Development of an Ensemble Learning-based intelligent model for Stock Market Forecasting

Mohammad_Taghi Faghihi_Nezhad, Behrouz Minaei_Bidgoli

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

Abstract

The use of artificial intelligence-based models have 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 the accuracy of the results and increasing the forecasting efficiency. Another challenge, which is a prerequisite for making decision and using the results of the forecast for profitability of transactions, is to forecast the trend of stock price movements in forecasting price. 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, in order to consider the direction of price change 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) technique 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 compared to other models used in the literature.

Keywords: Forecasting the direction of price movement, Ensemble learning, Bagging, Forecasting stock price, Evolutionary computing, Intelligent trading system

* Corresponding author

Email: mfganghi@pnu.ac.ir (Mohammad-Taghi Faghihi-Nezhad), b_minaei@iust.ac.ir (Behrouz Minaei)
1. Introduction

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

Improvement in 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 trend of the stock price (the direction of stock price movement)[5]. The investor should have an accurate forecasting of the price change (increase or decrease) for trading. Most of the proposed models in this area have been developed for price forecasting. In these researches, the performance is evaluated through several criteria such as Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) [6-9]. These criteria are able to evaluate the proximity of the forecasted price to the real price, but they are not able to evaluate the model in forecasting the stock price trends.

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

![Fig. 1. To be placed here](image)

To determine the importance of forecast the direction of price movement shown in Fig. 1, consider the price of the fourth day that is 145.87$. A trader predicts a price of 145.9$, which shows that he predicts an increment in price for the next (fifth) day. He gets the buy position, while the real price of the next day would be $145.63 (lower than predicted) and he would face a loss on the trade. The main reason for such a forfeiture in this trading is that forecasting the direction of price movement was wrong, while the price forecasting had only 0.18% error.

In order to avoid similar losses happening again, it is necessary to forecast the price considering forecasting the direction of the stock price movement. To overcome the two above-mentioned challenges, the proposed model uses an ensemble-learning algorithm equipped with intelligent-based models and meta-heuristic technique to maximize the quality of the prediction results.

In addition, 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 to consider both direction of the stock price movement and 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 ensemble learning using intelligence-based models. The proposed model is evaluated and thoroughly compared to the other models in Section 4 and experimental results are then presented in this section. Finally, conclusions are given in Section 5.

1. 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 made based on results of new information. According to this hypothesis, it is impossible to obtain higher returns through intelligently 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 expectations and information of all participants in the market are well reflected using prices, fluctuation of prices cannot be forecasted. 2) Another hypothesis which is compatible with the efficient market hypothesis is “random walk (RW)” that says the trend of volatilities in stock market prices are random and thus cannot be forecasted. However, some recent studies reject the random walk behavior of stock prices [11]. Also, the application of artificial intelligence in financial area has strengthened this idea that the market might not be always efficient and one can forecast the future prices from historical data by means of 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 for forecasting stock market by using historical data. One approach categorizes them into “linear” and “nonlinear” techniques, while the other approach considers them as “statistical” and “machine learning” [15]. Among these categories, intelligent and classical ones are suitable. In classic forecasting approach, it is assumed that the future value of the price follows the linear trend of the past values. The autoregressive moving average (ARIMA), autoregressive conditional heteroscedasticity (GARCH) and regression belong to this class. The artificial neural networks (ANNs), fuzzy logic, support vector machines (SVMs), ensemble learning (EL) and meta-heuristic algorithms all belong to intelligent techniques [9, 15]. These methods, unlike the classic ones, are capable to obtain a nonlinear relationship between input variables without having information about the statistical distribution of these inputs.
2.1 Intelligent models

The intelligent models for time series forecasting have some advantages and disadvantages [16]. The conducted comparisons show that the intelligent models can overcome limitations of linear models, where they can better extract a pattern from data with a 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 ensemble learning models shown in Fig 2.

Fig. 2. To be placed here

The first group in Fig 2. uses a single model for forecasting which itself includes two types: 1) the models that employ one technique and 2) the models that use multi techniques to forecast. According to studies conducted by Atsalakis [43] and Tkáč [18] among different intelligent models, ANNs techniques have been applied more than other techniques, since they had better performance [18]. The ANNs are used in these researches to forecast the stock price [23] or the direction of stock price movement [2]. Despite the stock market forecasting is complex, it is shown that ANNs with only a hidden layer can model such a complex system with an acceptable accuracy.

