ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2016.0165

APPLYING ARTIFICIAL NEURAL NETWORK OPTIMIZED BY FIREWORKS ALGORITHM FOR STOCK PRICE ESTIMATION

Khuat Thanh Tung[1], Nguyen Thi Bich Loan[2], Le Quang Chanh[3] and Le Thi My Hanh[4]
DATIC Laboratory, The University of Danang, University of Science and Technology, Vietnam
E-mail: [1]thanhtung09t2@gmail.com, [2]s.viva13@gmail.com, [3]quangchanh.11tclc.dut@gmail.com, [4]ltmhanh@dut.udn.vn

Abstract
Stock prediction is to determine the future value of a company stock dealt on an exchange. It plays a crucial role to raise the profit gained by firms and investors. Over the past few years, many methods have been developed in which plenty of efforts focus on the machine learning framework achieving the promising results. In this paper, an approach based on Artificial Neural Network (ANN) optimized by Fireworks algorithm and data preprocessing by Haar Wavelet is applied to estimate the stock prices. The system was trained and tested with real data of various companies collected from Yahoo Finance. The obtained results are encouraging.

Keywords:
Fireworks Algorithm, Artificial Neural Network, Stock Price Forecasting, Back-Propagation algorithm, Wavelet Transform

1. INTRODUCTION

Stock market is one of the most active and crucial financial markets. Traders usually use many methods to analyze the huge and volatile data to make decisions. However, the stock markets are influenced by plenty of factors including economic environments, government policies, market capitalization, demand and supply, market news, inflation rate, etc. Therefore, the stock market trends are nonlinear, uncertain, and non-stationary and nowadays it tends to be more risk than before for predicting the stock price.

The purpose of traders is to gain stable and high profits. Thus, a lot of approaches have been carried out to generate the accurate predictions in which artificial neural network is one of data mining techniques being widely adopting for the stock price estimation problems. Several studies have indicated that the ANN outperforms statistical regression models and also enable deeper analysis of huge data sets, especially those that have the tendency to oscillate within a short time period [4-7]. In order to enhance the prediction accuracy, preprocessing techniques and optimization algorithms are usually combined into the ANN. Our study uses Haar Wavelet Transforms to preprocess data sets as well as applying the Fireworks algorithm [8] to optimize the weights and biases of the ANN before training the network by Back-propagation (BP) algorithm. Because the representation of a wavelet can deal with the non-stationary involved in the economic and financial time series, the Haar Wavelet Transform is used to decompose the stock price time series and eliminate noise [9]. The Fireworks algorithm is a novel swarm intelligence algorithm introduced by Ying Tan [10]. Because of a promising performance about optimization accuracy and convergence speed, this algorithm can be used for seeking high-quality optimal solutions and creating the balance between complexity and performance, as well as optimizing predictions efficiently.

The main purpose of this research is to figure out the effectiveness of the ANN improved by adopting the Fireworks algorithm in order to resolve the regression problems on a concrete domain such as the Stock Market.

The rest of the paper is organized as follows. Section 2 presents some related works carried out on the stock price estimation problem. The proposed method is described in detail in section 3. Section 4 is experiments and evaluations while section 5 gives conclusions of the problem.

2. RELATED WORKS

A lot of researches about the financial data mining have been done. Chun and Kim [11] implemented a neural network system using the technical analysis variables for listed companies in Shanghai Stock Market. The efficiency of two learning algorithms and two weight initialization methods has been compared in their study. The results showed that the prediction of stock prices is fairly acceptable with both the algorithm and initialization methods, but the performance of the back-propagation can be grown by conjugate gradient learning and multiple linear regression weight initializations. Nevertheless, when the structure of ANN is complex and there are large training samples, the speed of convergence in these algorithms will become very slow. This impacts the rigor of the predicted results of ANN. To cope with this problem, a new method employing the evolutionary algorithms is proposed in some researches [5, 12-14].

