Scientia Iranica E (2021) 28(1), 395–411

SCIENTIA IRANICA

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
http://scientiairanica.sharif.edu

Development of an ensemble learning-based intelligent model for stock market forecasting

M.T. Faghihi Nezhad[a] and B. Minaei Bidgoli[b,*]

a. *Department of Information Technology, Faculty of Engineering, Payame Noor University, Tehran, Iran.*
b. *Faculty of Computer Engineering, Iran University of Science and Technology, Tehran, Iran.*

Received 27 January 2018; received in revised form 28 March 2019; accepted 2 September 2019

KEYWORDS
Forecasting the direction of price movement;
Ensemble learning;
Bagging;
Forecasting stock price;
Evolutionary computing;
Intelligent trading system.

Abstract. The use of artificial intelligence-based models has shown that the market is predictable despite its uncertainty and unstable nature. The most important challenge of the proposed models in the stock market is to ensure high accuracy of results and high forecasting efficiency. Another challenge, which is a prerequisite for making decisions and using the results of the forecast for profitability of transactions, is to forecast the trend of stock price movements in forecasting price targets. To overcome the mentioned challenges, this paper employs Ensemble Learning (EL) model using intelligence-based learners and metaheuristic optimization methods to maximize the improvement of forecasting performance. In addition, to take into account the direction of price changes in stock price forecasting, a two-stage structure is used. In the first stage, the next movement of the stock price (increase or decrease) is forecasted and its outcome is then employed to forecast the price in the second stage. In both stages, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are used to optimize the aggregation results of the base learners. The evaluation results of stock market dataset show that the proposed model has higher accuracy than other models used in the literature.

1. Introduction

Accurate forecasting of stock market behavior is invaluable for traders in this market. So, forecasting financial time series is an important and challenging problem [1], in which researchers try to extract hidden patterns to forecast the future behavior of the market [2].

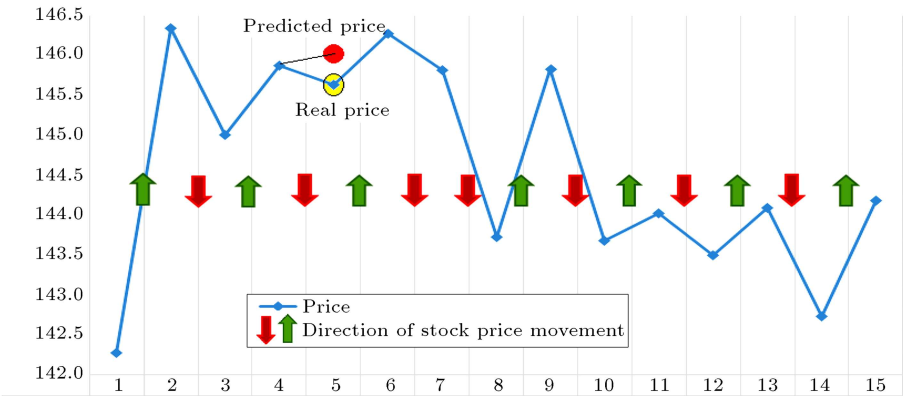
Improvement of forecasting performance and accurate results is the main challenge of the stock market [3,4]. The second challenge in forecasting models is the lack of attention to the stock price trend (the direction of stock price movement) [5]. The investor should accurately forecast price changes (increase or decrease) for trading. Most of the proposed models in this area have been developed for price forecasting. In the above studies, forecasting performance was evaluated based on such criteria as Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) [6–9]. These criteria facilitate the evaluation of the proximity of the forecasted price to the real price; however, they are not able to evaluate the model ability to forecast the stock price trends.

To illustrate the price concept and the direction of price movement, Apple stock price and the direction of its stock price movement are depicted in Figure 1 for 15 consecutive days from 6/16/2017 to 7/7/2017.

To determine the importance of forecasting the direction of price movement shown in Figure 1, consider

*. *Corresponding author.*
E-mail addresses: mfaghihi@pnu.ac.ir (M.T. Faghihi Nezhad); b_minaei@iust.ac.ir (B. Minaei Bidgoli)

doi: 10.24200/sci.2019.50353.1654



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Price	142.27	146.34	145.01	145.87	145.63	146.28	145.82	143.73	145.83	143.68	144.02	143.50	144.09	142.73	144.18
Direction	1	-1	1	-1	1	-1	-1	1	-1	1	-1	1	-1	1	0

Figure 1. The price and direction of the price movement.

the price on the fourth day at 145.87$. A trader predicts the price target of 145.9$, implying that he predicts an increment in price for the next (fifth) day. He opens a buy position while the real price on the next day would be $145.63 (lower than predicted) and he would face a loss in the trade. The main reason for such a loss in this trade is that forecasting the direction of price movement was wrong in the first place, while the price forecasting was only subject to 0.18% error.

To avoid similar losses occurring again, it is necessary to forecast the price by forecasting the direction of the stock price movement. To overcome the two above-mentioned challenges, the proposed model uses an Ensemble Learning (EL) algorithm equipped with intelligent-based models and meta-heuristic technique to maximize the quality of the prediction results.

Moreover, a two-stage structure is used to take into account the direction of price movement in price forecasting. In the first stage, the next direction of the price movement (increase or decrease) is forecasted and it is used for forecasting the price in the second stage. To the best of our knowledge, this is the first study that has considered both the direction of the stock price movement and the stock price itself simultaneously so as to forecast the stock price.

The rest of the paper is organized as follows. In Section 2, the predictability of the stock market and various models in stock market forecasting are reviewed. In Section 3, a two-stage procedure is proposed based on EL using intelligence-based models. The proposed model is evaluated and thoroughly compared to other models in Section 4 and experimental results are then presented in this section. Finally, conclusions are given in Section 5.

2. Literature review

There are two main hypotheses about forecasting stock market:

1. All available information is fully reflected by the market prices, according to "efficient market hypothesis" and instability in prices is then considered based on the results of new information. According to this hypothesis, it is impossible to earn higher returns through intelligent stock selection methods or other forecasting techniques, and the only way to do so is selecting stocks with high risks or by chance [10]. In an efficient market, if the expectations and knowledge of all market participants are well reflected in prices, price fluctuations cannot be forecasted;
2. Another hypothesis that is compatible with the efficient market hypothesis is "Random Walk (RW)" according to which the trend of volatilities in stock market prices is random and thus cannot be forecasted. However, some recent studies have rejected the RW behavior of stock prices [11]. Also, the application of artificial intelligence in financial fields has reinforced the idea that the market might not be always efficient and one can forecast future prices based on historical data using various techniques [12,13]. Since the nature of the financial time series is fundamentally complex, noisy, dynamic, nonlinear, non-parametric, and chaotic [13], the stock market forecasting is a challenging issue for researchers [1,14].

There are different approaches to forecasting stock market using historical data. One way is to generally categorize the approaches into "linear" and "nonlinear" techniques, while another approach considers them as "statistical" and "machine learning" [15]. Among these categories, intelligent and classical ones are suitable. According to the classical forecasting approach, it is assumed that the future price value follows the linear trend of the past values. Autoregressive Moving Average (ARIMA), Autoregressive Conditional Heteroscedasticity (GARCH), and regression belong to the classical class. Artificial Neural Networks (ANNs), fuzzy logic, Support Vector Machines (SVMs), EL, and meta-heuristic algorithms all belong to intelligent techniques [9,15]. These methods, unlike the classic ones, are capable of obtaining a nonlinear relationship between input variables without having information about the statistical distribution of these inputs.

2.1. Intelligent models

The intelligent models used for time series forecasting have some advantages and disadvantages [16]. The conducted comparisons show that intelligent models can overcome the limitations of linear models, in which they can extract a pattern better from data with higher forecasting accuracy [9,17]. In recent years, most of the studies conducted over forecasting stock market have focused on intelligent models [18]. These models for stock market forecasting can be divided into three groups: single models, hybrid models, and EL models, as shown in Figure 2.

The first group in Figure 2 uses a single model for forecasting and it is of two types:

1. The models that employ one technique;
2. The models that use multiple techniques to forecast.

According to the studies conducted by Atsalakis and Valavanis [43] Tkáč and Verner [18] among different intelligent models, ANN techniques have been applied more than other techniques due to their better performance [18]. The ANNs were used in these researches to forecast the stock price [23] or the direction of stock price movement [2]. Despite the complexity of stock market forecasting, it is shown that ANNs with only a hidden layer can model such a complex system with acceptable accuracy.

Although the application of ANNs has promoted the forecasting accuracy more than the classical models, there are some obstacles in this regard such as getting stuck in local optimum and over-fitting, which make the forecasting accuracy challengeable [15]. One of the suitable approaches to improving forecasting accuracy is using multi-technique models. For example, the ANNs combined with meta-heuristic algorithms are ordinarily employed to overcome the mentioned problems and improve Neural Network (NN) training [21,22,29]. Another approach is applying Neuro-fuzzy techniques equipped with meta-heuristic algorithms [24,28].