Although using ANNs has led to increasing in forecasting accuracy compared to the classic models, however, 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 improve the forecasting accuracy is using multi-technique models. For example, using the ANNs combined with meta-heuristic algorithms, which is ordinarily employed to overcome the mentioned problems and improvement of NN training [21, 22, 29]. Another approach is applying Neuro-fuzzy techniques equipped with meta-heuristic algorithms [24, 28].

The second group in Fig 2. is named hybrid forecasting models combining different single models for attaining higher accuracy results. By integrating the ANN as nonlinear technique and ARIMA as a linear method, one can benefit their both advantages [31]. Also, a hybrid model comprising RW for exploring linear patterns and two NN models for uncovering non-linear patterns has been proposed in [34]. Tsai [33] obtained a higher rate of forecasting accuracy through combining the ANN and DT models. Wang [14] proposed a hybrid model combining ESM, ARIMA, and BPNN, where the weight of each single model was determined by the GA. Andrawis [37] combined computational intelligence and linear models with methods such as mean, trimmed mean and winsorized mean. In these types of researches, the hybrid models were compared with the existing single models in the
literature and the obtained results demonstrated that such combined models outperform single models in forecasting accuracy [6, 9, 32, 36].

The third group in Fig 2. is the models based on ensemble learning algorithm. These algorithms belong to computational intelligence approach that integrates a set of base learners into a single model [44]. It is also shown that the necessary and sufficient conditions for an ensemble learner to have higher accuracy than its base learners dependent on being accurate and diversity in categorizing the members. Here, the “accuracy” means “better than random prediction” and “diverse” means the learners make uncorrelated errors.

Tsai [12] examined two types of ensembles, i.e., ‘homogeneous’ classifier ensembles and ‘heterogeneous’ classifier ensembles for prediction accuracy of stock returns. The results indicated that the homogeneous multiple classifiers using NNs outperform the single classifiers. Lin [16] proposed the RF-based extreme learning machine (ELM) ensemble model in order to achieve the accuracy, stability, and efficiency simultaneously in time series forecasting. Ballings [41] investigated ensemble methods against single models in the stock forecasting and suggested that the novel studies in this domain should be included ensembles algorithms.

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

2.2 Problem definition

By introducing various methods, researchers are trying to demonstrate the ability of the proposed models to increase the accuracy of stock market forecasting. According to the results of conducted literature review in this research, the ensemble learning algorithms are more accurate, more reliable and stable to forecast the stock market. Therefore, the proposed forecasting model should use ensemble-learning algorithms to maximize the performance of the predictive result. In addition, in order 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 paid attention (Fig. 3).

In Fig. 3, the Apple's stock price is displayed along with the two-price forecast. The calculated MAPE for the first forecast is 0.46%. The price of 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 that
leads to a loss of $1.3. According to this forecast, a loss of $5.9 in 14 days will eventually happen, 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, by carrying the daily trade out, a profit of $ 2.3 in 14 days will take place, based on this forecast. The reason for obtaining this amount of profit in accurate forecast about the direction of stock price movement and the price forecast itself. While in the first forecast, as it can be observed in Fig. 3, the view is to follow the sequence of the previous movement.

Fig. 3. To be placed here

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 ensemble learning through using intelligent-based models as well as meta-heuristic algorithms in both stages that can maximize the performance of the forecasted results.

2. Development of an Intelligent Ensemble-based Model for Stock Price Forecasting

Various studies in applying training methods show that there is no specific training algorithm that can be the most accurate and best for all predictions. To overcome such obstacle, EL algorithms have been developed to a large extent so as to reduce the error, which is the main motivation for their developments.

The basic assumption of this methodology is that in EL, the probability of prediction error in an unknown sample is much less than the predictions error in an individual model. In comparison with common machine learning methods which try to learn a hypothesis from training data, in EL models, several learners are trained to attain the greatest possible accuracy and also 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, 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 to the following four proposed methods:

1- Using different training datasets for training of the base learner; through resampling methods in which sub-set of the original training data is selected randomly and will be replaced by the original training dataset.
2. In order to ensure that the boundaries are different diverse, in addition to using different training data, unstable models are used as base models because they can make different decision boundaries, even with the low change in their training parameters [47].