Chan, Leung,and Daouk employed the probabilistic neural network (PNN) to forecast the direction of index after training with historical data. Their empirical results showed that the PNN-based investment strategies obtain higher returns than other investment strategies examined in their study [15]. Parasuraman and Elshorbagy utilized Wavelet networks (WNs) as a promising alternative to traditional neural networks. In traditional ANNs, genetic algorithms are used to optimize the weights and bias values. However, in WNs, the translation and dilation factors of wavelets are also optimized [16]. Nevertheless, these prediction models have the disadvantage of massive noise and high dimensionality of stock price data.

Kim and Han [17] adopted a genetic algorithm to transform continuous input values into discrete ones. The genetic algorithm was used to reduce the complexity of the feature space. Kishikawa and Tokinaga [18] employed a wavelet transform to extract the short-term feature of stock trends.

The previous works have used different forecasting techniques in order to estimate the stock market trends. Several methods attempted to predict the daily returns, while some other studies developed forecasting models to predict the rate of returns of individual stocks. In many studies, it was also found that researchers have attempted to compare their results with other statistical tools. Each approach has advantages and downsides, so it is able to use one of them to reduce the disadvantage of another. These findings come up with a strong motivation for modeling predictable tools for stock market prediction. In addition to applying wavelet-based data preprocessing, this work utilizes the Fireworks algorithm to optimize the weights and biases of ANN to enhance the accuracy for obtained results of the stock price prediction.

3. METHODOLOGY

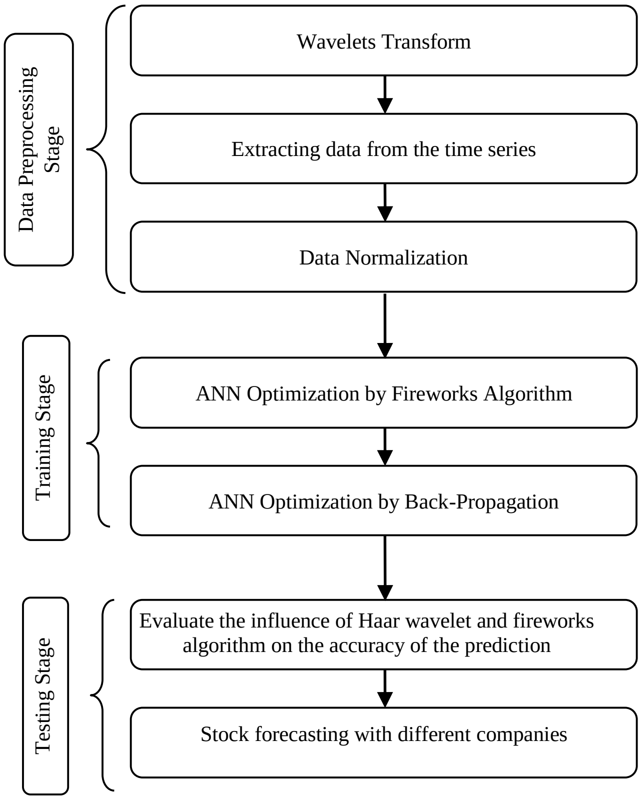


Fig.1. The overview of proposed approach

In general, there are two stock estimation approaches: Technical Analysis and Fundamental Analysis. Technical analysis using time-series analysis to tackle the identification of the stock price relied on the historical data, while Fundamental analysis focuses on the forces of supply, the past efficiency of the firms and the earnings forecast.

To combine both Fundamental and Technical Analyses, this study shows a novel method that integrates the Haar Wavelet Transforms and the Fireworks algorithm into the Multilayer Perceptron (MLP) Neural Network. The Fig.1 briefly illustrates the key process used in this research and it will be explained in more detail below.

3.1 DATA FORMATTING

The factors used for training the ANN are taken based on the experience of trader with regard to the specific stocks. There are lots of technical indicators and fundamental factors such as: Relative Strength Index (RSI), Boilinger Bands, Volume Oscillator, Moving Average (MA), Close/Open prices, etc. being able to be utilized to analyze the stock market. In this study, close price is used to train the ANN, so the output of the ANN is the close price.

