some extent (Bollerslev, et al, 2014; Chen, et, al, 2003; Feuerriegel, & Gordon, 2018; Kim, et al, 2011; Phan, et, al, 2015). Statistical based methods and machine learning models are used to forecast and analyze the stock market. The statistical based approaches are not able to predict the stock market very well due the chaotic, noisy and

nonlinear in nature of the market. Contrary to statistical approaches, machine learning methods are able deal with the dynamic, chaotic, noisy, and nonlinear data of the stock market and have been widely used for a more accurate forecasting of stock market (Enke & Mehdiyev 2013; Hsu, et al, 2016; Meesad & Rasel, 2013; Thawornwong & Enke 2004; Rather et al. 2015). From the literature, application of machine learning models in stock market prediction can be grouped into a.) application of individual/single machine learning (ML) models (Alkhatib, et al, 2013; Chong et al, 2017; Guresen, et al, 2011; Khansa & Liginlal 2011; Meesad & Rasel 2013; Patel et al. 2015a; Tsai & Hsiao 2010; Wang, et al, 2011; Zhang & Wu 2009). b.) application of ensemble machine learning models. (Araújo, et al, 2015; Booth, et al, 2014; Chen, et al, 2007; Hassan, et al, 2007; Patel et al. 2015b; Rather et al, 2015; Wang, et al, 2012; Wang, et al, 2015). The ensemble models create several individual models to make predictions and then aggregate the outcomes of each individual model to make a final prediction. The performance of ensemble models is better than that of individual models as the ensemble models reduce the generalization error of the predictions. The dominance of ensemble models over individual models has been demonstrated in the field of financial expert systems (Chen et al., 2007; Haung et al, 2008; Tsai et al., 2011). Hence, in this work, we study and compare the effectiveness of tree-based bagging ensemble machine

learning models and the impact of Boosting on the tree-based bagging ensemble models. Specifically, the study compares the effectiveness of the following classifiers: Random forest classifier (RF), Bagging classifier (BAG), and Extra trees classifier (ET), AdaBoost of RandomForest classifier (ADA_of_RF) model, AdaBoost of Bagging classifier (ADA_of_BAG) model and AdaBoost of ExtraTrees classifier (ADA_of_ET) models in forecasting one-day ahead stock price movement.

2 Related studies

There have been a number of research studies on forecasting stock market behavior with machine learning algorithms. In this section, we provide a review of some of these studies. Tsai, et al, (2011) studied the performance of ensemble classifiers in analyzing stock returns. They considered the hybrid approaches of majority voting and bagging. They compared the performance homogeneous and heterogeneous ensemble classifiers with those of single baseline classifiers (decision trees, neural networks, and logistic regression). The experimental results indicated that ensemble classifiers outperformed the single classifiers in terms of prediction. In terms of prediction accuracy, there was no significant difference between majority voting and bagging, however, the majority voting had better stock returns than the bagging. Finally, the homogeneous neural networks ensemble classifiers produced the best performance by majority voting when predicting stock returns. Huang et al, (2008) applied wrapper approach to select subset of optimal features from the initial feature set of 23 technical indices and then employed an ensemble voting scheme that combines different classifiers to forecast the trend in Korea and Taiwan stock markets. Experimental outcome shows that the wrapper approach is able to produce better performance than the commonly used features filters, including χ^2 Statistic, Information gain, ReliefF, Symmetrical uncertainty and CFS. In addition, the proposed ensemble voting scheme performed better than the single classifier such as SVM, kth nearest neighbor, back-propagation neural network, decision tree, and logistic regression. Lunga & Marwala, (2006) investigated the predictability of direction of movement of stock market with Learn++ algorithm by predicting the daily movement direction of the Dow Jones. The Learn++ algorithm is derived from the AdaBoost algorithm. The framework was implemented with multi-layer Perceptron (MLP) as a weak Learner. Initially, a weak learning algorithm, which attempts to learn a class concept with a single input Perceptron, is established. The Learn++ algorithm is applied to improve the learning capacity of the weak MLP and introduces the concept of online incremental learning. The proposed framework can adapt as new data are introduced and is able to classify. Balling et al, (2015) compared the performance of ensemble classifier models (Random Forest, AdaBoost and Kernel Factory) against individual classifier models (Neural Networks, Logistic Regression, SVM, and K-Nearest Neighbor). They used data from 5767 publicly listed European companies and AUC metric to evaluate the

models. The experimental results indicated that Random Forest was the best performer with SVM, Kernel Factory, AdaBoost, Neural Networks, K-Nearest Neighbors and Logistic Regression following in that order. Nayak et al, (2016) made an attempt to predict stock market trend. Two models, one for daily prediction and the other for monthly prediction were built. Three supervised machine learning algorithms namely Decision Boosted Tree, Support Vector Machine, and Logistic Regression were used. With the daily prediction model, historical stock price data were combined with sentiment data. An accuracy of up to 70% were observed using the supervised machine learning algorithms on daily prediction model. It was observed that Decision Boosted Tree performed better than Support Vector Machine and Logistic Regression. The monthly prediction models were used to evaluate the similarity among any two different months trend. The evaluation demonstrated that trend of one month were least correlated with the trend of other months. Khan et al, (2020) employed machine learning algorithms on social media and financial news data to establish the influence of this data on stock market prediction accuracy for ten subsequent days. In order to improve performance and quality of predictions, the authors performed feature selection and spam tweets reduction on the data sets. In addition, experiments to determine stock markets that are difficult to predict and those that are more influence by social media and financial news. A comparison of results of different algorithms to find a consistent classifier was done. Deep learning is used and some classifiers are ensembled. The experimental outcome showed that highest prediction accuracies of 80.53% and 75.16% were attained using social media and financial news, respectively. Also, the results showed that, the New York and Red Hat stock markets are difficult to predict, the New York and IBM stocks are strongly influenced by social media, while London and Microsoft stocks are strongly influenced by financial news. Random forest classifier proved to be consistent and provided the highest accuracy of 83.22% by its ensemble. Nti et al, (2020), conducted a comparative analysis of ensemble machine learning techniques including boosting, bagging, blending and super learners (stacking). The authors build 25 different ensembled regressors and classifiers Using Decision Trees (DT), Support Vector Machine (SVM) and Neural Network (NN). A comparison of their execution times, accuracy, and error metrics over stock-data from Ghana Stock Exchange (GSE), Johannesburg Stock Exchange (JSE), Bombay Stock Exchange (BSE-SENSEX) and New York Stock Exchange (NYSE), from 2012 to 2018 was undertaken. The experimental results showed that stacking and blending ensemble techniques provide higher prediction accuracies (90–100%) and (85.7–100%) respectively, as compared with that of bagging (53–97.78%) and boosting (52.7–96.32%). Also, the root means square error obtained by stacking (0.0001–0.001) and blending (0.002–0.01) provided a better fit of ensemble classifiers and regressors based on these two techniques in market analyses in comparison with bagging (0.01–0.11) and boosting (0.01–0.443). The outcomes suggested that studies in the domain of stock market

direction prediction ought to include ensemble techniques in their sets of algorithms. Vijha et al, (2020) utilized artificial neural network and random forest techniques to predict the next day closing price for five companies which belong to different sectors of operation. The authors generated new variables which are used as inputs to the model from the financial data: Open, High, Low and Close prices of stocks. The evaluation of the models was done using standard RMSE and MAPE.