The second group in Figure 2 is called hybrid forecasting models that combine different single models for attaining results with higher accuracy. By integrating the ANN as a nonlinear technique with ARIMA as a linear method, one can enjoy their advantages [31]. Also, a hybrid model comprising RW for exploring linear patterns and two NN models for uncovering non-linear patterns were proposed in [34]. Tsai and Chiou [33] achieved higher forecasting accuracy with a combination of ANN and Decision Tree (DT) models. Wang et al. [14] proposed a hybrid model combining Exponential Simple Model (ESM), ARIMA, and Back-Propagation Neural Network (BPNN), in which the weight of each single model was determined by the Genetic Algorithm (GA). Andrawis et al. [37] combined computational intelligence and linear models with such methods as mean, trimmed mean, and winsorized mean. In these types of research, hybrid models were compared with the existing single models in the literature and the obtained results demonstrated that such combined models could outperform single models in forecasting accuracy [6,9,32,36].

The third group in Figure 2 comprises the models

<table>
<tr><th colspan="4">Intelligent models</th></tr>
<tr><td rowspan="2">Single models</td><td>Single-technique</td><td>NN: (Kara, 2011) [2], (Zhong, 2017) [13]
SVM: (Kara, 2011) [2]
DEEP NN: (Arevalo, 2016) [19], (Chong, 2017) [20]</td><td rowspan="4">GA: genetic algorithm
FS: Feature Selection
SA: Simulated Annealing
FC: Fuzzy Clustering
ASFA: artificial fish swarm algorithm
MLP: multilayer perceptron
MOPSO: Multi-Objective Particle Swarm Optimization
NSGA-II: Non-dominated Sorting Genetic Algorithm

DT: decision tree
ESM: exponential simple model
SVR: support vector regression
RNN: recurrent neural network
FANNs: feedforward ANNs
EANNs: Elman ANNs
SVR: support vector regression
RF: Random forest
RBF: Radial basis function Multi-technique</td></tr>
<tr><td>Multi-technique</td><td>NN & GA: (Hassan, 2007) [21], (Asadi, 2012) [22], (Göçken., 2016)[23]
Neuro Fuzzy & SA: (Chang and Liu, 2008) [24]
SVM &GA: (Yu et al., 2009) [25]
Neuro Fuzzy & SVR: Huang, Yang, & Lee, 2018 [26]
Neuro Fuzzy & FC: (Esfahanipour, 2010) [27], (Chen, 2016) [28]
RBF & K-means & AFSA: (Shen et al., 2011) [29]
FS, FC & Fuzzy NN: (Enke, 2013) [30], (Chen, 2016) [28]
Chaos theory & MLP & MOPSO & NSGA-II (Ravi, 2017) [9]</td></tr>
<tr><td colspan="2">Hybrid models</td><td>ARIMA, NN: (Khashei, 2009) [31], (Babu, 2014) [32]
NN, DT: (Tsai and Chiou, 2009) [33]
ARIMA, ESM, NN: (Wang, 2012) [14]
RW, FANN, EANN: (Adhikari and Agrawal, 2014) [34]
SVR, RF: (Patel, 2015) [6]
RBF, RW: (Freitas and Rodrigues, 2006) [35]
Linear and nonlinear Model: (Rather, 2015) [36]
Computational Intelligence, Linear Model: (Andrawis, 2010 [37]</td></tr>
<tr><td colspan="2">Ensemble learning model</td><td>(Dietterich, 2000) [38], (Yu, 2008) [39], (Tsai, 2011) [12], (Xiao, 2013) [40], (Ballings, 2015) [41], (Maknickiene, 2016) [42], (Lin, 2017) [16]</td></tr>
</table>

Figure 2. Classification of intelligent models.

based on the EL algorithm. These algorithms belong to the computational intelligence approach that integrates a set of base learners into a single model [44]. It is also shown that the necessary and suitable conditions for an ensemble learner to achieve higher accuracy than base learners depend on the accuracy and diversity of categorizing the members. Here, "accuracy" implies "better than random prediction" and "diversity" means that the learners make uncorrelated errors.

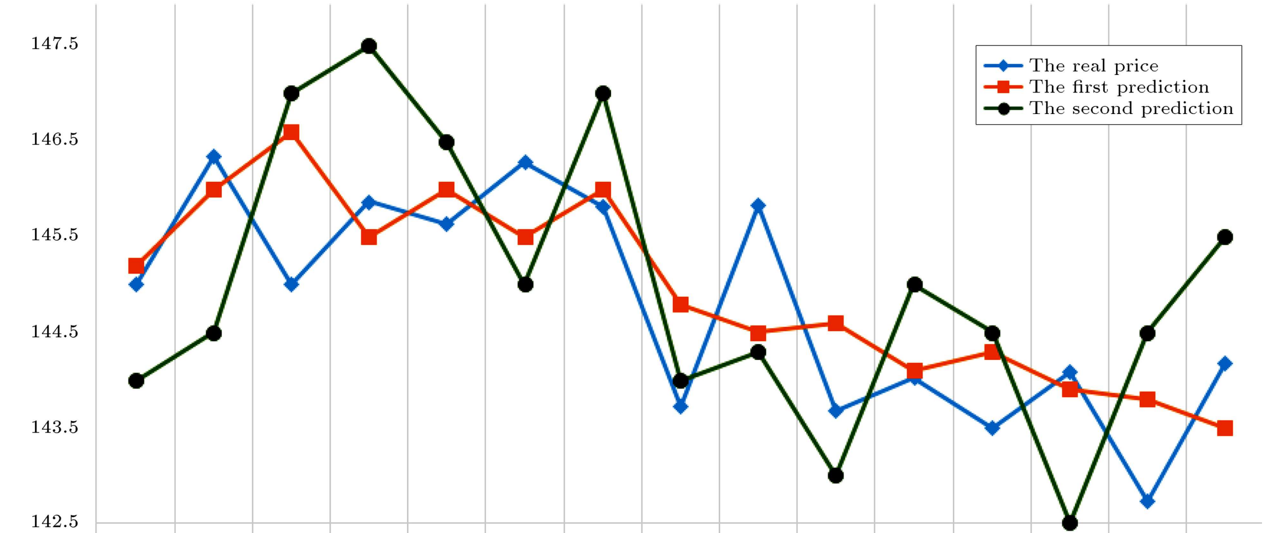
Tsai et al. [12] examined two types of ensembles, i.e., 'homogeneous' and 'heterogeneous' classifier ensembles, for prediction accuracy of stock returns. The results indicated that the homogeneous multiple classifiers using NNs outperformed single classifiers. Lin et al. [16] proposed the RF-based Extreme Learning Machine (ELM) ensemble model to achieve accuracy, stability, and efficiency simultaneously in time series forecasting. Ballings et al. [41] investigated the performance of ensemble methods against single models in the stock forecasting and suggested that novel studies in this domain should include ensemble algorithms.

By considering the natural complexity, instability, and noisiness of the stock market forecasting problem, it is required that several computing techniques be integrated synergistically rather than exclusively [45]. By exploring the literature, one can obviously find that EL algorithms have better performance than single models for a wide range of applications and different scenarios. Their results are more accurate, reliable, and stable [12,16,37,38].

2.2. Problem definition

By introducing various methods, researchers attempt to demonstrate the ability of the proposed models to increase the accuracy of stock market forecasting. According to the results of the conducted literature review in this research, the ensemble learning algorithms are more accurate, reliable, and stable in forecasting the stock market. Therefore, the proposed forecasting model should use EL algorithms to maximize the performance of the predictive result. In addition, to use the results of a stock market forecasting model in a real environment and generate profit, the direction of price movement in price forecasting should be considered (Figure 3).

In Figure 3, Apple's stock price is displayed along with the two-price forecast. The calculated MAPE for the first forecast is 0.46%. The price on the second day is 146.34 and the forecasted price for the third day is 146.6. Since the price is forecasted to increase, the trader takes the buying position, while the real price on the third day will be 145.01, which leads to a loss of \$1.3. According to this forecast, a loss of \$5.9 in 14 days will eventually occur based on daily trading. The calculated MAPE for the second forecast is 0.87, which is almost 90% more than that of the first forecast; however, a profit of \$2.3 in 14 days will take place through daily trade based on this forecast. The reason for such profitability is accurate forecasting of the direction of stock price movement and the price forecast itself. While in the first forecast, as it can be



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	145.00	146.34	145.01	145.87	145.63	146.28	145.82	143.73	145.83	143.68	144.02	143.50	144.09	142.73	144.18
	145.2	146.0	146.6	145.5	146.0	145.5	146.0	144.8	144.5	144.6	144.1	144.3	143.9	143.8	143.5
Profit-1		1.3	-1.3	-0.9	-0.2	-0.6	-0.5	2.1	-2.1	-2.2	-0.3	-0.5	-0.6	1.4	-1.4
	144.0	144.5	147.0	147.5	146.5	145.0	147.0	144.0	144.3	143.0	145.0	144.5	142.5	144.5	145.5
Profit-2		1.8	-2.0	1.6	0.9	-1.3	-1.2	0.3	1.5	0.7	1.0	1.0	-1.6	-1.8	1.3

Figure 3. The real price besides two forecasted price tags.

observed in Figure 3, the view is to follow a sequence of the previous movements.