3.2 DATA PREPROCESSING

3.2.1 Noise filtering using Haar Wavelet Transform:

The first step of data preprocessing is the use of the Haar Wavelet Transforms to decompose the financial time series and eliminate noise since the representation of a wavelet can deal with the non-stationary involved in the economic and financial time series [9]. Wavelets are mathematical functions that break data into different frequency components, and then each component is studied with a resolution matched to its scale.

A time series can be viewed in multiple resolutions when employing Wavelets. Each resolution presents a various frequency. The wavelet technique computes averages and differences of a signal, breaking the signal down into spectrums. The Haar Wavelet algorithm works on time series whose size is a power of two (e.g., 32, 64, 128...). Each step of the wavelet transform creates two sets of values: a set of averages and a set of differences known as the wavelet coefficients. Each set is a half of the size of the input data. For instance, if the time series has 128 elements, the first step will generate 64 averages and 64 coefficients. The set of averages then becomes the input for the next step (e.g., 64 averages creating a new set of 32 averages and 32 coefficients). This process is iterated until one average and one coefficient are obtained.

The strength of two coefficient spectrums generated by a wavelet calculation reflects the change in the time series at different resolutions. The first coefficient band describes the highest frequency changes. This is the noisiest part of the time series. This noise can be removed by employing threshold techniques. Each later band reflects changes at lower and lower frequencies.

3.2.2 Extracting data from time series:

In the stock price estimation problem, we have to decide that how many prices of the recent days will be used to predict the price of the next day. That value is called as “windowSize”. Investors might utilize any values for windowSize that they want, commonly in the range of 10 to 180 days.

This work uses 30-to-1 model that means using 30 recent days to predict the next day. WindowSize is also the number of inputs used in the input layer of the ANN. In order to train the ANN, many 30-to-1 sequences need to be employed, and each sequence consists of two vectors. The input vector consists of 30 prices of 30 recent days while the output vector contains the price of the next day. In order to get n sequences, we have to slide the window back n steps, and then extract one sequence at each step [7].

3.2.3 Data Normalization:

Normalization is a process converting the time series data points into a small pre-defined range generally from 1 to -1 or 0 [19]. In order to make the training process easy, the data needs to be normalized before training the ANN since the prices are in the various ranges. This study employs Vector Normalization [20] for normalizing data.

Mathematical formula of Vector Normalization is the Eq.(1),

$$N_i = \frac{T_i}{\sqrt{\sum_{j=1}^{k} T_j^2}} \quad (1)$$

where, N_i is the normalized data and T_i is the time-series data, k is the number of values in series and $i = 1,\ldots,k$.

3.3 MLP NEURAL NETWORK SETTING

In general, the architecture of MLP-ANN can consist of many hidden layers and each hidden layer can include many neurons. However theoretical works have shown that a ANN with one hidden layer is good enough to approximate any complicated non-linear functions [19, 21]. In addition, many studies and experimental results also indicated that one hidden layer is sufficient for most of the forecasting problems [4, 19, 22]. Therefore, this work uses the architecture of MLP-ANN with one hidden layer.

Other hard tasks when selecting good parameters for the ANN are the number of hidden neurons, and activation function. Setting an appropriate architecture of the ANN for a particular problem is a crucial task, because the network topology directly influences to its computational complexity and generalization capability. Too much hidden layers or hidden neurons will lead the ANN to the over-fitting. Based on our experiments and other researches [7, 23], the ANN with 8 neurons for the hidden layer and Bipolar Sigmoid function (Fig.2) as activation function for both hidden and output layers is suitable for predicting the stock price.

The Fig.3 shows the structure of the ANN used for our prediction system. The input layer is mapped with the input vector containing 30 (windowSize) latest close prices. The output layer consisting of one neuron denotes the close price of the next day.