3 Method

The stock data were subjected to (i) data cleaning; to deal with the missing and erroneous values, (ii) data normalization; to ensure that, the machine learning models perform well. Each dataset was split into training and test sets for the purpose of this experiment. The training set was made up of the initial 70% of the data set, and the final 30% of the data set constituted the test set. Each model was trained with the training set and evaluated using the test set.

3.1 Data and features

For this research study, we randomly collected ten different stock data from three different stock markets (namely NYSE, NASDAQ, and NSE) through the yahoo finance API. The data from the following companies and indices are used: Apple Inc. ('AAPL'), Abbott Laboratories ('ABT'), Bank of America Corp ('BAC'), Exxon mobile corporation ('XOM'), S&P_500 Index, Microsoft Corporation ('MSFT'), Dow Jones Industrial Average Index ('DJIA'), CarMax Inc. ('KMX'), Tata Steel Limited ('TATASTEEL'), and HCL Technologies Ltd ('HCLTECH'). Table 1 provides a description of the

Data Set	Stock Market	Time Frame	Number of Sample
AAPL	NASDAQ	2005-01-01 to 2019-12-30	3774
ABT	NYSE	2005-01-01 to 2019-12-30	3774
BAC	NYSE	2005-01-01 to 2019-12-30	3774
XOM	NYSE	2005-01-01 to 2019-12-30	3774
S&P_500	INDEXSP	2005-01-01 to 2019-12-30	3774
MSFT	NASDAQ	2005-01-01 to 2019-12-30	3774
DJIA	INDEXDJX	2005-01-01 to 2019-12-30	3774
KMX	NYSE	2005-01-01 to 2019-12-30	3774
TATASTEEL	NSE	2005-01-01 to 2019-12-30	3279
HCLTECH	NSE	2005-01-01 to 2019-12-30	3477

Table 1: Description of the data sets.

data sets used. To ensure generalizability of results, forty (40) technical indicators are computed from the original OHLCV data and used as input features. These technical indicators are selected from four categories of technical indicators which are volume indicators, price transform,

overlap studies, and momentum indicators. The details of these technical indicators are provided by table 10-13 in the appendix section.

3.2 Feature scaling

The input features have different range of values. Hence, we apply standardization scaling (z-score) to bring all the input features within the same range. The z-score centres values around the mean with a unit standard deviation. The scaling of input features assures that the larger value features do not overwhelm smaller value inputs, and also helps minimize the prediction errors (Kim, 2003).

$$z(x) = (x[:, i] - \mu_i) / \sigma_i \tag{1}$$

Where μ_i = mean of the *ith* feature, σ_i = standard deviation of the *ith* feature.

3.3 Machine learning algorithms

The study considered and compared the efficacy of Random forest classifier (RF), Bagging classifier (Bag), and Extra trees classifier (ET), AdaBoost of RandomForest (ADA_of_RF) model, AdaBoost of Bagging (ADA_of_BAG) model and AdaBoost of ExtraTrees (ADA_of_ET) in forecasting one-day ahead stock price movement. A discussion of these machine learning (ML) algorithms is presented here.

3.3.1 AdaBoost algorithm

AdaBoost is an ensemble/meta-learning approach that builds a strong classifier as a linear combination in an iterative way. In every iteration, it makes a call to a weak learning algorithm (the base learner) which returns a classifier, and gives a weight coefficient to it. AdaBoost tweaks subsequent base learners in favor of those instances misclassified by preceding classifiers. The outcome of the weak learners is aggregated into a weighted sum that represents the final outcome of the boosted classifier. The final output of the boosted classifier is decided by a weighted "vote" of the base classifiers. The smaller the error of the base classifier, the larger is its weight in the final vote (Freund & Schapire, 1996). AdaBoost is sensitive to outliers and noisy data. AdaBoost ML algorithm is given by algorithm 1 below.

3.3.2 Decision tree algorithm

Decision tree is a hierarchical tree structure that is used to determine the class label of instances based on a series of if-then rules about the features /attributes of the class. A decision tree consists of nodes (root, internal, and leaf), and branches. The root and internal nodes specify a test condition on a feature, each branch represents one of the possible values of the feature, and each leaf node contains a class label. To classify an instance, we start from the root node and apply the test condition to the instance and follow the branch with the value corresponding to the test outcome. This will take us to either an internal node, for which another test condition is executed, or to a leaf node.

Input

Given instances: $(x_1, y_1) \dots (x_m, y_m)$; $x \rightarrow X$, with labels $y_i \in Y = \{-1, +1\}$

Initialize: $D_t(i) = \frac{1}{m}$ for $i = 1, \dots, m$.

for $t = 1, \dots, T$:

1. Call and train a weak learner which returns the weak classifier $h_t : X \rightarrow \{-1, 1\}$ with minimum error with respect to distribution D_t

2. Compute the error of h_t :

$$\varepsilon_t = P_{r \sim D_t} [h_t(x_i) \neq y_i]$$

3. Select $\beta_t = \frac{1}{2} \ln \left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right)$

4. Update the distribution

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\beta_t y_i h_t(x_i))}{Z_t}$$

where Z_t a normalization constant is chosen such that

D_{t+1} is a distribution

Output:

The final hypothesis: $H(x) = \text{sign} \left(\sum_{t=1}^T \beta_t h_t(x) \right)$

Algorithm 1: AdaBoost ML algorithm (Freund & Schapire, 1996).

The class label contained in the leaf node is assigned to the instance (Rokach & Maimon, 2008).

3.3.3 Bagging algorithm

A Bagging classifier is an ensemble classifier which generates multiple base learners (decision tree) and fits each of these base learners on random subsets of the initial dataset and then combine their individual predictions (through voting or averaging) to produce a final prediction. All the base learners are trained in parallel with the new training sets which are generated by randomly drawing N samples with replacement from the original training dataset – where N is the size of the original training set. The training set for each base learner is independent of the one another. Since the training set for each base learner is generated by resampling initial training data set with replacement, some instances may appear many times while others may not appear. If perturbing the training set can cause significant changes in the models built, then bagging can increase accuracy (Breiman, 1996). Bagging is less sensitivity to outliers and noise, and has a parallel structure for efficient implementations. It is a technique that reduces the variance of an estimated prediction function.

3.3.4 Random forest algorithm

Random Forest constructs an ensemble of de-correlated trees and aggregates them to improve upon the robustness and performance of the decision trees (Breiman, 2001). Each tree is trained with a bootstrap sample from the original training data. In addition, a subset of features is selected randomly from the full set of original features to grow the tree at each node. To establish the class label of a new instance, each decision tree delivers a class label for this instance, and random forest then aggregates the class labels predicted and selects the most voted prediction as the label for the new instance. Since RF searches for the best feature among a random subset of features, it leads to a wide diversity that generally produce a better model. RF can handle larger input datasets.