The proposed model in this research comprises a two-stage structure to solve the mentioned problem. In the first stage, the direction of the stock price movement (increase or decrease) is forecasted and its result will be used so as to forecast the price in the second stage. The model also employs EL through intelligent-based models as well as meta-heuristic algorithms in both stages that can maximize the performance of the forecasted results.

3. Development of an intelligent ensemble-based model for stock price forecasting

Various studies that have applied training methods show that no specific, ideal training algorithm can be flawless in all predictions. To overcome such an obstacle, EL algorithms have been developed to a large extent so as to reduce the error, which is the main motivation for their development.

The basic assumption about this methodology is that in EL, the probability of prediction error in an unknown sample is much lower than that in an individual model. In comparison with common machine learning methods that try to learn a hypothesis from training data, in EL models, several learners are trained to attain the greatest possible accuracy and also, they try to construct a set of hypotheses and compositions [46]. The learners used in EL are called base learners. The EL algorithm, which is a combination of base learners, has better accuracy than the individual models [38]. The general rule in EL systems is that the results of the base learners are different from each other as much as possible. This diversity can be obtained in different ways. In this regard, one can mention the four following proposed methods:

1. Using different training datasets for training the base learner by resampling methods in which the sub-set of the original training data is selected randomly and will be replaced by the original training dataset;
2. To ensure that the boundaries are different while using different training data, unstable models are used as base models because they can make different decision boundaries, even with fewer changes in their training parameters [47];
3. Another way to ensure diversity in parameters is to use different models. For example, a set of multi-layer perceptron NNs can be trained with initial weights, a number of layers and nodes, different error criteria, and so on. Considering such parameters can control the instability of the individual models and ultimately, diversify them. The ability to control unstable ANNs has become the ideal feature of EL algorithms;
4. By using different features, the input space is divided into different subsets of original features that might overlap and each subset is given to a model as an input. By using this method, every base learner explores some part of knowledge and also, the diversity of features makes EL algorithms yield better results.

Bagging as one of the simplest EL algorithms is offered to improve the performance of prediction models, while the combinative strategy of base learners in them is the majority vote. Diversity in bagging is made through the bootstraps that are randomly selected and replaced by the original training data. Each bootstrap is used to train a learner of the same type. Failure to use unstable predictors could lead to almost identical predictors that no longer make improvements in efficiency of the individual predictor. For the same reason, in bagging, unstable learning models like DT and ANNs are very efficient and effectively used because small changes in data can cause large changes to the prediction result [47]. After training different base learners, to achieve final prediction, the obtained results of all learners are combined to predict an instance with different methods. In the simple weighted mean method, the weights of all learners are the same for producing the final result of an instance. The weight of each learner in the weighted mean method for final forecast is determined based on the accuracy of the training step and then, it is compared to other learners. The effect of each learner on the result of the final forecast can be considered as an optimization problem. The objective of this optimization problem is to determine the best weights for each learner such that the prediction accuracy of the test data is maximized. In this research, two well-known meta-heuristic algorithms, i.e., Particle Swarm Optimization (PSO) and GA, are employed to tackle this optimization problem.

3.1. The proposed model

The existing challenge of models, as presented in the previous sections, was their failure to pay attention to stock price and direction of the price movement, simultaneously. To tackle such difficulty, in this subsection, a new stock price forecasting model is introduced by considering the price and direction of price movement, concurrently. The proposed model includes two interdependent stages. First, the direction of price change is forecasted and added to other features as a new characteristic and this new dataset is then used for the forecast later on. To maximize the classification accuracy (forecasting the direction of price movement) in the first stage, the bagging algorithm, as a kind of EL algorithms, is used, while this algorithm is employed in the second stage to maximize the regression accuracy (the price forecasting). The results of the base models should be as much diverse as possible so as to achieve appropriate accuracy. The diversity is attained using different training datasets for each model. Diverse datasets are obtained by resampling the subset of the training data randomly through replacement. In addition, the NN that can create different decision boundaries, even with low deviations in training parameters, is used as the base models. The aggregation of the results is carried out in four ways: optimization with GA, optimization with PSO, weighted aggregation based on the weight of each model obtained by the accuracy of the training data, and aggregation result with equal weights for each model. The best way to aggregate the base model is opted based on the accuracy.

3.1.1. The first stage (forecasting the direction of price movement)

In the first stage, the direction of the next stock price movements is forecasted. Most of the time series data in the stock market are non-stationary and trendy, which reduces the accuracy of forecasting stock market. The data must be as much de-trended and stationary as possible so that the hidden pattern in the series can be extracted more accurately [48]. Differentiation and logarithmic conversion can discover much knowledge from the data. The first difference in the time series creates a new time series whose values are different from the two successive values of the initial time series including the series of changes from one period to the next:

$$\nabla x_t = x_t - x_{t-1}, \tag{1}$$

where x_t denotes the value of the time series x_t in period t and the first difference of x_t in period t is $x_t - x_{t-1}$. By differencing the initial series, a new time series is obtained. The elements of the initial time series are stock prices, while the elements of the new time series are price changes.

The value of x in period t is auto-correlated with respect to its value in earlier periods, where the nth element of the series with k lag is added to the model as input and the $(n+1)$th element is then predicted. This value is considered as the price change in the next period. In the proposed model, the price data close to the values of the previous days are assigned as the initial inputs and with their differentiation, the new series is created. The output of the model is the difference between the "closed" price of today and that of the previous day.

Data preparation and formation of new time series are performed, in which the value of the new time series is obtained by a one-time difference between two successive elements of the initial series and the number of k lagged of that. Then, the new dataset is divided into "training" and "test" data groups. If the records contained in the dataset N are assumed, the N bootstraps are created through N times of sampling with replacement in training data. One NN is created and is trained N times through N bootstraps until N base models are obtained. In the following, the training data are added to each of the trained base models and the output is compared with the target output to determine the forecasting accuracy of the base model. If the forecasting accuracy is better than random forecasting (greater than 0.5), then the output of this model is maintained and the results (the forecasted direction of price movement) are added to the matrix result. After applying the training data to all trained models and completing the matrix of results, this matrix is aggregated using four methods and the best weight vector for the combination of the trained models is then obtained.

The results are aggregated using four methods: Simple Average Aggregation (SAV), Weighted Average Aggregation (WAV), GA, and PSO. Then, the methods that could determine the weights with highest accuracy are finally selected.

By considering the importance weight of the learner for the final performance, as already explained, obtaining the weights is defined as an optimization problem. In the following, the weight of each model is attained using PSO explained. Every particle in this algorithm is defined as a weight vector for combining the learners to compute the final output. Therefore, every particle of a vector equals the dimensions of a number of learners obtained in the previous steps. The weights and initial velocities are determined randomly for each particle. In the following, the performance (accuracy) of each particle (weights of the base learners) is calculated. The performance of each particle implies the performance of the learner in teamwork to reach the least possible error for all the training data obtained by particle-related weight combination. For example, for a particle with 0.5, 0.3, and 0.2 weights, the learner for a specific sample yields 59, 65, and 62 as outputs; therefore, the final output will be $61.4 = 0.5^*59 + 0.3^*65 + 0.2^*62$. A complete update of the group of particles is made based on the best personal and group experiences with a certain number of iterations. Finally, the best particle in the last iteration is used as the final weight for the combination of the base models.