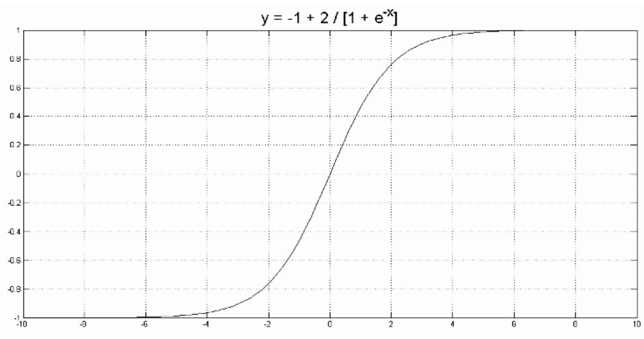


Fig.2. Bipolar Sigmoid function

It can be also known that the ANN structure also depends on the experience of investors and other factors, so our settings for ANN's parameters are just a recommendation for traders.

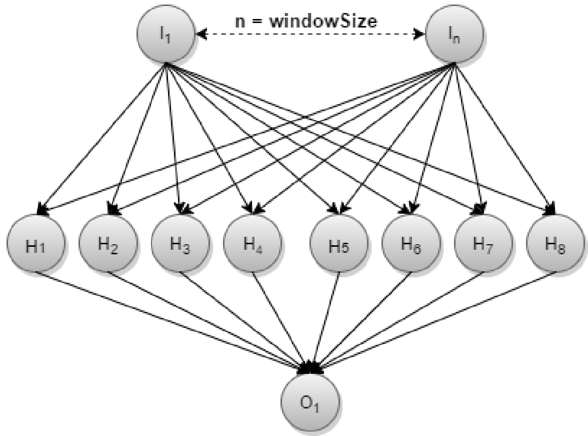


Fig.3. The Architecture of Proposed ANN – windowSize-8-1

3.4 MLP NEURAL NETWORK TRAINING

3.4.1 Initializing optimized weights with the fireworks algorithm:

3.4.1.1 FA Framework:

After the data preprocessing process, the weights and biases for the ANN are initialized by utilizing the fireworks algorithm. This algorithm is an effective method for the convergence to global optimum [8].

Fireworks algorithm (FA) is a novel swarm intelligence algorithm inspired by observing fireworks explosion and is proposed for global optimization of complex functions. In this algorithm, two types of search processes are used as well as creating mechanisms for keeping the diversity of sparks.

When a firework explodes, a shower of sparks will be generated around the firework. The explosion process of a firework can be regarded as a local search around a specific point. To find a point x_j satisfying $f(x_j) = y$, 'fireworks' are continually set off in potential space until one 'spark' targets or is fairly close to the point x_j. Imitating the explosion process of fireworks, a rough framework of the FA is described in Fig.4.

In this paper, a location represents a possible set of optimized weights and biases for ANN. In the FA, for each generation of explosion, n locations where n fireworks are set off are chosen. After explosion, we obtain and assess the locations of sparks. The algorithm stops when the optimal location is found. Otherwise, n other locations are chosen from the current sparks and fireworks for the next generation of explosion.

From Fig.4, it can be seen that the success of the FA depends on a good design of the explosion process and a suitable approach for opting locations.

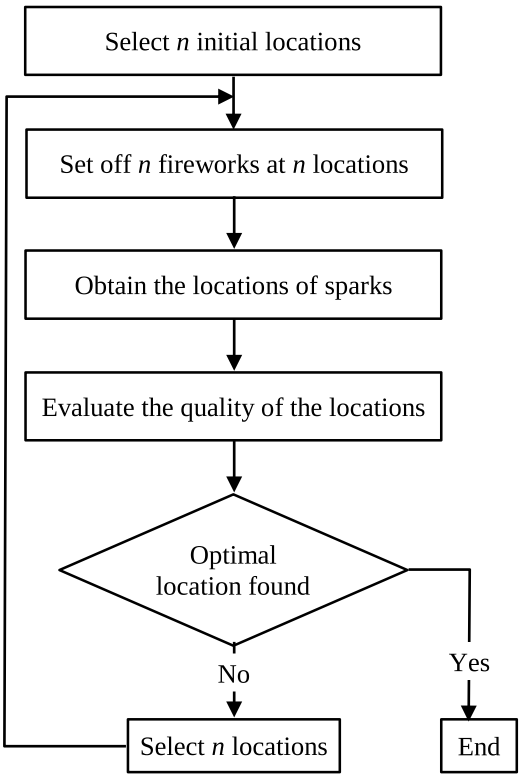
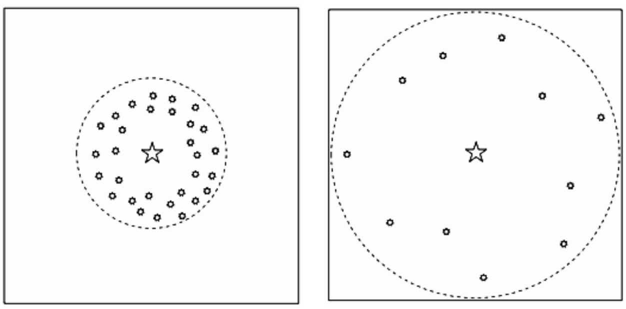