3.3.5 Extra trees algorithm

Extra trees algorithm is a tree-based ensemble machine learning algorithm. ET constructs an ensemble of base learners (decision trees) using the classical top-down procedure. The predictions of all the trees are combined to generate the final prediction through majority vote. ET is similar to RF in that it constructs the trees and split nodes with random subsets of features. However, ET differs from RF on two main counts which are (i) ET uses the entire training data to grow the trees (instead of a bootstrap replica). (ii) ET splits nodes by selecting split-points fully at random. The randomization of the cut-point and features together with ensemble averaging reduces variance while the use of the entire original training sample minimizes bias (Geurts, et al, 2006). ET is computationally efficient.

3.4 Hyperparameter optimization

Machine learning algorithms have a set of hyperparameters, and these hyperparameters determine how the model is structured. Our aim is to find the right combination of values for these hyperparameters which will ensure that the machine learning models perform at their best. In this work, we set the hyperparameters of the various machine learning algorithms using Bayesian hyperparameter optimization technique (Feurer & Hutter, 2019). Bayesian hyperparameter optimization (BHO) is an iterative technique which has two basic ingredients: a probabilistic surrogate model and an acquisition function to choose the next point to evaluate. In each iteration, the surrogate model is trained on all observations of the target function made so far. The acquisition function then determines the usefulness of various candidate points, trading off exploration and exploitation. It is much cheaper to compute the acquisition function than to evaluate the blackbox function. Therefore, BHO provides an efficient and cheap way to select good hyperparameter for ML models (Bergstra et al, 2011).

Data Sets	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET
AAPL	0.9065	0.8982	0.8861	0.9093	0.9019	0.8824
ABT	0.8232	0.8898	0.8852	0.8889	0.8963	0.8843
KMX	0.9176	0.9167	0.8889	0.9139	0.9102	0.8722
S&P_500	0.9111	0.9019	0.8852	0.9157	0.9046	0.8926
TATASTEEL	0.9442	0.9378	0.9067	0.9378	0.9356	0.9088
HPCL	0.9203	0.9193	0.9021	0.9294	0.9203	0.8981
BAC	0.9028	0.8870	0.8704	0.9065	0.8917	0.8917
Mean	0.9037	0.9072	0.8892	0.9145	0.9087	0.8900

Table 2: Accuracy Scores of the tree-based ensemble models.

3.5 Evaluation metric

The following classical quality evaluation metrics are used to evaluate the performance of the tree-based AdaBoost ensemble ML models: (a) Accuracy, (b) Precision, (c) Recall, (d) F-measure, (e) Specificity, (f) Area under receiver operating characteristics curve (AUC-ROC).

Accuracy: measures the overall number of predictions that the model gets right

$$accuracy = \frac{tp+tn}{tp+tn+fp+fn} \tag{2}$$

F1-score: provides a harmonic mean of precision and

$$F1_score = \frac{2 \times precision \times recall}{precision + recall} \tag{3}$$

Specificity: assesses how well the classifier is able to identify negative instances.

$$specificity = \frac{tn}{tn+fp} \tag{4}$$

Where tp = true positive, fp = false positive, tn = true negative, and fn = false negative

ROC curve: shows the trade-off between true positive to false positive rates.

AUC: it tells a model’s ability to discriminate between positive and negative instances. The worst AUC is 0.5, and the best AUC is 1.0.

4 Results and discussion

The performances of the different tree-based ensemble ML models on the stock data sets are summarized and discussed in this section.

Table 2 displays the accuracy results of the tree-based ensemble models on the various stock data. From this table, the accuracy values of ADA_of_BAG was the best on AAPL, S&P_500, BAC and HPCL stock data sets. Similarly, Bag recorded the highest accuracy values on KMX and TATASTEEL stock data sets. ADA_of_RF obtained the highest accuracy value on the ABT data set.

Data Sets	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET
AAPL	0.9130	0.9060	0.8928	0.9160	0.9080	0.8881
ABT	0.8210	0.8996	0.8944	0.8936	0.9038	0.8914
KMX	0.9190	0.9185	0.8911	0.9154	0.9125	0.8727
S&P_500	0.9184	0.9099	0.8901	0.9214	0.9123	0.8988
TATASTEEL	0.9448	0.9387	0.9085	0.9387	0.9363	0.9091
HPCL	0.9217	0.9205	0.9046	0.9303	0.9209	0.9003
BAC	0.9077	0.8939	0.8772	0.9119	0.8992	0.8992
Mean	0.9065	0.9124	0.8941	0.9182	0.9133	0.8942

Table 3: F1 Scores of the tree-based ensemble models.

Data Sets	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET
AAPL	0.8926	0.8748	0.8847	0.8907	0.8966	0.8926
ABT	0.9002	0.8543	0.8603	0.9102	0.8822	0.8822
KMX	0.9328	0.9271	0.9002	0.9290	0.9156	0.9002
S&P_500	0.9080	0.8978	0.9284	0.9325	0.9018	0.9182
TATASTEEL	0.9457	0.9348	0.8978	0.9348	0.9348	0.8978
HPCL	0.9255	0.9275	0.8986	0.9400	0.9358	0.8986
BAC	0.8660	0.8377	0.8302	0.8604	0.8321	0.8321
Mean	0.9101	0.8934	0.8857	0.9139	0.8998	0.8888

Table 4: Specificity Scores of the tree-based ensemble models.

DataSets	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET
AAPL	0.9648	0.9645	0.9562	0.9665	0.9633	0.9469
ABT	0.9263	0.9453	0.9564	0.9645	0.9578	0.9516
KMX	0.9756	0.9677	0.9558	0.9750	0.9662	0.9453
S&P_500	0.9548	0.9646	0.9623	0.9708	0.9684	0.9608
TATASTEEL	0.9832	0.9814	0.9730	0.9821	0.9809	0.9725
HPCL	0.9766	0.9726	0.9704	0.9792	0.9722	0.9671
BAC	0.9644	0.9584	0.9507	0.9716	0.9660	0.9512
Mean	0.9637	0.9649	0.9607	0.9728	0.9678	0.9565

Table 5: AUC Scores of the tree-based ensemble models.

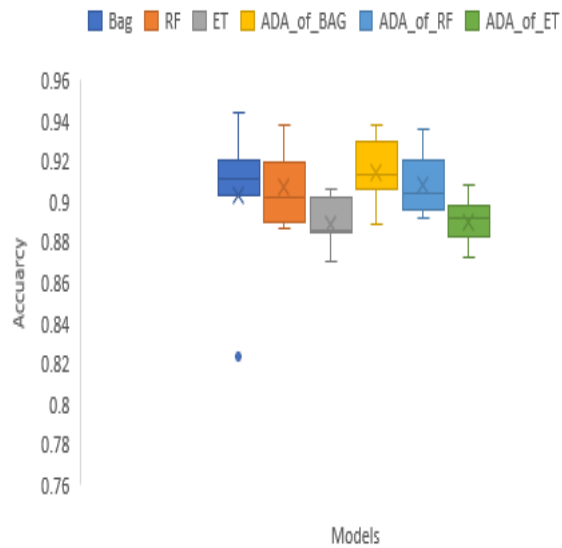


Figure 1: Boxplot of accuracy results of the tree-based ensemble models on the test datasets.