3. Another way to achieve a diversity in parameters is to use different models. For example, a set of multi-layer perceptron neural networks can be trained with initial weights, a number of layers and nodes, different error criteria, and so on. Setting such parameters can control individual model instability and ultimately diversify them. The ability to control the unstable ANNs has become the ideal candidate for using EL algorithms.

4. By using different features; input space is divided into different sub-sets of original features that might overlap and each sub-set is given to a model as an input. Through this method, every base learner explores some part of knowledge and also diversity in using features make the EL algorithms to yield better results.

Bagging as one of the simplest EL algorithms is offered to improve the performance of predictions 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. Lack of using unstable predictor leads to collection creation of almost identical predictors that makes no longer improvement in individual predictor’s efficiency. 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 big changes in the result of prediction[47]. After training different base learners, in order to achieve final prediction, the obtained results from 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 training step and then compared to other learners. The effect of each learner on the result of the final forecast can be considered as an optimization problem. The goal of this optimization problem is to determine the best weights for each learner in such a way that the prediction accuracy of the test data is maximized. In this research, two well-known meta-heuristic algorithms, i.e., PSO and GA are employed to tackle this optimization problem.

3.1 The proposed model

The existing challenge of models was presented in the previous sections that was paying no attention to stock price and direction of the price movement, simultaneously. In order to tackle such difficulty, in this subsection, a new stock price forecasting model is introduced by considering the price and the direction of price movement, concurrently. The proposed model includes two dependent stage. Firstly, 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 in the next time. In order 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 diverse as much as possible so as to achieve the appropriate accuracy. The diversity is attained through 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 weight 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 stock price movement is forecasted for the next time. 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 de-trend and stationary as possible so that the hidden pattern in the series can be extracted more accurately [48]. Differentiation and logarithmic conversion can discover more the knowledge in the data. The first difference of a time series creates a new time series whose values are different of two successive values of the initial time series is the series of changes from one period to the next:

\[ \nabla x_t = x_t - x_{t-1} \]  

(1)

Where \( x_t \) denotes the value of the time series \( x_t \) in period \( t \), 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 changes in price.

The value of \( x \) in period \( t \) is auto-correlated with respect to its value at earlier periods, where the \( n \)-th element of the series with \( k \) lag is entered into 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, those price data closed 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 “close” price of today and the previous day.
Data preparation and formation of new time series is performed, where the value of the new time series is obtained by one times differencing of 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 on training data. One NN is created and trained N times with N bootstraps until N base models are obtained. In the following, the training data is entered into each of the trained base models and its output is compared with the target output in order 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 (forecasted the direction of price movement) are added to the matrix result. After applying training data to all trained models and completing the results matrix, this matrix is aggregated with four methods and the best weigh 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. The obtained weights from the method having the most accuracy are finally selected.

By considering the importance of the learner's weight 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 in computation of the final output. Therefore, every particle of a vector equals to 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 means that the performance of the learner in teamwork to reach the least possible error for all training data obtained by particle-related weight combination. For example, for a particle with weights of 0.5, 0.3 and 0.2, the learner for a specific sample yields 59, 65 and 62, as outputs, so the final output will be $61.4 = 0.5 \times 59 + 0.3 \times 65 + 0.2 \times 62$. 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 are equal to the number of base models obtained in the previous steps. The initial amounts of weights are randomly generated in each chromosome. Then, chromosomes are arranged based on their performance (exactly similar
to PSO). To generate the next generation, the selection is conducted through roulette wheel mechanism. In addition, 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 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 implementing process of the first stage of the proposed model is shown in Fig. 4.

Fig. 4. To be placed here

3.1.2. The Second Stage (Price Forecasting)

After termination of the model’s first stage, its output, i.e., the direction of price upward/downward movement in stock market is obtained. In the second stage, by adding this feature to the existing ones (new dataset), a model is trained with the 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, however different in usage. The evaluation criteria for base models and aggregation methods at this stage are different, where instead of evaluating the accuracy of the results in correct 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, in order to improve the accuracy and further assurance, the results of the base models are aggregated with 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 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 at this stage are the same with those in the previous stage. The only difference is the criterion for evaluating and selecting the optimal weight of the vectors in the matrix, which is MAPE. The same issue exists about the algorithms used in this stage, where 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 Fig. 5.

Fig. 5. To be placed here

3. 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 with other researches.

4.1. Datasets

In order to compare the results of the proposed model with accredited paper, the same datasets in the literature are used [22, 24, 27]. These data include different indices of the world’s validated stock exchanges showing the changes of the prices general level 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.