In addition to PSO algorithm, GA is employed for obtaining the optimal weights of the base models. In this algorithm, each chromosome is considered as one weight vector. The number of genes concerned with each chromosome is equal to that of base models obtained in the previous steps. The initial weights are randomly generated in each chromosome. Then,

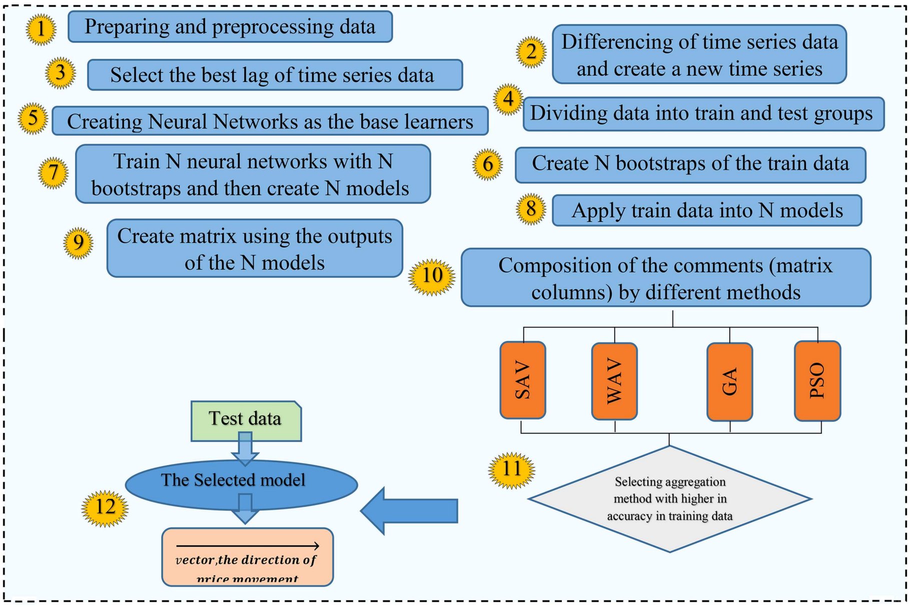


Figure 4. The implementation process of the first stage of the proposed model.

chromosomes are arranged based on their performance (exactly similar to PSO). To generate the next generation, the selection is conducted through the roulette wheel mechanism. Besides, the canonical two-point crossover and two-point mutation are applied over the selected parents. In the last iteration, after sorting the chromosomes based on their accuracy in determining the direction of price movement (their fitness) for all training data, the best one is selected as the final weight for the base learner combination.

WAV is another method for training the output matrix aggregation. Firstly, the accuracy level of each matrix column (forecast a base model) is calculated to forecast the target vector of training. The accuracy of each base model is divided by total accuracy, while the coefficient of each matrix column is obtained in the optimal combination vector. SAV is the simple average of the base models' output for aggregation of the results with equal weights.

The implementation process of the first stage of the proposed model is shown in Figure 4.

3.1.2. The second stage (price forecasting)

Following the termination of the model's first stage, its output, i.e., the direction of price upward/downward movement in the stock market, is obtained. In the second stage, by adding this feature to the existing ones (new dataset), a model is trained with a new dataset and the best combination of the lags is chosen through trial and error and it is then used as the input of the second stage.

The applied techniques in this stage are to some extent conceptually similar to the first stage and, yet, different in usage. The evaluation criteria for base models and aggregation methods in this stage are different, where instead of evaluating the accuracy of the results in correctly forecasting the direction of price movement, the accuracy evaluation in forecasting price is carried out through MAPE criterion. In this stage, the base models are trained using bootstraps and the next-time price is then forecasted. Also, to improve the accuracy and further assurance, the results of the base models are aggregated using different methods and the method with the highest accuracy is finally selected.

In the second stage, initially, a new dataset is created by adding the feature taken from the previous stage; then, the (near) optimal lag is selected by trial and error. Then, the new dataset is divided into "training" and "test" data and after that, N bootstraps are created from the training dataset.

One NN is created and later, it is trained with N bootstraps until N trained base models are learned. Training data are applied to all trained models and their output is added to the training output matrix. According to the EL algorithm mentioned in the previous section, the results are aggregated through four methods, where the best vector is ultimately chosen.

The aggregation methods of results in this stage are the same as those in the previous stage. The only difference is the criterion used for evaluating and selecting the optimal weight of the vectors in the matrix, which is MAPE. The same issue holds for the algorithms used in this stage, where based on the results of aggregation with PSO, each particle will be evaluated by MAPE criterion.

The process of implementing the second stage of the proposed model is depicted in Figure 5.

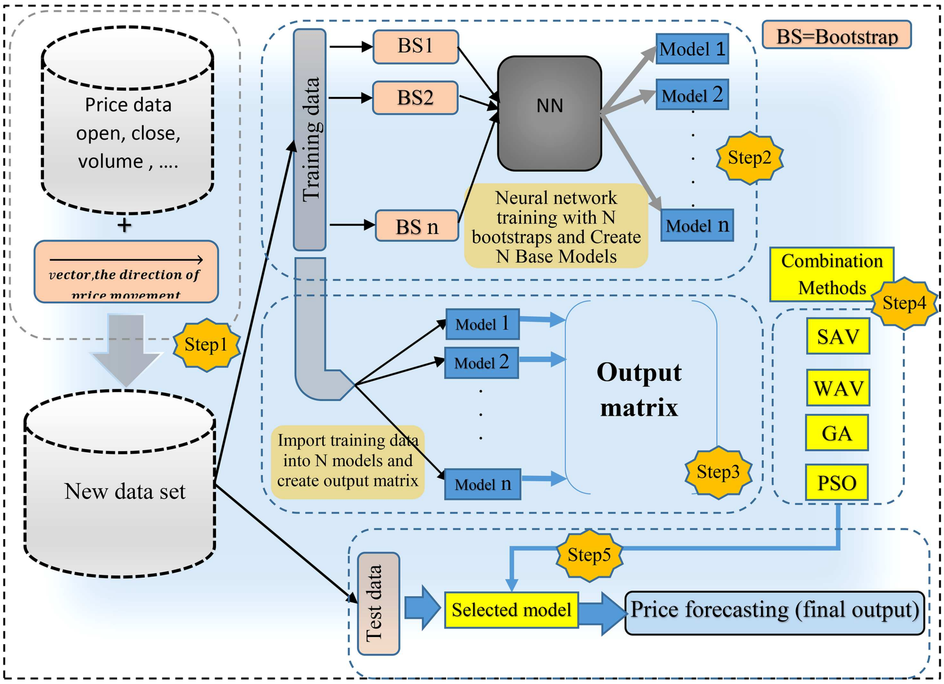


Figure 5. The implementation process of the second stage of the proposed model.

Table 1. Descriptions of stock indices.

Stock index name	From	To	Average	Standard deviation
Dow Jones Industrial Average Index (DJIA)	March 7, 2001	August 26, 2003	9345.324	925.15
Taiwan Stock Exchange index (TSE)	July 18, 2003	December 31, 2005	6070.557	1910.45
Tehran Prices Index (TEPIX)	April 10, 2006	January 30, 2009	9991.631	973.63
Tehran Index of Top 50 Companies (TIT50C)	April 10, 2006	January 30, 2009	16562.49	3715.06
Tehran Industry Index (TII)	April 10, 2006	January 30, 2009	7869.937	834.69
Tehran Index of Financial Group (TIFG)	April 3, 2006	January 30, 2009	20584.07	2383.53

4. Experimental results

In this section, the performance of the proposed model is evaluated through several datasets including the introduction of datasets, evaluation criteria, implementation of the proposed model, and comparison results of the proposed model and other researches.

4.1. Datasets

In order to compare the results of the proposed model and the accredited papers, the same datasets in the literature are used [22,24,27]. These data include different indices of the world's validated stock exchanges, showing general price changes in the market. In this paper, Dow Jones Industrial Average (DJIA), Taiwan Stock Exchange (TSE), and Tehran Price Index (TEPIX) together with three other Tehran's indices are investigated. Tehran Industry Index (TII) shows the average changes in the stock price of operating companies in the industrial sector; Tehran Index of Financial Group (TIFG) expresses the average changes in the stock price of operating companies in the financial sector; and the Tehran Index of Top 50 Companies (TIT50C) demonstrates the liquidity. The information of different indices is shown in Table 1.

4.2. Evaluation criteria

Since the objective of this paper is to improve the prediction of the direction of price movement and the price itself simultaneously, the criteria used for evaluating the outcomes should measure these two standards. The first criterion used to compare the models is MAPE [6–8]. Accordingly, at first, the absolute difference between the real and predicted values is divided by the real amount. Then, the outcome is divided by the number of total data. Eq. (2) shows how MAPE works, where y_i and p_i are the real and predicted amounts, respectively, and N is the number of data.

$$\text{MAPE} = 100 * \frac{1}{N} \sum\nolimits_{i=1}^{N} \frac{\mathrm{y_i} - p_i}{y_i}. \tag{2}$$

The Prediction On Change In Direction (POCID) presented in Eqs. (3) and (4) shows the calculation of the direction change prediction. The accuracy of the model in the direction prediction of the price movement besides the proper prediction of the price plays a leading role in gaining the profit. The POCID criterion ranges at the interval [0, 100]. The closer the value of POCID to 100, the higher the prediction accuracy [49].

$$\text{POICD} = 100 * \frac{1}{N} \sum_{i=0}^{N} D_i, \tag{3}$$

$$D_i = \begin{cases} 1, \text{ if } (y_i - y_{i-1})(p_i - p_{i-1}) > 0 \\ 0, \text{ otherwise} \end{cases} \tag{4}$$

The third criterion is Theil's U statistic (Eq. 5), which compares the model performance with RW model. If Theil's U is equal to 1, the model performance is equivalent to RW; however, if it is larger than 1, the performance is worse than RW. Obviously, if it is less than 1, the performance of the proposed model is better than RW [9].

$$D_i = \text{U of Tail} = \frac{\sum_{i=1}^{N} (y_i - p_i)^2}{\sum_{i=1}^{N} (y_i - y_{i+1})^2}. \tag{5}$$

The fourth used criterion in this study is Average Relative Variance (ARV) shown in Eq. (6). If the average of the time series is used instead of the forecasted values, the accuracy does not change. The value of this criterion, which is lower than 1 and close to 0, indicates better forecasting accuracy [50].

$$ARV = \frac{\sum_{i=1}^{N} (y_i - p_i)^2}{\sum_{i=1}^{N} (\bar{y} - p_i)^2}. \tag{6}$$

4.3. Evaluation of the proposed model

Researches show that the input variables of stock price are used in predictive models including price, technical, fundamental, and macroeconomic variables that can be categorized into different groups [51]. One common categorization for input variables of stock forecasting models divides them into two types: the first type is the price variables such as open, close, low, and high prices as well as the volume and the number of trading in a period. The second type is the technical variables that are derived from the price data using different formulas. Some researchers have used the price variables [9], while some others have employed technical variables [2,6].