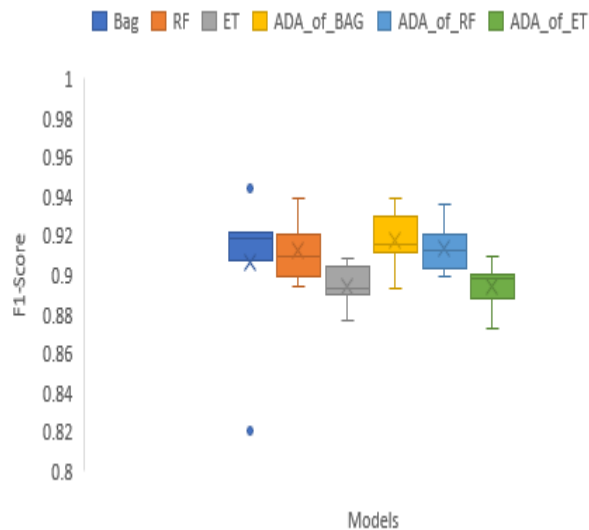


Figure 2: Boxplot of F1-Scores of the tree-based ensemble models on the test datasets.

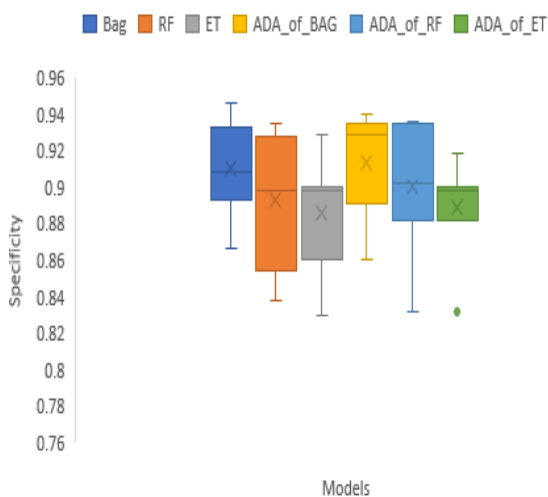


Figure 3: Boxplot of Specificity of the tree-based ensemble models on the test datasets.

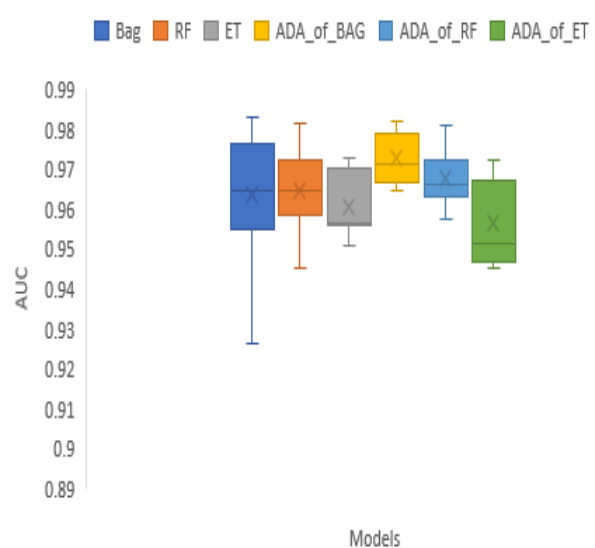


Figure 4: Boxplot of AUC results of the tree-based ensemble models on the test datasets.

Overall, the mean accuracy value of ADA_of_BAG was the best among all the tree-based ensemble algorithms. Boosting of the bagging algorithms (ADA_of_BAG, ADA_of_RF and ADA_of_ET) improved the mean accuracy values of their respective bagging algorithms (Bag, RF and ET). Figure 1 presents the box plot of the accuracy values of the various models.

Table 3 presents the F1-Scores of the tree-based ensemble models on the various stock data. ADA_of_BAG obtained the highest F1-Score on AAPL, S&P_500, BAC and HPCL stock data sets. Also, Bag recorded the highest accuracy values on KMX and TATASTEEL stock data sets. ADA_of_RF achieved the best F1-Score on the ABT stock data set. In general, the mean F1-value of ADA_of_BAG was the best among all the tree-based ensemble algorithms. In addition, boosting of the bagging algorithms (ADA_of_BAG, ADA_of_RF and ADA_of_ET) improved the mean F1 values of their respective base bagging algorithms (Bag, RF and ET). Figure 2 presents the box plot of the F1-Scores of the various models.

Table 4 shows the specificity results of the tree-based ensemble models on the various stock data. ADA_of_BAG had the highest specificity on ABT, S&P_500 and HPCL stock data sets. Also, Bag obtained the highest specificity on KMX, TATASTEEL and BAC stock data sets. ADA_of_RF achieved the highest specificity on the ABT stock data set. The mean specificity value of ADA_of_BAG was the best among all the tree-based ensemble algorithms. Moreover, boosting of the bagging algorithms (ADA_of_BAG, ADA_of_RF and ADA_of_ET) improved the mean specificity results of their respective base bagging algorithms (Bag, RF and ET). Figure 3 presents the box plot of the specificity results of the various models.

Table 5 presents the AUC results of the tree-based ensemble models on the various stock data. ADA_of_BAG performed better than the other models on AAPL, ABT, S&P_500, BAC and HPCL stock data sets.

Similarly, the performance of Bag was higher than the other models on KMX and TATASTEEL stock data sets. In general, the mean AUC of ADA_of_BAG was the best among all the tree-based ensemble algorithms. In addition, boosting of the bagging algorithms Bag and RF (ADA_of_BAG and ADA_of_RF) recorded a better mean AUC value than their respective base bagging algorithms (Bag and RF). Figure 4 shows the box plot of the AUC results of the various models.

Figure 5-11 shows the ROC curves of all the tree-based ensemble models considered in this study on the AAPL, ABT, KMX, S&P_500, TATASTEEL, HPCL and BAC stock data sets respectively.

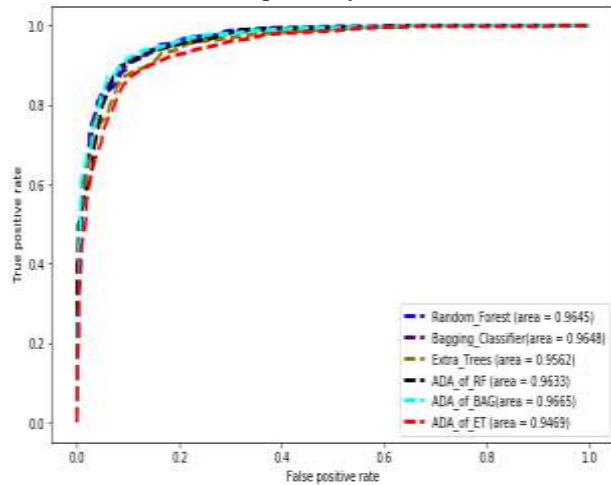


Figure 5: ROC curve of the tree-based ensemble models on AAPL stock data set.