Table 1. To be placed here

4.2. Evaluation criteria

Since the aim of this paper is to improve forecasting the direction of price movement and the price itself, simultaneously, the used criteria for evaluating the outcomes should measure these two standards. The first criterion used to compare the models is MAPE [6-8]. This criterion, the absolute difference between the real and predicted amounts is divided to the real amount at first. 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.

$$MAPE = 100 \times \frac{1}{N} \sum_{i=1}^{N} \frac{|y_i - p_i|}{y_i}$$

(2)
The Prediction On Change In Direction (POCID) presenting in Equations (3) and (4) shows the calculation of the direction change prediction. The model accuracy in direction prediction of the price movement besides the proper prediction of the price, plays a leading role in gaining the profit. The POCID criterion is ranged over the interval [0,100]. The closer the value of POCID to 100, the higher accuracy of the prediction [49].

\[
POICD = 100 \times \frac{1}{N} \sum_{i=0}^{N} D_i
\]

(3)

\[
D_i = \begin{cases} 
1, & \text{if } (y_i - y_{i+1})(p_i - p_{i+1}) > 0 \\
0, & \text{otherwise} 
\end{cases}
\]

(4)

The third criterion is Theil's U statistic (Eq. 5), which compares the model performance with RW model. If the Theil's U is equal to 1, the model performance is equivalent to RW, but if it is bigger 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].

\[
U_{\text{of Tail}} = \frac{\sum_{i=1}^{N} (y_i - p_i)^2}{\sum_{i=1}^{N} (y_i - y_{i+1})^2}
\]

(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 less than 1 and close to 0, indicates better forecasting accuracy [50].

\[
ARV = \frac{\sum_{i=1}^{N} (y_i - p_i)^2}{\sum_{i=1}^{N} (\bar{y} - p_i)^2}
\]

(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 price, as well as the volume and the number of trading in a period. The second type is the technical variables that derives from the price data using different formulas. Some researchers have used the price variables [9], while some others employed technical variables [2, 6].

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

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

In the proposed model, the base learners are composed of a three-layer feedforward neural network, while the number of inputs is equal to the number of used lags. Several factors in 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 using data 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 lot of times. To solve this problem and to 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.

Table 2. To be placed here

The prediction output vector obtained from the first stage is added to other price variables, which creates 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 get the best results.

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

Table 3. To be placed here

As shown in Table 3, using Dow Jones dataset yields the (near) optimal MAPE, 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 are 100 and WAV is used as the aggregations method of base models. Fig. 6 shows a comparison between the real and forecasted values in the proposed model.
Fig. 6. To be placed here
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 by trial and error, the best parameters are selected for each dataset that are shown in Table 4.

Table 4. To be placed here

4.4. Comparison of the proposed model with 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 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 compared to other models in most cases.

Table 5. To be placed here
Considering the price forecasting criteria and the direction of price movement in stock market is of the main advantages of the proposed model in this paper. The obtained results are compared to the results of Asadi’s model [22], in which POICD criterion was used for prediction of changes on the same datasets. The comparison that is shown in Table 6 demonstrates that the proposed model outperforms the Asadi’s one in terms of accuracy.

Table 6. To be placed here
The results of Theil's U evaluation show that the proposed model also performs better than RW model regarding ARV criterion in all datasets. Taking MAPE and POICD into account, the proposed model, in most cases, has better forecasted accuracy than them. The improved amount is shown in imp% column of Table 6. For example, for Dow Jones dataset, MAPE obtained from Asadi’s model and the proposed model in this research are 1.41 and 1.126, respectively that shows %18 improvement in price forecasting. However, the proposed model has shown %5 improvement in terms of the direction of price movement than this model.

In Figs. 7 and 8 comparisons of six indices between the proposed model and Asadi’s one in terms of MAPE and POICD criteria are shown.

Fig. 7. To be placed here
4.5. Discussion

As mentioned earlier, forecasting models have been divided into two groups including price forecasting and forecasting the direction of price movement. In addition, it was clarified that the use of models forecast the price regardless of price trends, may 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. A considerable amount of effort is made in this research so as to propose a model for forecasting the price considering price trend where better than random forecast than other models in real-world situation.

Three categories of models with different datasets are implemented in this research including, “price forecasting”, “forecasting the direction of price movement” and “price forecasting regarding the direction of price movement” (the proposed model). The obtained results are then compared through 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 depicted in Fig. 9.