Fig.4. Framework of fireworks algorithm

3.4.1.2 Design of Fireworks Explosion:

Through observing fireworks display, two specific types of fireworks explosion are found. A good firework explosion generates numerous sparks which centralize the explosion center. However, a bad firework explosion generates quite a few sparks which scatter around the explosion center.

The two types of explosion are shown in Fig.5. From the point of view of a search algorithm, a good firework means that the firework may be close to the optimal location. Thus, it is proper to use more sparks to search the local area around the firework. On the contrary, a bad firework means the firework may be far from the optimal location. Then, the search radius should be larger. In the FA, a good firework creates more sparks and the explosion amplitude is smaller, compared to a bad one.



(a) Good explosion (b) Bad explosion

Fig.5. Two types of fireworks explosion

Number of Sparks: Suppose the FA is designed for the general optimization problem:

$$\text{Minimize } f(x) \in R,\ x_{\min} \le x \le x_{\max} \tag{2}$$

where, $x = [x_1, x_2, \ldots, x_D]$ is a location in the potential space, $f(x)$ is an objective function, $x_{\min}$ and $x_{\max}$ denote the bounds of the potential space, D is the dimensionality of the location x. As for the proposed ANN architecture (windowSize-8-1), the value of D can be calculated using the below expression whose details are shown in Table.1.

$$D = IW\{1,1\} + b\{1,1\} + LW\{2,1\} + b\{2,1\}.$$

Table.1. Parameters for proposed ANN (windowsize-8-1)

Value	**Symbol**	**Description**
WindowSize*8	$IW\{1, 1\}$	Weights of the connections from input layer to hidden layer
8	$b\{1, 1\}$	Biases of neurons in hidden layer
8*1	$LW\{2, 1\}$	Weights of the connections between output layer and hidden layer
1	$b\{2, 1\}$	Biases of output neurons

Next the number of sparks generated by each firework x_i is determined as Eq.(3).

$$s_i = t \times \frac{y_{\max} - f(x_i) + \xi}{\sum_{i=1}^{n}\left(y_{\max} - f(x_i)\right) + \xi} \tag{3}$$

where, t is a parameter controlling the total number of sparks generated by the n fireworks, $y_{\max} = \max(f(x_i))$ ($i = 1, 2,\ldots, n$) is the maximum or minimum value of the objective function among n fireworks, and ξ denotes the smallest constant in the computer, is employed to avoid zero-division-error.

In order to avoid overwhelming effects of gorgeous fireworks, bounds are defined for s_i, which is described in Eq.(4).

$$\hat{s}_i = \begin{cases} \text{round}(a \times t) & \text{if } s_i < a \times t \\ \text{round}(b \times t) & \text{if } s_i < b \times t,\ a < b < 1 \\ \text{round}(s_i) & \text{otherwise} \end{cases} \tag{4}$$

where, a and b are const parameters.

Amplitude of Explosion: On the contrary to the design of sparks number, the amplitude of a good firework explosion is smaller than that of a bad one. Amplitude of the explosion for each firework is specified as Eq.(5).

$$A_i = \hat{A}_i \times \frac{f(x_i) - y_{\min} + \xi}{\sum_{i=1}^{n}\left(f(x_i) - y_{\min}\right) + \xi} \tag{5}$$

where, $\hat{A}_i$ denotes the maximum explosion amplitude, and $y_{\min} = \min(f(x_i))$ ($i = 1,2,\ldots,n$) is the minimum or best value of the objective function among the n fireworks.