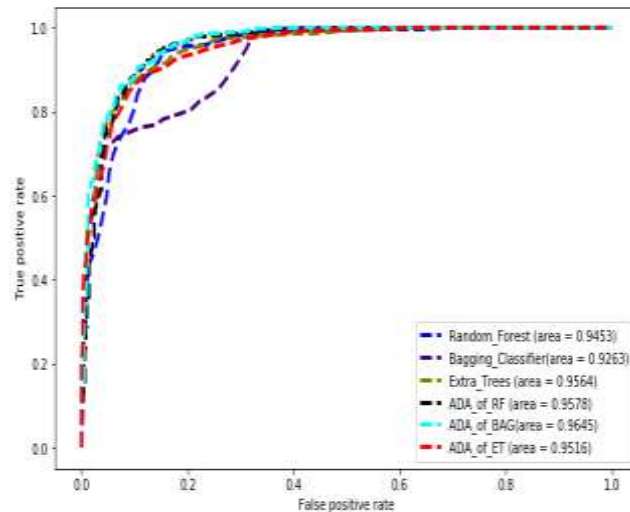


Figure 6: ROC curve of the tree-based ensemble models on ABT stock data set.

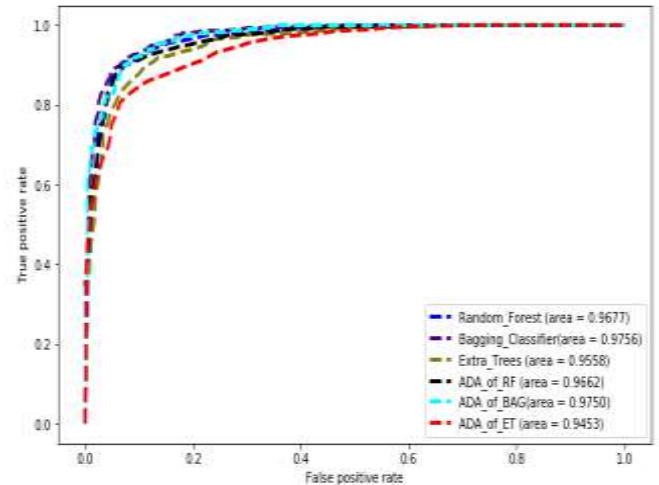


Figure 7: ROC curve of the tree-based ensemble models on KMX stock data set.

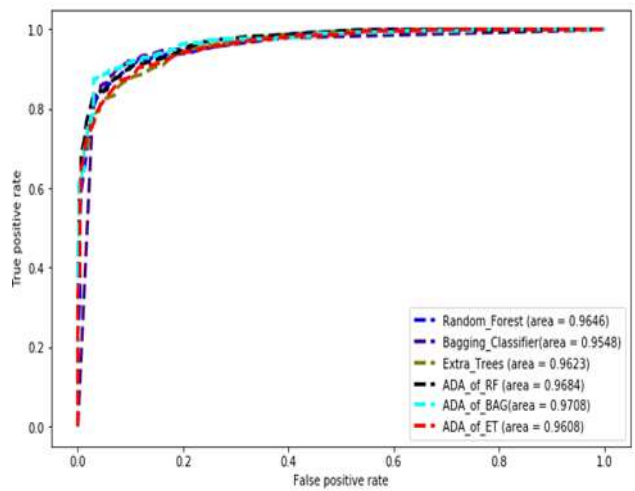


Figure 8: ROC curve of the tree-based ensemble models on S&P_500 stock data set.

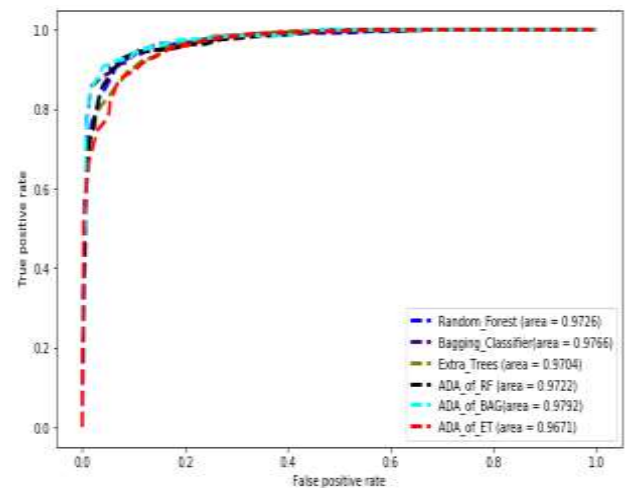


Figure 9: ROC curve of the tree-based ensemble models on HPCL stock data set.

Metric	W	χ^2	p	Ranks							
				Technique	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET	
Accuracy	0.61	21.29	0.00								
				Mean Rank	4.64	3.64	1.71	5.21	4.00	1.79	

Table 6: Kendall’s coefficient of concordance ranks of tree-based ensemble models using accuracy metric.

Metric	W	χ^2	p	Ranks							
				Technique	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET	
F1-Score	0.56	19.57	0.00								
				Mean Rank	4.71	3.64	1.86	5.07	3.93	1.79	

Table 7: Kendall’s coefficient of concordance ranks of tree-based ensemble models using F1-score metric.

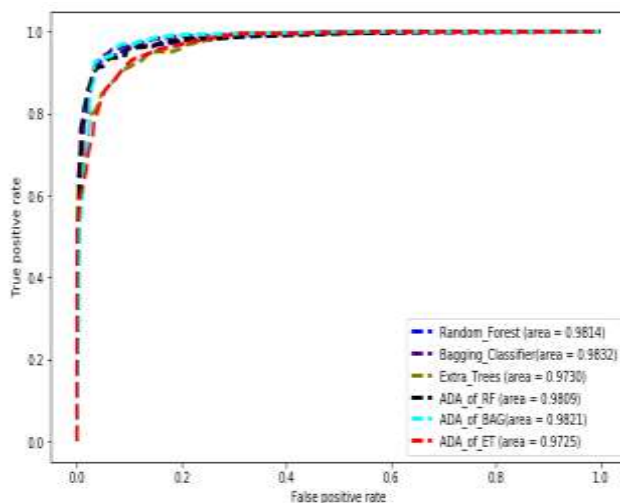


Figure 10: ROC curve of the tree-based ensemble models on TATASTEEL stock data set.

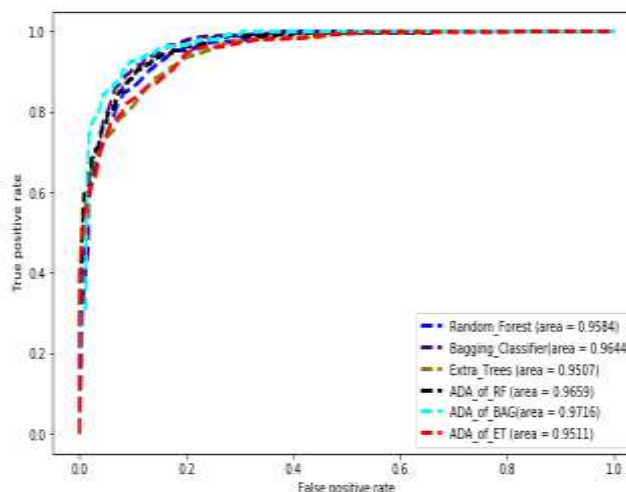


Figure 11: ROC curve of the tree-based ensemble models on S&P_500 stock data set.