Also, the results of forecasting the direction of price movement model are shown in Fig. 10, in which the value ‘1’ signifies the price increment and ‘-1’ denotes the price reduction.

These forecasting are evaluated through the following trading strategies. If the output of the forecasting model is ‘price’, one can buy the new stock as much as the prediction says, if the next forecasted price is being increased. Otherwise, if the forecasting shows reduction in price, 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 is done according to the forecasted direction. In better words, if the forecasting shows ascending trend, one should buy the stocks and vice versa. This strategy is implemented over the forecasted results of the three applied models.

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

4. Fig. 11. To be placed here

5. Conclusions and future studies

The high-accurate forecasting of stock price for trading is highly important in this market, leading to the preservation and increasing of the capital. Despite some of the classic 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, in order 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 caused a reduction in their models’ practicality and their application for trading in stock market which lead 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. The ensemble learning (EL) was employed in the proposed model to increase the accuracy of the forecasting. In addition, it presents higher-level criteria for evaluation in addition to simultaneous consideration of ‘price’ and ‘the direction’ than other one-dimensional models.

The proposed model, can be applied in a real-trading system, where 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. The proposed model was implemented over several datasets and the results showed that it has more desirable performance than other models and actually outperforms them. 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.
3. Simulating the results of the models in a practical way, considering transaction costs.
6. References