Generating Sparks: In the explosion process, sparks might undergo the influences of explosion from random z dimensions. In the FA, the number of the affected directions is randomly obtained as follows.

$$z = \text{round}(D \times rand(0,1)) \tag{6}$$

where, D is the dimensionality of the location x, and $rand(0,1)$ is a random number distributed uniform in the range of [0, 1].

The location of a spark of the firework x_i is obtained by using Algorithm 1.

Imitating the explosion process, a spark's location $\tilde{x}_j$ is first produced. Next, if the obtained location is out of the potential space, it is mapped to the potential space by using the algorithm.

Algorithm 1. Obtain the location of a spark

Initialize the location of the spark: $\tilde{x}_j = x_i$;
z = round($D \times rand(0,1)$)
Randomly choose z dimensions of $\tilde{x}_j$;
Compute the displacement: $d = A_i \times rand(-1,1)$;
for each dimension $\tilde{x}_k^j \in \{\text{pre-selected } z \text{ dimensions of } \tilde{x}_j\}$ **do**
  $\tilde{x}_k^j = \tilde{x}_k^j + d$;
  if $\tilde{x}_k^j < x_k^{\min}$ or $\tilde{x}_k^j > x_k^{\max}$ **then**
    map $\tilde{x}_k^j$ to potential space: $\tilde{x}_k^j = x_k^{\min} + \left|\tilde{x}_k^j\right| \% \left(x_k^{\max} - x_k^{\min}\right)$;
  end if
end for

To keep the diversity of sparks, there is another way of generating sparks called Gaussian explosion, which is shown in Algorithm 2. A function *Gaussian*(1,1), which is a Gaussian distribution with mean 1 and standard deviation 1, is used to define the coefficient of the explosion. $\widehat{m}$ sparks of this type are created in each explosion generation.

Algorithm 2. Obtain the location of a specific spark

Initialize the location of the spark: $\hat{x}_j = x_i$;
z = round($D \times rand(0,1)$)
Randomly chose z dimensions of $\hat{x}_j$;
Compute coefficient of Gaussian explosion: g = *Gaussian*(1,1);
for each dimension $\hat{x}_k^j \in \{\text{pre-selected } z \text{ dimensions of } \hat{x}_j\}$ **do**
  $\hat{x}_k^j = \hat{x}_k^j \times g$;
  if $\hat{x}_k^j < x_k^{\min}$ or $\hat{x}_k^j > x_k^{\max}$ **then**
    map $\hat{x}_k^j$ to potential space: $\hat{x}_k^j = x_k^{\min} + \left|\hat{x}_k^j\right| \% \left(x_k^{\max} - x_k^{\min}\right)$;
  end if
end for

3.4.1.3 Selection of Locations:

At the beginning of each explosion generation, n locations will be selected for the fireworks explosion. In the FA, the current best location x^*, upon which the objective function $f(x^*)$ is optimal among current locations, is always kept for the next explosion generation. After that, $n - 1$ locations are chosen based on their distance to other locations in order to keep the diversity of sparks. The general distance between a location x_i and other locations is defined as Eq.(7).

$$R(x_i) = \sum_{j \in C} d\left(x_i, x_j\right) = \sum_{j \in C} \left\| x_i - x_j \right\| \tag{7}$$

where, C is the set of all current locations of both fireworks and sparks.

Then the selection probability of a location x_i is defined as Eq.(8).

$$p(x_i) = \frac{R(x_i)}{\sum_{j \in K} R(x_j)}. \tag{8}$$

When assessing the distance, any distance measure can be used including Manhattan distance, Euclidean distance, Angle-based distance, etc. In this paper, Euclidean distance is employed.

3.4.1.4 Summary:

Algorithm 3 presents the framework of the FA. During each explosion generation, two kinds of sparks are created respectively as shown in Algorithm 1 and Algorithm 2. In the first kind, explosion amplitude and the number of sparks rely on the quality of the corresponding firework. On the contrary, the second type is produced using a Gaussian explosion process, which carries out seek in a local Gaussian space around a firework.