The Kendall’s coefficient of concordance (W) is applied to rank the efficiency of the different tree-based AdaBoost ensemble models. This test is a measure that applies ranks to establish an agreement among raters (Kendall & Babington, 1939). It determines the agreement

among diverse raters who are evaluating a given set of n objects. Depending on the area where it is being applied, the raters can be variables, characters, and so on. The raters are the different data sets in this article. Kendall’s coefficient of concordance has been applied in many researches including Kendall’s Coefficient of Concordance for Sociometric Rankings with Self Excluded by Gordon et al, (1971), Use of Kendall’s coefficient of concordance to assess agreement among observers of very high-resolution imagery by Gearhart et al, (2013), Measuring and testing interdependence among random vectors based on Spearman’s ρ and Kendall’s τ by Zhang & Wang, (2020), In this study a cut-off value of 0.05 for the significance level (p-value) is used. The Kendall’s coefficient is considered to be significant and having the capability of giving an overall ranking when $p < 0.05$. At $p = 0.05$, the critical value of chi-square (χ^2) for five (5) degrees of freedom is 11.07. The degrees of freedom equal the total number of ML algorithms (which is six) minus one. The results of Kendall’s coefficient of concordance are given by tables 6-9 below using accuracy, precision, recall, F1-score, specificity, and AUC respectively.

Table 6 shows that Kendall’s coefficient using the accuracy metric is significant ($p < 0.05$, $\chi^2 > 11.07$) and that the performance of *ADA_of_BAG* model is the best among the ensemble methods. The overall ranking is *ADA_of_BAG* > *Bag* > *ADA_of_RF* > *RF* > *ADA_of_ET* > *ET*.

Table 7 presents that Kendall’s coefficient using the F1-Score metric is significant ($p < 0.05$, $\chi^2 > 11.07$) and the performance of *ADA_of_BAG* model is the best among the ML ensemble models. The overall ranking is *ADA_of_BAG* > *Bag* > *ADA_of_RF* > *RF* > *ET* > *ADA_of_ET*.

Table 8 demonstrates that Kendall’s coefficient using the specificity metric is significant ($p > 0.05$, $\chi^2 < 11.07$), and *ADA_of_BAG* had the highest rank. The overall ranking is *ADA_of_BAG* > *Bag* > *ADA_of_RF* > *RF* = *ADA_of_ET* > *ET*.

Table 9 demonstrates that Kendall’s coefficient using the AUC metric is significant ($p < 0.05$, $\chi^2 > 11.07$) and the performance of *ADA_of_BAG* model has the best rank

Metric	W	χ^2	p	Ranks						
Specificity	0.41	14.45	0.01	Technique	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET
				Mean Rank	4.79	2.71	2.07	5.00	3.71	2.71

Table 8: Kendall’s coefficient of concordance ranks of tree-based ensemble models using specificity metric.

Metric	W	χ^2	p	Ranks						
AUC	0.600	20.95	0.00	Technique	Bag	RF	ET	ADA_of_BAG	ADA_of_RF	ADA_of_ET
				Mean Rank	4.00	3.57	2.29	5.71	3.86	1.58

Table 9: Kendall’s coefficient of concordance ranks of tree-based ensemble models using AUC metric.

among the tree-based AdaBoost ML ensemble models. The overall ranking is **ADA_of_BAG > Bag > ADA_of_RF > RF > ET > ADA_of_ET**

5 Conclusion

This study compares the efficacy of tree-based of bagging ensemble machine learning models and boosting of tree-based bagging machine learning models in forecasting movement direction of stock prices. Seven randomly collected stock data from three different stock exchanges were used. The data sets were split into training and test sets. The performance of the models was evaluated using accuracy, F1-score, specificity, and AUC metrics on the test data set. Kendall W test of concordance was used to ranked the performance of the different models. The results indicated that boosting of tree-based bagging ensemble models, improves the performance of the bagging models. Overall, the performance of ADA_of_BAG model was superior to the remaining models used in the study. The limitation of this study is that it only considered bagging models and boosting of bagging models. Hence, future study will investigate boosting models and bagging of boosting models in predicting stock price behaviour.

6 Acknowledgement

This work was supported by the NSFC-Guangdong Joint Fund (Grant No. U1401257), National Natural Science Foundation of China (Grant Nos. 61300090, 61133016, and 61272527), science and technology plan projects in Sichuan Province (Grant No. 2014JY0172) and the opening project of Guangdong Provincial Key Laboratory of Electronic Information Products Reliability Technology (Grant No. 2013A061401003).

7 References

- [1] Abu-Mostafa, Y. S., & Atiya, A. F. Introduction to financial forecasting. *Applied Intelligence*, 6(3), 205–213, 1996. <https://doi.org/10.1007/bf00126626>
- [2] Alkhatib, K., Najadat, H., Hmeidi, I., & Shatnawi, M. K. A. Stock price prediction using k-nearest neighbor (knn) algorithm. *International Journal of Business, Humanities and Technology*, 3 (3), 32–44, 2013.
- [3] Ampomah, E., K., Qin Z., & Nyame, G. Evaluation of Tree-based Ensemble Machine Learning Models in Predicting Stock Price Direction of Movement, *information*, 11, 332, 2020. <https://doi.org/10.3390/info11060332>
- [4] Araújo, R. d. A., Oliveira, A. L., & Meira, S. A hybrid model for high-frequency stock market forecasting. *Expert Systems with Applications*, 42 (8), 4081–4096, 2015. <https://doi.org/10.1016/j.eswa.2015.01.004>.
- [5] Bacchetta, P., Mertens, E., & Van Wincoop, E. Predictability in financial markets: What do survey expectations tell us? *Journal of International Money and Finance*, 28 (3), 406–426, 2009. <https://doi.org/10.1016/j.jimonfin.2008.09.001>
- [6] Ballings, M., Van den Poel, D., Hespels, N., & Gryp, R. Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*, 42 (20), 7046–7056, 2015. <https://doi.org/10.1016/j.eswa.2015.05.013>
- [7] Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24, 2546–2554, 2011.
- [8] Bollerslev, T., Marrone, J., Xu, L., & Zhou, H. Stock return predictability and variance risk premia: Statistical inference and international evidence. *Journal of Financial and Quantitative Analysis*, 49 (03), 633–661, 2014. <https://doi.org/10.1017/s0022109014000453>
- [9] Booth, A., Gerding, E., & McGroarty, F. Automated trading with performance weighted random forests and seasonality. *Expert Systems with Applications*, 41(8), 3651–3661, 2014. <https://doi.org/10.1016/j.eswa.2013.12.009>
- [10] Breiman, L. Bagging predictors. *Mach Learn* 24, 123–140, 1996. <https://doi.org/10.1007/bf00058655>
- [11] Breiman, L. Random forests. *Machine learning*, 45(1), 5–32, 2001.
- [12] Campbell, J. Y., & Hamao, Y. Predictable stock returns in the united states and japan: A study of long-term capital market integration. *The Journal of Finance*, 47 (1), 43–69, 1992. <https://doi.org/10.1111/j.1540-6261.1992.tb03978.x>
- [13] Chen, A.-S., Leung, M. T., & Daouk, H. Application of neural networks to an emerging financial market: Forecasting and trading the taiwan stock index.