1. Li X, Yang L, Xue F, Zhou H: “Time series prediction of stock price using deep belief networks with intrinsic plasticity”.
   In: Control And Decision Conference (CCDC), 2017 29th Chinese: 2017: IEEE; 2017: 1237-1242.
2. Kara Y, Acar Boyacioglu M, 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 2011, 38(5):5311-5319.
3. Guresen E, Kayakutlu G, Daim TU: “Using artificial neural network models in stock market index prediction”.
   Expert Systems with Applications 2011, 38(8):10389-10397.
4. Zhang X, Zhang Y, Wang S, Yao Y, Fang B, Yu PS: “Improving Stock Market Prediction Via Heterogeneous Information Fusion”.
   arXiv preprint arXiv:180100588 2018.
5. Zhang J, Cui S, Xu Y, Li Q, Li T: “A novel data-driven stock price trend prediction system”.
   Expert Systems with Applications 2018, 97(1):60-69.
6. Patel J, Shah S, Thakkar P, Kotecha K: “Predicting stock market index using fusion of machine learning techniques”.
   Expert Systems with Applications 2015, 42(4):2162-2172.
7. Yu L, Dai W, Tang L: “A novel decomposition ensemble model with extended extreme learning machine for crude oil price forecasting”.
   Engineering Applications of Artificial Intelligence 2016, 47:110-121.
8. Niu M, Hu Y, Sun S, Liu Y: “A novel hybrid decomposition-ensemble model based on VMD and HGWO for container throughput forecasting”.
   Applied Mathematical Modelling 2018.
9. Ravi V, Pradeepkumar D, Deb K: “Financial time series prediction using hybrids of chaos theory, multi-layer perceptron and multi-objective evolutionary algorithms”.
   Swarm and Evolutionary Computation 2017, 36:136-149.
10. Fama EF, Malkiel BG: “Efficient capital markets: A review of theory and empirical work”.
    The Journal of Finance 1970, 25(2):383-417.
11. Lo AW, MacKinlay AC: “Stock market prices do not follow random walks: Evidence from a simple specification test”.
    The review of financial studies 1988, 1(1):41-66.
12. Tsai C-F, Lin Y-C, Yen DC, Chen Y-M: “Predicting stock returns by classifier ensembles”.
    Applied Soft Computing 2011, 11(2):2452-2459.
13. Zhong X, Enke D: “Forecasting daily stock market return using dimensionality reduction”.
    Expert Systems with Applications 2017, 67:126-139.
14. Wang J-J, Wang J-Z, Zhang Z-G, Guo S-P: “Stock index forecasting based on a hybrid model”.
    Omega 2012, 40(6):758-766.
15. Cavalcante RC, Brasilheiro RC, Souza VL, Nobrega JP, Oliveira AL: “Computational Intelligence and Financial Markets: A Survey and Future Directions”.
    Expert Systems with Applications 2016, 55:194-211.
16. Lin L, Wang F, Xie X, Zhong S: “Random forests-based extreme learning machine ensemble for multi-regime time series prediction”.
    Expert Systems with Applications 2017, 83:164-176.
17. Adebiyi AA, Adewumi AO, Ayo CK: “Comparison of ARIMA and artificial neural networks models for stock price prediction”.
    Journal of Applied Mathematics 2014, 2014.
18. Tkáč M, Verner R: “Artificial neural networks in business: Two decades of research”.
    Applied Soft Computing 2016, 38(1):788-804.
19. Arévalo A, Niño J, Hernández G, Sandoval J: “High-frequency trading strategy based on deep neural networks”.
    In: International Conference on Intelligent Computing: 2016: Springer; 2016: 424-436.
20. Chong E, Han C, Park FC: “Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, Data Representations, and Case Studies”.
    Expert Systems with Applications 2017.
21. Hassan MR, Nath B, Kirley M: “A fusion model of HMM, ANN and GA for stock market forecasting”, Expert Systems with Applications 2007, 33(1):171-180.
22. Asadi S, Hadavandi E, Mehanparaz F, Nakhhostin MM: “Hybridization of evolutionary Levenberg Marquardt neural networks and data pre-processing for stock market prediction”, Knowledge-Based Systems 2012, 35:245-258.
23. Göçken M, Özçalıcı M, Boru A, Dosdoğru AT: “Integrating metaheuristics and Artificial Neural Networks for improved stock price prediction”, Expert Systems with Applications 2016, 44:320-331.
24. Chang P-C, Liu C-H: “A TSK type fuzzy rule based system for stock price prediction”, Expert Systems with Applications 2008, 34(1):135-144.
25. Yu L, Chen H, Wang S, Lai KK: “Evolving least squares support vector machines for stock market trend mining”, Evolutionary Computation, IEEE Transactions on 2009, 13(1):87-102.
26. Huang C, Yang F, Lee C: “The strategy of investment in the stock market using modified support vector regression model”, Scientia Iranica Transaction E, Industrial Engineering 2018, 25(3):1629-1640.
27. Esfahanipour A, Aghamiri W: “Adapted neuro-fuzzy inference system on indirect approach TSK fuzzy rule base for stock market analysis”, Expert Systems with Applications 2010, 37(7):4742-4748.
28. Chen Y-S, Cheng C-H, Chiu C-L, Huang S-T: “A study of ANFIS-based multi-factor time series models for forecasting stock index”, Applied Intelligence 2016, 45(2):277-292.
29. Shen W, Guo X, Wu C, Wu D: “Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm”, Knowledge-Based Systems 2011, 24(3):378-385.
30. 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 2013, 19(4):636-648.
31. Khashei M, Bijari M, Ardali GAR: “Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs)”, Neurocomputing 2009, 72-128(4):956-967.
32. Babu CN, Reddy BE, Babu CN: “A moving-average filter based hybrid ARIMA–ANN model for forecasting time series data”, Applied Soft Computing 2014, 23:27-38.
33. Tsai C-F, Chiou Y-J: “Earnings management prediction: A pilot study of combining neural networks and decision trees”, Expert systems with applications 2009, 36(3):7183-7191.
34. Adhikari R, Agrawal R: “A combination of artificial neural network and random walk models for financial time series forecasting”, Neural Computing and Applications 2014, 24(6):1441-1449.
35. Freitas PS, Rodrigues AJ: “Model combination in neural-based forecasting”, European Journal of Operational Research 2006, 173(3):801-814.
36. Rather AM, Agarwal A, Sastry V: “Recurrent neural network and a hybrid model for prediction of stock returns”, Expert Systems with Applications 2015, 42(6):3234-3241.
37. Andrawis RR, Atiya AF, El-Shishiny H: “Forecast combinations of computational intelligence and linear models for the NN5 time series forecasting competition”, International Journal of Forecasting 2011, 27(3):672-688.
38. Dietterich TG: “Ensemble methods in machine learning”, In: International workshop on multiple classifier systems: 2000: Springer: 2000: 1-15.
39. Yu L, Lai KK, Wang S: “Multistage RBF neural network ensemble learning for exchange rates forecasting”, Neurocomputing 2008, 71(16-18):3295-3302.
40. Xiao Y, Xiao J, Lu F, Wang S: “Ensemble ANNs–PSO-GA Approach for Day-ahead Stock E-exchange Prices Forecasting”, International Journal of Computational Intelligence Systems 2013.
41. Ballings M, Van den Poel D, Hespeels N, Gryp R: “Evaluating multiple classifiers for stock price direction prediction”, Expert Systems with Applications 2015, 42(20):7046-7056.
42. Maknickienė N: “Prediction Capabilities of Evolino RNN Ensembles”, In: Computational Intelligence. edn.: Springer: 2016: 473-485.
43. Atsalakis GS, Valavanis KP: “Surveying stock market forecasting techniques—Part II: Soft computing methods”, Expert Systems with Applications 2009, 36(3):5932-5941.
44. Cheng C, Xu W, Wang J: “A comparison of ensemble methods in financial market prediction”, In: Computational Sciences and Optimization (CSO), 2012 Fifth International Joint Conference on: 2012: IEEE; 2012: 755-759.
45. Atsalakis GS, Valavanis KP: “Forecasting stock market short-term trends using a neuro-fuzzy based methodology”, *Expert systems with Applications* 2009, 36(7):10696-10707.