In the proposed model in this paper, the price variable is used. Each time series in Table 1 contains 620 records, divided into two "training" and "test" data, of which 80% (500 records) are assigned to training and the rest of 20% (120 records) to testing. Several papers have divided the data into the same training and testing and also compared their results with the results of this model [22].

The final output of the proposed model is the next price forecasting owing to the price change in a trend. In the first stage that is responsible for specifying the direction of the next price changes, the data added to the first stage of the model is differentiated, in which the new data include price changes in two consecutive times.

In the proposed model, the base learners are composed of a three-layer feedforward NN, while the number of inputs is equal to that of used lags. Several factors in the model setting can affect the accuracy of the results, each of which has different levels. The settings for training data include use/non-use of logarithmic transformation, determining the percentage of validation dataset, the number of bootstraps, and amount of data used in each bootstrap as well as the number of inputs, the number of neurons, the training method, and the aggregation method of the results of the base learners. In this regard, combining all possible scenarios and testing them require a long time. To solve this problem and reduce the number of tests, the Taguchi optimization method is employed [52]. To do so, 32 different modes are selected using the Minitab software and the model is then implemented to achieve the highest possible accuracy among these combinations. The models with the highest accuracy in the training data are shown in Table 2 for each dataset and model settings.

The prediction output vector obtained from the first stage is added to other price variables to create a new input dataset for the second stage. In this stage, the learning model is repeated by changing its settings (similar to the first stage) to achieve the best results.

After training each individual model and their aggregation, the combination that has the best evaluation result of the training data is selected. The test data will then be added to the model and the model will be evaluated next. The evaluation results of the test data along with the settings that produced these results are shown in Table 3.

As shown in Table 3, using Dow Jones dataset yields an (near) optimal MAPE value, i.e., 1.126, when 3 lags are used. Also, 9 neurons in base models and %0.05 of data from each bootstrap are used for validation. The number of bootstraps is 100 and WAV is used as the aggregation method of base models. Figure 6 shows a comparison between the real and forecasted values using the proposed model.

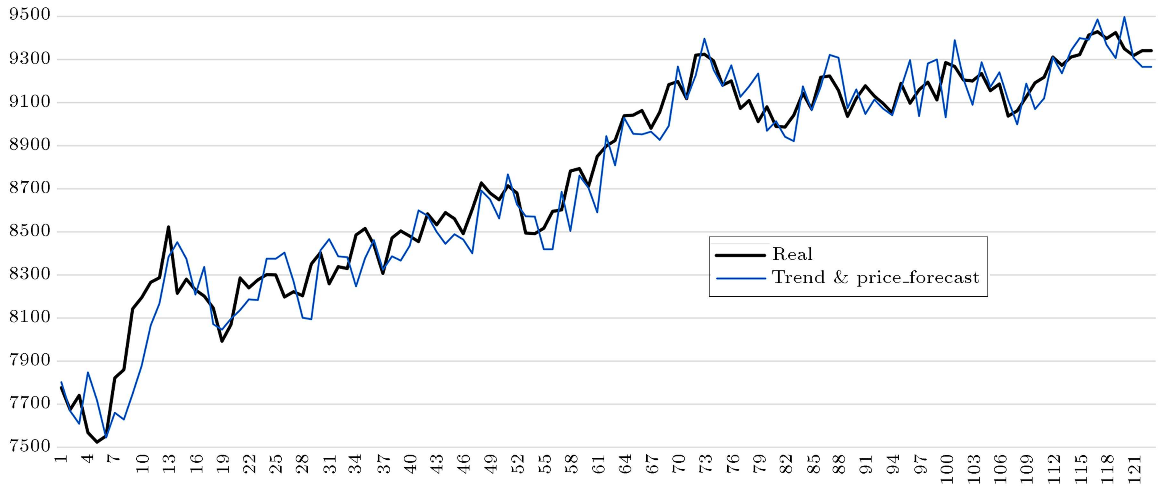


Figure 6. Comparison of real and forecasted values in the proposed model.

Table 2. Results of the first implementation stage for the training data in the proposed model.

Data name	DIJA	TSE	TEPIX	TIT50C	TII	TIFG
Using logarithmic transformation	√	–	√	–	–	√
The number of lag used	7	11	7	9	7	7
Percentage of validation data	0.03%	0.05%	0.05%	0.03%	0.05%	0.03%
The use amount of training data in each bootstrap	90%	100%	100%	95%	95%	95%
The number of bootstraps	300	400	200	100	200	300
The training method	Traingdm	Trainlm	Trainlm	Traingdm	Trainlm	Trainlm
The number of neurons	5	6	5	4	5	6
The aggregation method	PSO	GA	PSO	PSO	PSO	PSO
Percentage of correct forecasts of the price direction movement	73.1%	75.3%	76.4%	70.6%	80.7%	79.50%

Table 3. Results of the second implementation stage for the test data in the proposed model.

Data name	DIJA	TSE	TEPIX	TIT50C	TII	TIFG
The number of used lag	3	3	1	2	3	1
Percentage of validation data	0.05%	0.03%	0.03%	0.03%	0.05%	0.05%
Number of neurons in the neural network	9	9	7	9	7	5
The use amount of data from each bootstrap	100%	95%	100%	100%	95%	100%
The number of bootstraps	100	300	100	300	300	100
The aggregation method	WAV	PSO	GA	WAV	PSO	PSO
MAPE	1.126	0.659	0.37	0.368	0.286	0.31
POCID	61.15	81	60.202	58.491	74.3	69.182
U of Theil	0.518	0.494	0.552	0.483	0.588	0.672
ARV	0.068	0.068	0.003	0.086	0.04	0.013

In this paper, for aggregation of the results in both stages, two well-known meta-heuristic optimization algorithms are used, GA and PSO. Among different settings obtained through trial and error, the best parameters are selected for each dataset, as shown in Table 4.

4.4. Comparison between the proposed model and other models

The results of the proposed model are compared with the validated models reported in the literature [22,24,27], which have used the same datasets in the field of stock price forecasting with MAPE criterion.

Table 4. The values of used parameters in employed meta-heuristic algorithms for each dataset.

Data	**DJIA**		**TSE**		**TEPIX**		**Top 50 companies**		**Industry index**		**Financial group**	
Stage	**Stage I**	**Stage II**	**Stage I**	**Stage II**	**Stage I**	**Stage II**	**Stage I**	**Stage II**	**Stage I**	**Stage II**	**Stage I**	**Stage II**
GA parameters												
Population Size	150	150	200	200	200	200	200	150	150	200	200	200
Crossover Rate	0.75	0.75	0.80	0.80	0.75	85.00	0.75	0.75	0.75	85.00	0.75	0.75
Mutation Rate	0.15	0.15	0.08	0.08	0.10	0.13	0.10	0.10	0.10	0.13	0.15	0.15
Number of iterations	1500	1500	1000	1000	1500	2000	1500	1000	1000	2000	1500	1500
PSO parameters												
$C1 = C2$	2	2	2	2	2	2	2	2	2	2	2	2
Number of particles	200	200	200	200	150	150	150	200	200	150	100	100
Number of iterations	1200	1000	700	700	800	700	800	700	600	500	600	500

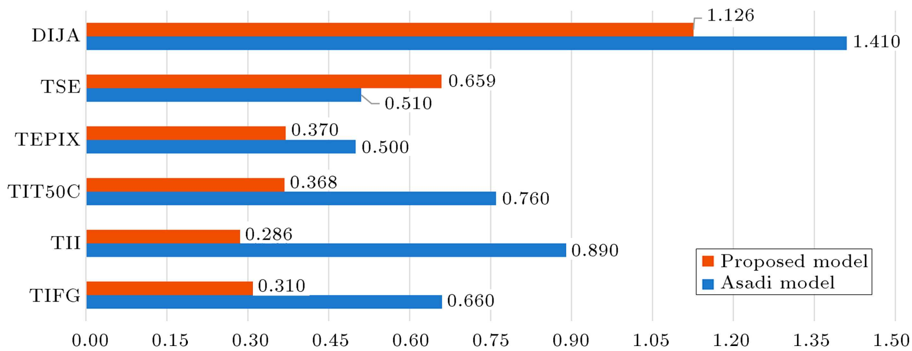


Figure 7. Comparison of two models considering Mean Absolute Percentage Error (MAPE) criterion.