- Computers & Operations Research, 30 (6), 901–923, 2003.
[https://doi.org/10.1016/s0305-0548\(02\)00037-0](https://doi.org/10.1016/s0305-0548(02)00037-0)
- [14] Chen, Y., Yang, B., & Abraham, A. Flexible neural trees ensemble for stock index modeling. *Neurocomputing*, 70 (4), 697–703, 2007.
<https://doi.org/10.1016/j.neucom.2006.10.005>
- [15] Chong, E., Han, C., & Park, F. C. Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83, 187–205, 2017.
<https://doi.org/10.1016/j.eswa.2017.04.030>
- [16] Enke, D., & Mehdiyev, N. Stock market prediction using a combination of stepwise regression analysis, differential evolution-based fuzzy clustering, and a fuzzy inference neural network. *Intelligent Automation & Soft Computing*, 19 (4), 636–648, 2013.
<https://doi.org/10.1080/10798587.2013.839287>
- [17] Feurer M., Hutter F. Hyperparameter Optimization. In: Hutter F., Kotthoff L., Vanschoren J. (eds) *Automated Machine Learning*. The Springer Series on Challenges in Machine Learning. Springer, Cham, 2019.
- [18] Feuerriegel, S., & Gordon, J. Long-term stock index forecasting based on text mining of regulatory disclosures. *Decision Support Systems*, 112: 88–97, 2018.
<https://doi.org/10.1016/j.dss.2018.06.008>
- [19] Freund, Y., & Schapire, R. Experiments with a new boosting algorithm. In *machine learning. proceedings of the thirteenth international conference (ICML '96)*. 148–156. Bari, Italy, 1996.
- [20] Gearhart, A., Booth, D. T., Sedivec, K. & Schauer, C. Use of Kendall's coefficient of concordance to assess agreement among observers of very high-resolution imagery. *Geocarto International*, 28(6), 517–526, 2013. <https://doi.org/10.1080/10106049.2012.725775>.
- [21] Geurts P., Ernst, D., Wehenkel L. Extremely randomized trees, *Mach Learn*, 63: 3–42, 2006.
<https://doi.org/10.1007/s10994-006-6226-1>
- [22] Ghorbani, M., & Chong E., K., P. (2020), Stock price prediction using principal components, *PLoS One*, 15(3): e0230124.
<https://doi.org/10.1371/journal.pone.0230124>.
- [23] Gordon H. L., & Richard G. J. Kendall's Coefficient of Concordance for Sociometric Rankings with Self Excluded, *Sociometry*, 34(4), 496–503, 1971.
<https://doi.org/10.2307/2786195>
- [24] Guresen, E., Kayakutlu, G., & Daim, T. U. Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, 38 (8), 10389–10397, 2011.
<https://doi.org/10.1016/j.eswa.2011.02.068>
- [25] Granger, C. W. J., & Morgenstern, O. *Predictability of stock market prices: 1*. DC Heath Lexington, Mass. 1970.
- [26] Hassan, M. R., Nath, B., & Kirley, M. A fusion model of hmm, ann and ga for stock market forecasting. *Expert Systems with Applications*, 33 (1), 171–180, 2007.
<https://doi.org/10.1016/j.eswa.2006.04.007>
- [27] Hsu, M.-W., Lessmann, S., Sung, M.-C., Ma, T., & Johnson, J. E. Bridging the divide in financial market forecasting: Machine learners vs. financial economists. *Expert Systems with Applications*, 61, 215–234, 2016.
<https://doi.org/10.1016/j.eswa.2016.05.033>
- [28] Huang, C.-J., Yang, D.-X., & Chuang, Y.-T. Application of wrapper approach and composite classifier to the stock trend prediction. *Expert Systems with Applications*, 34(4), 2870–2878, 2008.
<https://doi.org/10.1016/j.eswa.2007.05.035>
- [29] Khan, W., Ghazanfar, M. A., Azam, M. A., Karami, A., Alyoubi, K. H., & Alfakeeh, A. S. Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, 1-24, 2020.
<https://doi.org/10.1007/s12652-020-01839-w>
- [30] Khansa, L., & Liginlal, D. Predicting stock market returns from malicious attacks: A comparative analysis of vector autoregression and time-delayed neural networks. *Decision Support Systems*, 51 (4), 745–759, 2011.
<https://doi.org/10.1016/j.dss.2011.01.010>
- [31] Kim, J. H., Shamsuddin, A., & Lim, K. P. Stock return predictability and the adaptive markets hypothesis: Evidence from century-long us data. *Journal of Empirical Finance*, 18 (5), 868–879, 2011.
<https://doi.org/10.1016/j.jempfin.2011.08.002>
- [32] Meesad, P., & Rasel, R. I. (2013). Predicting stock market price using support vector regression. In *Informatics, electronics & vision (iciev)*, 2013 international conference on (pp. 1–6). IEEE.
<https://doi.org/10.1109/iciev.2013.6572570>
- [33] Nayak, A., Pai M., M., M., & Pai R., M. Prediction Models for Indian Stock Market. Twelfth International Multi-Conference on Information Processing-2016 (IMCIP-2016). *Procedia Computer Science* 89 441 – 449, 2016.
<https://doi.org/10.1016/j.procs.2016.06.096>
- [34] Nti, I. K., Adekoya, A. F., & Weyori, B. A. A comprehensive evaluation of ensemble learning for stock-market prediction. *Journal of Big Data*, 7(1), 1–40, 2020.
<https://doi.org/10.1186/s40537-020-00299-5>
- [35] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, 42 (1), 259–268, 2015a.
<https://doi.org/10.1016/j.eswa.2014.07.040>
- [36] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, 42 (4), 2162–2172, 2015b.
<https://doi.org/10.1016/j.eswa.2014.10.031>
- [37] Phan, D. H. B., Sharma, S. S., & Narayan, P. K. Stock return forecasting: Some new evidence. *International Review of Financial Analysis*, 40, 38–51, 2015.