Algorithm 3: Framework of the FA

Randomly choose n locations for fireworks;
while stopping criteria is not met **do**
  Set off n fireworks respectively at then locations:
  for each firework x_i **do**
    Compute the number of sparks that the firework produces: $\hat{s}_i$, using Eq.(4);
    Obtain locations of $\hat{s}_i$ sparks of the firework x_i using Algorithm 1;
  end for
  for $k = 1 \rightarrow \widehat{m}$ **do**
    Randomly choose a firework x_j;
    Create a specific spark for the firework using Algorithm 2;
  end for
  Choose the best location and keep it for next explosion generation;
  Randomly select $n - 1$ locations from the two types of sparks and the current fireworks according to the probability given in Eq.(8);
end while

3.4.2 Back-Propagation Training:

After optimizing the ANN by using the Fireworks algorithm, the training process is continued with Back-propagation algorithm to adjust the weights in the steepest descent direction (the most negative of the gradients). The ANN will be initialized with the optimized weights and biases in the first training phase and then Back-propagation algorithm will be used to train the ANN for more 50 cycles.

4. EXPERIMENTS AND RESULTS

4.1 EVALUATION CRITERIA

The proposed approaches were evaluated according to the root mean squared error (RMSE), the mean absolute error (MAE) and the mean absolute percentage error (MAPE) criteria. These criteria are defined as follows:

$$RMSE = \sqrt{\frac{1}{N} \times \sum_{j=1}^{N} \left(O_j - Y_j\right)^2} \tag{9}$$

$$MAE = \frac{1}{N} \times \sum_{j=1}^{N} \left| O_j - Y_j \right| \quad (10)$$

$$MAPE = \frac{1}{N} \times \sum_{j=1}^{N} \left| \frac{O_j - Y_j}{Y_j} \right| \quad (11)$$

where, N denotes the size of testing sets.

These criteria measure how the predicted value O is close to the real value Y. The lower these measures are, the better result is. In this study, three these criteria will be employed to evaluate the effectiveness of the following experiments.

4.2 TEST SUITES

The experiment system is implemented in C# .NET and assessed on several historical prices data of different companies including the Apple Inc. (AAPL) in period 2009-2013, Yahoo! Inc. (YHOO) in period 2013-2014 and Google Inc. (GOOG) in period 3/2014-7/2015. These data were taken from Yahoo Finance [24].

4.3 THE INFLUENCE OF WAVELET TRANSFORM ON THE ACCURACY OF THE STOCK PRICE PREDICTION

As mentioned above, the Haar Wavelet Transform can eliminate noise. Therefore, it is suitable for handling highly irregular data series. After applying noise filters, the data of time series become smoother than the original data containing a lot of jags and can be used to train the network. The proof of the benefits of using Wavelets is showed in Table.2 and Fig.6. As can be seen in the Table.2, all the accuracy measuring criteria (RMSE, MAE and MAPE) of training with noise filtered data significantly decrease. The reduction of values of the criteria is 9.567% on average with three testing data sets. The Fig.6 shows that convergence rate of using noise filtered data is better than using original data. This means that the accuracy of prediction is improved. The use of the Wavelet Transform gave the lower error value and the better convergence rate of neural network weights.

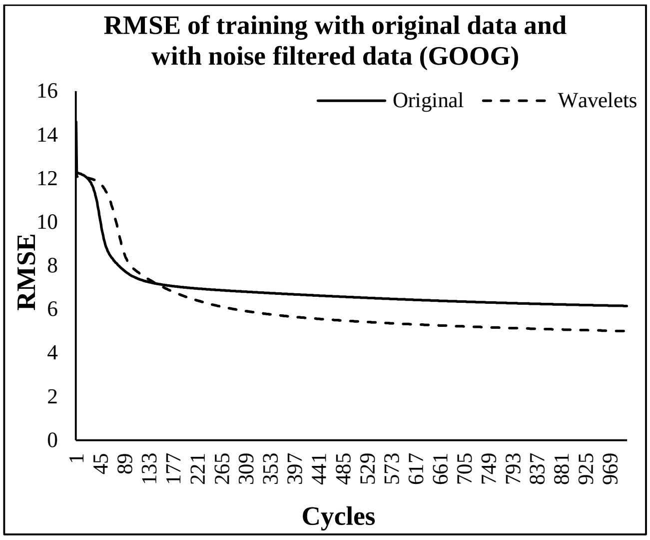


Fig.6. RMSE of training with original data and with noise filtered data (GOOG)