Table 5 shows the MAPE value of the proposed model and the results of other models as well as their improvement percentage by the proposed model. According to Table 5, one can imply the superiority of the proposed model to other models in most cases.

Considering the price forecasting criteria and the direction of price movement in stock market is one of the main advantages of the proposed model in this paper. The obtained results are compared to those of Asadi's model [22], in which POICD criterion was used to predict the changes in the same datasets. The comparison shown in Table 6 indicates that the proposed model outperforms Asadi's model in terms of accuracy.

The results of Theil's U evaluation show that the proposed model outperforms RW model in terms of ARV criterion in all datasets. By taking MAPE and POICD into account, the proposed model, in most cases, has better forecasting accuracy than other models. The improved amount is shown in the imp% column of Table 6. For example, for Dow Jones dataset, MAPE values obtained from Asadi's model and the proposed model in this research are 1.41 and 1.126, respectively, showing an 18% improvement in price forecasting. However, the proposed model has made a 5% improvement in terms of the direction of price movement than this model.

Figures 7 and 8 show comparisons of six indices between the proposed model and Asadi's model in terms of MAPE and POCID criteria.

4.5. Discussion

As mentioned earlier, forecasting models have been divided into two groups: (a) price forecasting and (b) forecasting of the direction of price movement. In addition, it was clarified that the use of models forecasting the price regardless of price trends might cause a loss in real trading despite having less error in some important criteria. This is also true for models concerned only with forecasting the direction of price movement. Considerable effort has been made in this research to propose a model for forecasting the price with emphasis on price trend, which is better than

Table 5. Comparison of the results of the proposed model and other models

Dataset	Model	MAPE	Improvement
Dow Jones Industrial Average index (DJIA)	ARIMA	10.23	88.99%
	ANN (LM)	3.9	71.13%
	TAEF [50]	1.13	0.00%
	ANN trained with Back-Propagation Neural Network (BPNN) [22]	3.5	67.83%
	Pre-processing Evolutionary Neural Networks (PENN) [22]	2.4	53.08%
	Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22]	2	43.70%
	Pre-processed Evolutionary LM Neural Networks (PELMNN) [22]	1.4	19.57%
	Proposed model	1.126	–
Taiwan Stock Exchange index (TSE)	Hybrid of fuzzy clustering and TSK fuzzy system [27]	1.3	49.31%
	TSK-type fuzzy rule-based system [24]	2.4	72.54%
	ANN trained with Back-Propagation Neural Network (BPNN) [22]	0.78	15.51%
	Pre-processing Evolutionary Neural Networks (PENN) [22]	0.67	1.64%
	Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22]	0.52	–
	Pre-processed Evolutionary LM Neural Networks (PELMNN) [22]	0.51	–
	Proposed model	0.659	–
Tehran Prices Index (TEPIX)	Hybrid of fuzzy clustering and TSK fuzzy system [27]	2.4	84.58%
	ANN trained with back-propagation (BPNN) [22]	0.97	61.86%
	Pre-processing Evolutionary Neural Networks (PENN) [22]	0.64	42.19%
	Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22]	0.61	39.34%
	Pre-processed Evolutionary LM Neural Networks (PELMNN) [22]	0.5	26.00%
	Proposed model	0.37	–
Tehran Index of Top 50 Companies (TIT50C)	Hybrid of fuzzy clustering and TSK fuzzy system [27]	1.85	80.11%
	ANN trained with back-propagation (BPNN) [22]	1.4	73.71%
	Pre-processing Evolutionary Neural Networks (PENN) [22]	1.12	67.14%
	Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22]	0.83	55.66%
	Pre-processed Evolutionary LM Neural Networks (PELMNN) [22]	0.76	51.58%
	Proposed model	0.368	–
Tehran Industry Index (TII)	Hybrid of fuzzy clustering and TSK fuzzy system [27]	2.02	85.84%
	ANN trained with back-propagation (BPNN) [22]	1.73	83.47%
	Pre-processing Evolutionary Neural Networks (PENN) [22]	1.3	78.00%
	Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22]	0.98	70.82%
	Pre-processed Evolutionary LM Neural Networks (PELMNN) [22]	0.89	67.87%
	Proposed model	0.286	–
Tehran Index of Financial Group (TIFG)	Hybrid of fuzzy clustering and TSK fuzzy system [27]	1.03	69.90%
	ANN trained with back-propagation (BPNN) [22]	0.94	67.02%
	Pre-processing Evolutionary Neural Networks (PENN) [22]	0.79	60.76%
	Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22]	0.69	55.07%
	Pre-processed Evolutionary LM Neural Networks (PELMNN) [22]	0.66	53.03%
	Proposed model	0.31	–

Table 6. Comparison of the results of the proposed model and Asadi's model.

Criteria	MAPE		–	POICD		–	U of Theil	ARV
Stock name	**Asadi's model**	**Prop.* model**	**Imp.%**	**Asadi's model**	**Prop. model**	**Imp.%**	**Prop. model**	
Tehran Index of Financial Group (TIFG)	0.66	0.31	0.53	66.6	69.182	0.039	0.67	0.013
Tehran Industry Index (TII)	0.89	0.286	0.68	71.5	74.3	0.039	0.59	0.04
Tehran Index of top 50 Companies (TIT50C)	0.76	0.368	0.52	57.5	58.491	0.017	0.48	0.086
Tehran Stock Exchange Prices Index (TEPIX)	0.5	0.37	0.26	60	60.202	0.003	0.55	0.003
Taiwan Stock Exchange index (TSE)	0.51	0.659	–	85	81	–	0.49	0.068
Dow Jones Industrial Average Index (DJIA)	1.41	1.126	0.18	58.3	61.15	0.049	0.52	0.068

*Prop.: Proposed

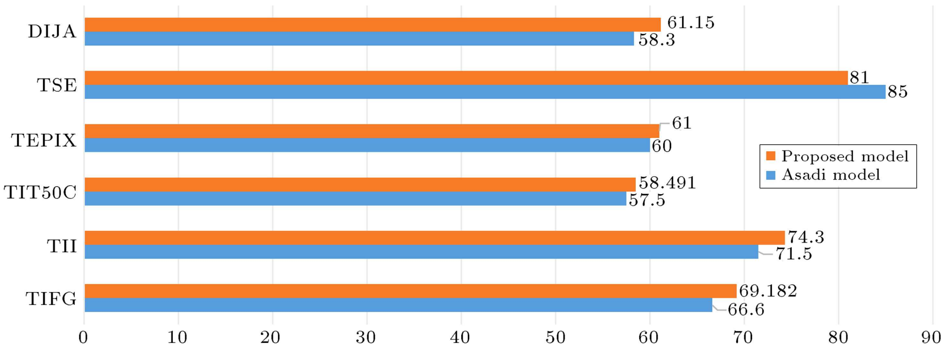


Figure 8. Comparison of two models considering Prediction On Change In Direction (POICD) criterion.

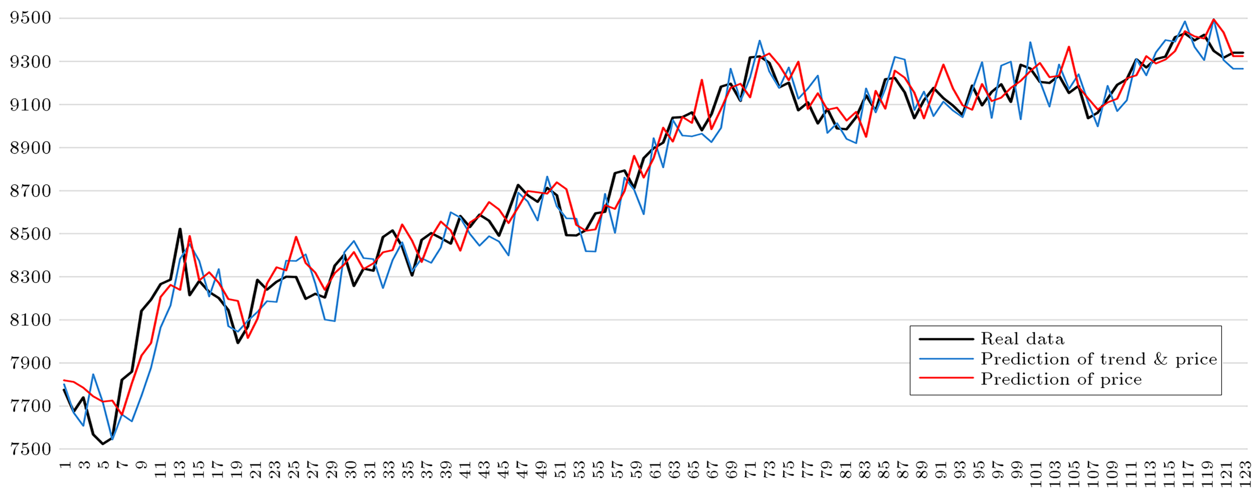


Figure 9. "Real values", "forecasted prices", and "forecasted prices considering price direction" for Dow Jones Industrial Average (DIJA) dataset.

random forecast of other models in the real-world situation.