- <https://doi.org/10.1016/j.irfa.2015.05.002>
- [38] Rajashree D., & Pradipta K., D. A hybrid stock trading framework integrating technical analysis with machine learning techniques. *The Journal of Finance and Data Science*, 2: 42–57, 2016.
<https://doi.org/10.1016/j.jfds.2016.03.002>
- [39] Rather, A. M., Agarwal, A., & Sastry, V. Recurrent neural network and a hybrid model for prediction of stock returns. *Expert Systems with Applications*, 42 (6), 3234–3241, 2015.
<https://doi.org/10.1016/j.eswa.2014.12.003>
- [40] Rokach, L., & Maimon, O. Z. *Data mining with decision trees: theory and applications* (Vol. 69). World scientific, 2008.
- [41] Tan, T. Z., Quek, C., & See, Ng. G. Biological brain-inspired genetic complementary learning for stock market and bank failure prediction. *Computational Intelligence*, 23(2), 236–261, 2007.
<https://doi.org/10.1111/j.1467-8640.2007.00303.x>
- [42] Thawornwong, S., & Enke, D. The adaptive selection of financial and economic variables for use with artificial neural networks. *Neurocomputing*, 56, 205–232, 2004.
<https://doi.org/10.1016/j.neucom.2003.05.001>
- [43] Tsai, C.-F., Lin, Y.-C., Yen, D. C., & Chen, Y.-M. Predicting stock returns by classifier ensembles. *Applied Soft Computing*, 11 (2), 2452–2459, 2011.
<https://doi.org/10.1016/j.asoc.2010.10.001>
- Tsai, C.-F., & Hsiao, Y.-C. Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches. *Decision Support Systems*, 50 (1), 258–269, 2010.
<https://doi.org/10.1016/j.dss.2010.08.028>
- [44] Vijha M., Chandolab D., Tikkiwalb V. A., & Kumarc A. Stock Closing Price Prediction using Machine Learning Techniques, *Procedia Computer Science* Vol. 167, 599–606, 2020.
<https://doi.org/10.1016/j.procs.2020.03.326>
- [45] Wang, J.-J., Wang, J.-Z., Zhang, Z.-G., & Guo, S.-P. Stock index forecasting based on a hybrid model. *Omega*, 40 (6), 758–766, 2012.
<https://doi.org/10.1016/j.omega.2011.07.008>
- [47] Wang, J.-Z., Wang, J.-J., Zhang, Z.-G., & Guo, S.-P. Forecasting stock indices with back propagation neural network. *Expert Systems with Applications*, 38(11), 14346–14355, 2011.
<https://doi.org/10.1016/j.eswa.2011.04.222>
- [48] Wang, L., Zeng, Y., & Chen, T. Back propagation neural network with adaptive differential evolution algorithm for time series forecasting. *Expert Systems with Applications*, 42 (2), 855–863, 2015.
<https://doi.org/10.1016/j.eswa.2014.08.018>
- [49] Weng, B., Lu L., Wang, X., Megahed, F., M., Martinez, W. Predicting short-term stock prices using ensemble methods and online data sources. *Expert Systems with Applications*, 112: 258–273, 2018.
<https://doi.org/10.1016/j.eswa.2018.06.016>
- [50] Zhang, L., Lu, D. & Wang, X. Measuring and testing interdependence among random vectors based on Spearman's ρ and Kendall's τ ., *Comput Stat*, 2020.
<https://doi.org/10.1007/s00180-020-00973-5>
- [51] Zhang, Y., & Wu, L. Stock market prediction of s&p 500 via combination of improved bco approach and bp neural network. *Expert Systems with Applications*, 36 (5), 8849–8854, 2009.
<https://doi.org/10.1016/j.eswa.2008.11.028>
- [52] Zhong, X., Enke, D. Predicting the daily return direction of the stock market using hybrid machine learning algorithms. *Financial Innovation*, 5, 24, 2019.
<https://doi.org/10.1186/s40854-019-0138-0>

8 Appendix

Volume Indicator	Description
Chaikin A/D Line (ADL)	Estimates the Advance/Decline of the market.
Chaikin A/D Oscillator (ADOSC)	Indicator of another indicator. It is created through application of MACD to the Chaikin A/D Line
On Balance Volume (OBV)	Uses volume flow to forecast changes in price of stock

Table 10: Description of Volume Indicators used in the study.

Overlap Studies Indicators	Description
Bollinger Bands (BBANDS)	Describes the different highs and lows of a financial instrument in a particular duration.
Weighted Moving Average (WMA)	Moving average that assign a greater weight to more recent data points than past data points
Exponential Moving Average (EMA)	Weighted moving average that puts greater weight and importance on current data points, however, the rate of decrease between a price and its preceding price is not consistent.
Double Exponential Moving Average (DEMA)	It is based on EMA and attempts to provide a smoothed average with less lag than EMA.
Kaufman Adaptive Moving Average (KAMA)	Moving average designed to be responsive to market trends and volatility.
MESA Adaptive Moving Average (MAMA)	Adjusts to movement in price based on the rate of change of phase as determined by the Hilbert transform discriminator.
Midpoint Price over period (MIDPRICE)	Average of the highest close minus lowest close within the look back period
Parabolic SAR (SAR)	Heights potential reversals in the direction of market price of securities.
Simple Moving Average (SMA)	Arithmetic moving average computed by averaging prices over a given time period.
Triple Exponential Moving Average (T3)	It is a triple smoothed combination of the DEMA and EMA
Triple Exponential Moving Average (TEMA)	An indicator used for smoothing price fluctuations and filtering out volatility. Provides a moving average having less lag than the classical exponential moving average.
Triangular Moving Average (TRIMA)	Moving average that is double smoothed (averaged twice)

Table 11: Description of Overlap Studies Indicators used in the study.

Momentum Indicators	Description
Average Directional Movement Index (ADX)	Measures how strong or weak (strength of) a trend is over time
Average Directional Movement Index Rating (ADXRR)	Estimates momentum change in ADX.
Absolute Price Oscillator (APO)	Computes the differences between two moving averages
Aroon	Used to find changes in trends in the price of an asset
Aroon Oscillator (AROONOSC)	Used to estimate the strength of a trend
Balance of Power (BOP)	Measures the strength of buyers and sellers in moving stock prices to the extremes
Commodity Channel Index (CCI)	Determine the price level now relative to an average price level over a period of time
Chande Momentum Oscillator (CMO)	Estimated by computing the difference between the sum of recent gains and the sum of recent losses
Directional Movement Index (DMI)	Indicate the direction of movement of the price of an asset
Moving Average Convergence /Divergence (MACD)	Uses moving averages to estimate the momentum of a security asset
Money Flow Index (MFI)	Utilize price and volume to identify buying and selling pressures
Minus Directional Indicator (MINUS_DI)	Component of ADX and it is used to identify presence of downtrend.
Momentum (MOM)	Measurement of price changes of a financial instrument over a period of time
Plus Directional Indicator (PLUS_DI)	Component of ADX and it is used to identify presence of uptrend.
Log Return	The log return for a period of time is the addition of the log returns of partitions of that period of time. It makes the assumption that returns are compounded continuously rather than across sub-periods
Percentage Price Oscillator (PPO)	Computes the difference between two moving averages as a percentage of the bigger moving average
Rate of change (ROC)	Measure of percentage change between the current price with respect to a at closing price n periods ago.
Relative Strength Index (RSI)	Determines the strength of current price in relation to preceding price
Stochastic (STOCH)	Measures momentum by comparing closing of a security with earlier trading range over a specific period of time
Stochastic Relative Strength Index (STOCHRSI)	Used to estimate whether a security is overbought or oversold. It measures RSI over its own high/low range over a specified period.
Ultimate Oscillator (ULTOSC)	Estimates the price momentum of a security asset across different time frames.
Williams' %R (WILLR)	Indicates the position of the last closing price relative to the highest and lowest price over a time period.

Table 12: Description of Momentum Indicators used in the study.

Price Transform Indicator	Description
Median Price (MEDPRICE)	Measures the mid-point of each day’s high and low
Typical Price (TYPPRICE)	Measures the average of each day’s price.
Weighted Close Price (WCLPRICE)	Average of each day's price with extra weight given to the closing price.

Table 13: Description of Price Transform Indicators used in the study.