46. Wang G, Hao J, Ma J, Jiang H: “A comparative assessment of ensemble learning for credit scoring”, *Expert systems with applications* 2011, 38(1):223-230.

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

48. Kantelhardt JW, Zschiegner SA, Koscielny-Bunde E, Havlin S, Bunde A, Stanley HE: “Multifractal detrended fluctuation analysis of nonstationary time series”, *Physica A: Statistical Mechanics and its Applications* 2002, 316(1):87-114.

49. Lei L: “Wavelet neural network prediction method of stock price trend based on rough set attribute reduction”, *Applied Soft Computing* 2018, 62:923-932.

50. Ferreira TA, Vasconcelos GC, Adeodato PJ: “A new intelligent system methodology for time series forecasting with artificial neural networks”, *Neural Processing Letters* 2008, 28(2):113-129.

51. de Oliveira FA, Nobre CN, Zárate LE: “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* 2013, 40(2):7596-7606.

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

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

Behrouz Minaei-Bidgoli is Associate Professor in the School of Computer Engineering at Iran University of Science and Technology. He leads the Data Mining Lab that does research on various areas in artificial intelligence and data mining, including text mining, web information extraction, and natural language processing.
Fig. 1. The price and direction of the price movement
Fig. 2. Classification of intelligent models
Fig. 3. The real price besides two-forecasted price
Fig. 4. The implementing process of the first stage of the proposed model
Fig. 5. The implementing process of the second stage of the proposed model
Fig. 6. Comparison of real and forecasted values in the proposed model
Fig. 7. Comparison of two models considering MAPE criterion
Fig. 8. Comparison of two models considering POICD criterion
Fig. 9. “Real values”, “forecasted prices” and “forecasted prices considering price direction” for DIJA dataset
Fig. 10. The real and forecasted direction of the stock price movement for DIJA dataset
Fig. 11. Results of trading with three forecasting models and strategies expressed over 120 days
Table 1
Descriptions of stock indices

Table 2
The first stage’s results for implementation of the training data in the proposed model

Table 3
The second stage’s results for implementation of the test data in the proposed model

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

Table 5
Comparing the results of the proposed model and other models

Table 6
Comparing the results of the proposed model with the Asadi's one
**Fig. 1.** The price and direction of the price movement
## Intelligent models

| Single models | Single-technique | Multi-technique | Hybrid models | Ensemble learning model |
|---------------|------------------|-----------------|---------------|------------------------|
|               | NN: (Kara, 2011) [2], (Zhong, 2017) [13] | NN & GA: (Hassan, 2009) [21], (Asadi, 2012) [22], (Göçken, 2016) [23] | ARIMA, NN: (Khashei, 2009) [31], (Babu, 2014) [32] | Dietterich, 2000) [38], (Yu, 2008) [39], (Tsai, 2011) [12], (Xiao, 2013) [40], (Ballings, 2015) [41], (Maknickienè, 2016) [42], (Lin, 2017) [16] |
|               | SVM: (Kara, 2011) [2] | Neuro Fuzzy & SA: (Chang and Liu, 2008) [24] | NN, DT: (Tsai and Chiou, 2009) [33] |                     |
|               | DEEP NN: (Arévalo, 2016) [19], (Chong, 2017) [20] | SVM & GA: (Yu et al., 2009) [25] | ARIMA, ESM, NN: (Wang, 2012) [14] |                     |
|               |                   | Neuro Fuzzy & SVR: Huang, Yang, & Lee, 2018 [26] | RW, FANN, EANN: (Adhikari and Agrawal, 2014) [34] |                     |
|               |                   | Neuro Fuzzy & FC: (Esfahanipour, 2010) [27], (Chen, 2016) [28] | SVR, RF: (Patel, 2015) [36] |                     |
|               |                   | RBF & K-means & AFSA: (Shen et al., 2011) [29] | RBF, RW: (Freitas and Rodrigues, 2006) [35] |                     |
|               |                   | FS, FC & Fuzzy NN: (Enke, 2013) [30], (Chen, 2016) [28] | Linear and nonlinear Model: (Rather, 2015) [36] |                     |
|               |                   | Chaos theory & MLP & MOPSO & NSGA-II (Ravi, 2017) [9] | Computational Intelligence, Linear Model: (Andrawis, 2011) [37] |                     |