Three categories of models with different datasets are implemented in this research including "price forecasting", "forecasting of the direction of price movement", and "price forecasting regarding the direction of price movement" (the proposed model). The obtained results are then compared to those of various trading strategies and profits. For example, DIJA test data (Table 4) are used to be tested through each of the three models. The real and forecasted prices are given in Figure 9.

Also, the results of forecasting the direction of price movement model are shown in Figure 10, in which values '1' and '−1' denote price increment and price reduction, respectively.

These forecasts are evaluated through the following trading strategies. Considering that the output of

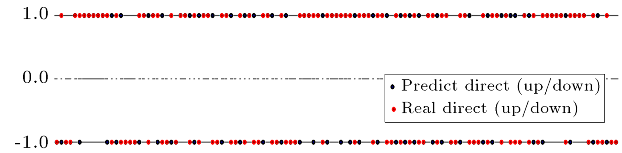


Figure 10. The real and forecasted direction of the stock price movement for Dow Jones Industrial Average (DIJA) dataset.

the forecasting model is ‘price’, one can buy new stocks as much as the prediction says if the next forecasted price increases. Otherwise, if the forecasting shows a price reduction, one can sell the existing stocks as much as the prediction forecasts. If the model output is ‘the direction of price movement’, buying and selling are done according to the forecasted direction. In other words, if the forecasting points to the ascending trend, one should buy the stocks, and vice versa. This strategy is applied to the forecasted results of the three applied models.

For instance, all the three models are employed for forecasting the trend of a deal with an initial capital of \$10,000,000. The obtained results depicted in Figure 11 show that the “price forecasting” model gained 8% profit owing to the correct forecasting of the direction of price movement, while the initial capital of the other two models decreased during this period.

5. Conclusions and future studies

Highly accurate forecasting of stock price for trading is highly important in this market and it ensures the preservation and increasing of the capital. Although some of the classical financial theories find the market unpredictable with unstable nature, in this paper, the behavior of the stock markets was forecasted using artificial intelligence models. The proposed model took into account the direction of price movement (increase or decrease) and correct stock price simultaneously to forecast the stock price and market behavior. The models in the literature mainly underscored the price forecasting and paid no attention to the next direction of the price movement. This attenuated the practicality of the models and their application to trading in stock market, which led to financial loss. To solve this obstacle, a two-stage model was proposed in this research so as to forecast the stock price considering the next direction of the price movement. In the first stage, the direction of price change was forecasted and in the second stage, this forecasted direction was added to input variables for price forecasting. Ensemble Learning (EL) was employed in the proposed model to increase forecasting accuracy. In addition, higher-level criteria were presented for evaluation and simultaneous consideration of ‘price’ and its ‘direction’ than other one-dimensional models.

The proposed model can be applied to a real-trading system, in which forecasting the direction of price movement (the first phase) is used for buying or selling, while the price forecasting (the second phase) is utilized for determining the volume of such buy/sell orders. The proposed model was implemented over several datasets and the results showed that it had a more desirable performance than other models. According to the results, it was also shown that the proposed model could be utilized as a backup system of certain decision-making in real-trading stock markets. It would be interesting to consider the following issues as future streams:

1. Using technical data for input variables in the model;

2. Employing other powerful meta-heuristic methods such as Ant Colony Optimization (ACO) to aggregate the results of the base models;

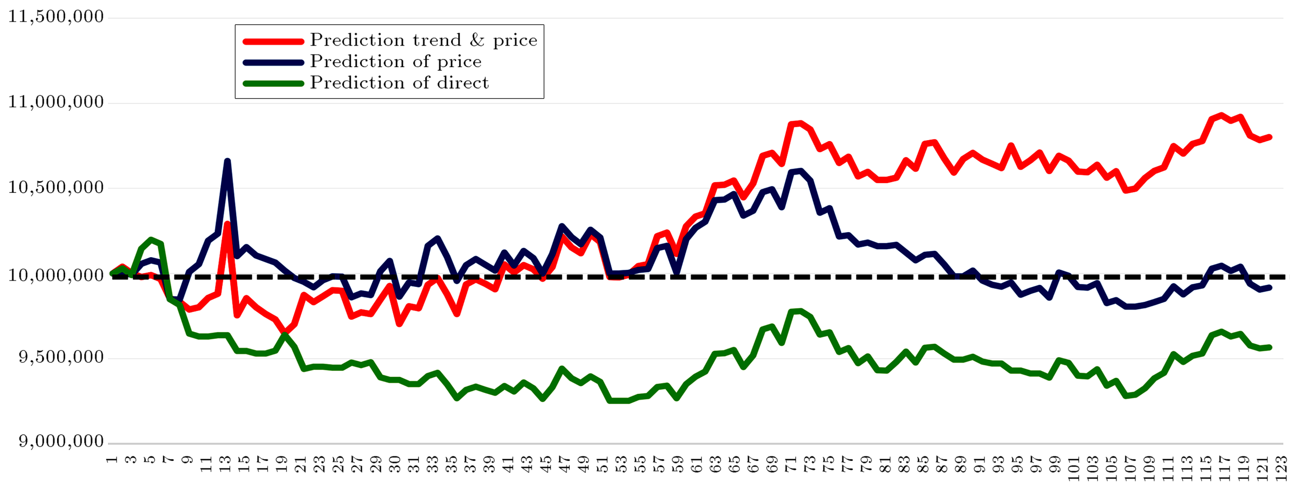


Figure 11. Results of trading with three forecasting models and strategies expressed over 120 days.

3. Simulating the results of the models in a practical way considering transaction costs.

References

1. Li, X., Yang, L., Xue, F., and Zhou, H. "Time series prediction of stock price using deep belief networks with intrinsic plasticity", In: *29th Chinese Control and Decision Conference (CCDC)*, 2 IEEE, pp. 1237–1242 (2017).
2. Kara, Y., Acar Boyacioglu, M., and Baykan, ÖK. "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul stock exchange", *Expert Systems with Applications*, **38**(5), pp. 5311–5319 (2011).
3. Guresen, E., Kayakutlu, G., and Daim, T.U. "Using artificial neural network models in stock market index prediction", *Expert Systems with Applications*, **38**(8), pp. 10389–10397 (2011).
4. Zhang, X., Zhang, Y., Wang, S., Yao, Y., Fang, B., and Yu, P.S. "Improving stock market prediction via heterogeneous information fusion", *arXiv Preprint arXiv: 1801.00588* (2018).
5. Zhang, J., Cui, S., Xu, Y., Li, Q., and Li, T. "A novel data-driven stock price trend prediction system", *Expert Systems with Applications*, **97**(1) pp. 60–69 (2018).
6. Patel, J., Shah, S., Thakkar, P., and Kotecha, K. "Predicting stock market index using fusion of machine learning techniques", *Expert Systems with Applications*, **42**(4), pp. 2162–2172 (2015).
7. Yu, L., Dai, W., and Tang, L. "A novel decomposition ensemble model with extended extreme learning machine for crude oil price forecasting", *Engineering Applications of Artificial Intelligence*, **47**, pp. 110–121 (2016).
8. Niu, M., Hu, Y., Sun, S., and Liu, Y. "A novel hybrid decomposition-ensemble model based on VMD and HGWO for container throughput forecasting", *Applied Mathematical Modelling*, **57**, pp. 163–178 (2018).
9. Ravi, V., Pradeepkumar, D., and Deb, K. "Financial time series prediction using hybrids of chaos theory, multi-layer perceptron and multi-objective evolutionary algorithms", *Swarm and Evolutionary Computation*, **36**, pp. 136–149 (2017).
10. Fama, E.F. and Malkiel, B.G. "Efficient capital markets: A review of theory and empirical work", *The Journal of Finance*, **25**(2), pp. 383–417 (1970).
11. Lo, A.W. and MacKinlay, A.C. "Stock market prices do not follow random walks: Evidence from a simple specification test", *The Review of Financial Studies*, **1**(1), pp. 41–66 (1988).
12. Tsai, C-F., Lin, Y-C., Yen, D.C., and Chen, Y-M. "Predicting stock returns by classifier ensembles", *Applied Soft Computing*, **11**(2), pp. 2452–2459 (2011).
13. Zhong, X. and Enke, D. "Forecasting daily stock market return using dimensionality reduction", *Expert Systems with Applications*, **67**, pp. 126–139 (2017).
14. Wang, J-J., Wang, J-Z., Zhang, Z-G., and Guo, S-P. "Stock index forecasting based on a hybrid model", *Omega*, **40**(6), pp. 758–766 (2012).
15. Cavalcante, R.C., Brasileiro, R.C., Souza, V.L., Nobrega, J.P., and Oliveira, A.L. "Computational intelligence and financial markets: A survey and future directions", *Expert Systems with Applications*, **55**, pp. 194–211 (2016).
16. Lin, L., Wang, F., Xie, X., and Zhong, S. "Random forests-based extreme learning machine ensemble for multi-regime time series prediction", *Expert Systems with Applications*, **83**, pp. 164–176 (2017).
17. Adebiyi, A.A., Adewumi, A.O., and Ayo, C.K. "Comparison of ARIMA and artificial neural networks models for stock price prediction", *Journal of Applied Mathematics*, **2014**, pp. 1–7 (2014).
18. Tkáč, M. and Verner, R. "Artificial neural networks in business: Two decades of research", *Applied Soft Computing*, **38**(1), pp. 788–804 (2016).
19. Arévalo, A., Niño, J., Hernández, G., and Sandoval, J. "High-frequency trading strategy based on deep neural networks", In: *International Conference on Intelligent Computing*, Springer, pp. 424–436 (2016).
20. Chong, E., Han, C., and Park, F.C. "Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies", *Expert Systems with Applications*, **83**, pp. 187–205 (2017).
21. Hassan, M.R., Nath, B., and Kirley, M. "A fusion model of HMM, ANN and GA for stock market forecasting", *Expert Systems with Applications*, **33**(1), pp. 171–180 (2007).
22. Asadi, S., Hadavandi, E., Mehmanpazir, F., and Nakhostin, M.M. "Hybridization of evolutionary Levenberg Marquardt neural networks and data preprocessing for stock market prediction", *Knowledge-Based Systems*, **35**, pp. 245–258 (2012).
23. Göçken, M., Özçalıcı, M., Boru, A., and Dosdoğru, A.T. "Integrating metaheuristics and artificial neural networks for improved stock price prediction", *Expert Systems with Applications*, **44**, pp. 320–331 (2016).
24. Chang, P-C. and Liu, C-H. "A TSK type fuzzy rule based system for stock price prediction", *Expert Systems with Applications*, **34**(1), pp. 135–144 (2008).
25. Yu, L., Chen, H., Wang, S., and Lai, K.K. "Evolving least squares support vector machines for stock market trend mining", *Evolutionary Computation, IEEE Transactions on 2009*, **13**(1), pp. 87–102 (2009).
26. Huang, C., Yang, F., and Lee, C. "The strategy of investment in the stock market using modified support vector regression model", *Scientia Iranica, Transaction E, Industrial Engineering*, **25**(3), pp. 1629–1640 (2018).

27. Esfahanipour, A. and Aghamiri, W. "Adapted neuro-fuzzy inference system on indirect approach TSK fuzzy rule base for stock market analysis", *Expert Systems with Applications*, **37**(7), pp. 4742–4748 (2010).

28. Chen, Y-S., Cheng, C-H., Chiu, C-L., and Huang, S-T. "A study of ANFIS-based multi-factor time series models for forecasting stock index", *Applied Intelligence*, **45**(2), pp. 277–292 (2016).

29. Shen, W., Guo, X., Wu, C., and Wu, D. "Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm", *Knowledge-Based Systems*, **24**(3), pp. 378–385 (2011).

30. Enke, D. and 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), pp. 636–648 (2013).

31. Khashei, M., Bijari, M., and Ardali, GAR. "Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs)", *Neurocomputing*, **72**–**128**(4), pp. 956–967 (2009).

32. Babu, C.N., Reddy, B.E., and Babu, C.N. "A moving-average filter based hybrid ARIMA-ANN model for forecasting time series data", *Applied Soft Computing*, **23**, pp. 27–38 (2014).

33. Tsai, C-F. and Chiou, Y-J. "Earnings management prediction: A pilot study of combining neural networks and decision trees", *Expert Systems with Applications*, **36**(3), pp. 7183–7191 (2009).

34. Adhikari, R., and Agrawal, R. "A combination of artificial neural network and random walk models for financial time series forecasting", *Neural Computing and Applications*, **24**(6), pp. 1441–1449 (2014).

35. Freitas, P.S. and Rodrigues, A.J. "Model combination in neural-based forecasting", *European Journal of Operational Research*, **173**(3), pp. 801–814 (2006).

36. Rather, A.M., Agarwal, A., and Sastry, V. "Recurrent neural network and a hybrid model for prediction of stock returns", *Expert Systems with Applications*, **42**(6), pp. 3234–3241 (2015).

37. Andrawis, R.R., Atiya, A.F., and El-Shishiny, H. "Forecast combinations of computational intelligence and linear models for the NN5 time series forecasting competition", *International Journal of Forecasting*, **27**(3), pp. 672–688 (2011).

38. Dietterich, T.G. "Ensemble methods in machine learning", In: *International Workshop on Multiple Classifier Systems*, Springer, pp. 1–15 (2000).

39. Yu, L., Lai, K.K., and Wang, S. "Multistage RBF neural network ensemble learning for exchange rates forecasting", *Neurocomputing*, **71**(16–18), pp. 3295–3302 (2008).

40. Xiao, Y., Xiao, J., Lu, F., and Wang, S. "Ensemble ANNs-PSO-GA approach for day-ahead stock E-exchange prices forecasting", *International Journal of Computational Intelligence Systems*, **7**(2), pp. 272–290 (2013).

41. Ballings, M., Van den Poel, D., Hespeels, N., and Gryp, R. "Evaluating multiple classifiers for stock price direction prediction", *Expert Systems with Applications*, **42**(20), pp. 7046–7056 (2015).

42. Maknickienė, N. "Prediction capabilities of evolino RNN ensembles", In: *Computational Intelligence*, Edn., Springer, pp. 473–485 (2016).

43. Atsalakis, G.S. and Valavanis, K.P. "Surveying stock market forecasting techniques-Part II: Soft computing methods", *Expert Systems with Applications*, **36**(3), pp. 5932–5941 (2009).

44. Cheng, C., Xu, W., and Wang, J. "A comparison of ensemble methods in financial market prediction", In: *Computational Sciences and Optimization (CSO), 2012 Fifth International Joint Conference on*, IEEE, pp. 755–759 (2012).

45. Atsalakis, G.S. and Valavanis, K.P. "Forecasting stock market short-term trends using a neuro-fuzzy based methodology", *Expert Systems with Applications*, **36**(7), pp. 10696–10707 (2009).

46. Wang, G., Hao, J., Ma, J., and Jiang, H. "A comparative assessment of ensemble learning for credit scoring", *Expert Systems with Applications*, **38**(1), pp. 223–230 (2011).

47. Cubiles-De-La-Vega, M-D., Blanco-Oliver, A., Pino-Mejías, R., and Lara-Rubio, J. "Improving the management of microfinance institutions by using credit scoring models based on statistical learning techniques", *Expert Systems with Applications*, **40**(17), pp. 6910–6917 (2013).

48. Kantelhardt, J.W., Zschiegner, S.A., Koscielny-Bunde, E., Havlin, S., Bunde, A., and Stanley, H.E. "Multifractal detrended fluctuation analysis of nonstationary time series", *Physica A: Statistical Mechanics and its Applications*, **316**(1), pp. 87–114 (2002).

49. Lei, L. "Wavelet neural network prediction method of stock price trend based on rough set attribute reduction", *Applied Soft Computing*, **62**, pp. 923–932 (2018).

50. Ferreira, T.A., Vasconcelos, G.C., and Adeodato, P.J. "A new intelligent system methodology for time series forecasting with artificial neural networks", *Neural Processing Letters*, **28**(2), pp. 113–129 (2008).

51. De Oliveira, F.A., Nobre, C.N., and Zárate, L.E. "Applying artificial neural networks to prediction of stock price and improvement of the directional prediction index-Case study of PETR4, Petrobras, Brazil", *Expert Systems with Applications*, **40**(18), pp. 7596–7606 (2013).

52. Yazdani, M., Zandieh, M., Tavakkoli-Moghaddam, R., and Jolai, F. "Two meta-heuristic algorithms for the dual-resource constrained flexible job-shop scheduling problem", *Scientia Iranica, Transactions E, Industrial Engineering*, **22**(3), p. 1242 (2015).

Biographies

Mohammad Taghi Faghihi Nezhad is an Assistant Professor at the Department of Information Technology, Faculty of Engineering, Payame Noor University, Iran. He received a PhD in Information Technology Engineering. His research interests focus on intelligent trading systems and stock prediction.

Behrouz Minaei Bidgoli is an Associate Professor at the School of Computer Engineering at Iran University of Science and Technology. He leads the Data Mining Lab that researches various areas in artificial intelligence and data mining, including text mining, web information extraction, and natural language processing.