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

**Fig. 2.** Classification of intelligent models
Fig. 3. The real price besides two-forecasted price
Fig. 4. The implementing process of the first stage of the proposed model
Fig. 5. The implementing process of the second stage of the proposed model
Fig. 6. Comparison of real and forecasted values in the proposed model
Fig. 7. Comparison of two models considering MAPE criterion
Fig. 8. Comparison of two models considering POICD criterion
Fig. 9. “Real values”, “forecasted prices” and “forecasted prices considering price direction” for DIJA dataset
Fig. 10. The real and forecasted direction of the stock price movement for DIJA dataset
Fig. 11. Results of trading with three forecasting models and strategies expressed over 120 days.
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            |
Table 2
The first stage’s results for implementation of 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                    | traindm | trainlm | trainlm | traindm | 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
The second stage’s results for implementation of the test data in the proposed model

| Data Name          | DIJA | TSE  | TEPIX | TIT50C | TII | TIFG |
|--------------------|------|------|-------|--------|-----|------|
| Number of neurons in the neural network | 9    | 9    | 7     | 7      | 5   | 5    |
| The use amount of data from each bootstrap | 100% | 95%  | 100%  | 100%   | 95% | 100% |
| The aggregation method | WAV  | PSO  | GA    | WAV    | PSO | PSO |

|               | DIJA | TSE  | TEPIX | TIT50C | TII | TIFG |
|---------------|------|------|-------|--------|-----|------|
| 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|
Table 4
The values of used parameters in employed meta-heuristic algorithms for each dataset

| Data                  | DJIA Stage I | DJIA Stage II | TSE Stage I | TSE Stage II | TEPIX Stage I | TEPIX Stage II | Top 50 companies Stage I | Top 50 companies Stage II | Industry index Stage I | Industry index Stage II | Financial group Stage I | Financial group Stage II |
|----------------------|--------------|---------------|-------------|--------------|---------------|---------------|--------------------------|--------------------------|-------------------------|--------------------------|-------------------------|--------------------------|
| Stage                |              |               |             |              |               |               |                          |                          |                         |                          |                         |                          |
| GA Parameters        |              |               |             |              |               |               |                          |                          |                         |                          |                         |                          |
| Population Size      | 150          | 150           | 200         | 200          | 200           | 200           | 150                      | 150                      | 200                     | 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           | 200                      | 200                      | 200                     | 150                      | 100                     | 100                     |
| Number of iterations | 1200         | 1000          | 700         | 700          | 800           | 700           | 800                      | 700                      | 600                     | 500                      | 600                     | 500                     |
Table 5
Comparing 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 (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 (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 (TIPX) | Hybrid of fuzzy clustering and TSK fuzzy system [27] | 1.85 | 80.11%      |
|         | Pre-processing Evolutionary Neural Networks (PENN) [22]              | 1.12 | 67.14%      |
|         | Pre-processing Evolutionary Neural Networks back propagation (PEBPNN) [22] | 1.12 | 67.14%      |
|         | Pre-processed Evolutionary LM neural networks (PELMNN) [22]          | 0.76 | 51.58%      |
|         | Proposed model                                                        | 0.368| -           |
| Tehran Index of top 50 Companies (TIT50C) | 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 Industry Index (TII) | Hybrid of fuzzy clustering and TSK fuzzy system [27] | 1.03 | 69.90%      |
|         | 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
Comparing the results of the proposed model with the Asadi’s one

| Stock name                                      | MAPE | - | POICD | - | U of Theil | ARV |
|------------------------------------------------|------|---|-------|---|------------|-----|
| 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 |