Table.2. RMSE, MAE and MAPE of training with original data and with noise filtered data

Data	RMSE (dollar)		MAE (dollar)		MAPE	
	Original	***Wavelet***	***Original***	***Wavelet***	***Original***	***Wavelet***
AAPL	12.66638878	11.90225584	9.602428909	9.036060803	0.0208109223	0.0196434697
YHOO	4.979928103	4.296104780	4.799414478	4.129660512	0.0957581056	0.0823762636
GOOG	6.028932264	5.250440238	4.795260808	3.731558769	0.0089421414	0.0069500225

Table.3. RMSE, MAE and MAPE of training with FA-ANN-W and ANN-W without using FA

Data	RMSE (dollar)		MAE (dollar)		MAPE	
	ANN-WT	***FA-ANN-WT***	***ANN-WT***	***FA-ANN-WT***	***ANN-WT***	***FA-ANN-WT***
AAPL	12.57714719	10.87330710	9.594449911	8.114399357	0.0209398961	0.0175579859
YHOO	4.717466493	4.089428786	4.544110026	3.919477189	0.0906652783	0.0781525934
GOOG	5.369479714	4.781775922	3.987986820	3.002353158	0.0074318867	0.0055750354

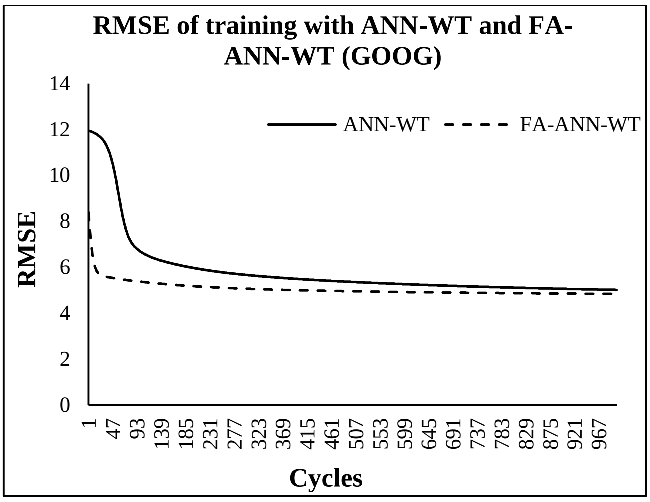


Fig.7. RMSE of training with FA-ANN-WT and ANN-WT (GOOG)

4.4 THE INFLUENCE OF THE FIREWORKS ALGORITHM ON THE ACCURACY OF THE STOCK PRICE PREDICTION

In this work, after data preprocessing with Wavelet Transform (WT), the Fireworks Algorithm (FA) is employed to optimize the network weights and biases by minimizing an objective function such as the root mean square error (RMSE) given by Eq.(9). The optimal set of weights and biases found by Fireworks was used in the next training process of the ANN using Back-propagation algorithm. The use of FA for the optimization of weights and biases gave the lower error value and the faster convergence of neural network weights. The Table.3 and Fig.7 indicated the influence of the FA-ANN-WT model and the ANN-WT (without using the FA) model. The FA-ANN-WT gives the lower values through different testing sets. The reduction of values using the criteria given by Eq(9) to Eq.(11) is 36.66% on average for three testing data sets. The Fig.7 also shows that the RMSE of the FA-ANN-WT forecasting model converges faster than that of the ANN-WT model. Those experimental results indicate that the FA optimization gives the prediction result more accuracy, and it also speeds up the second training stage by reducing the number of training cycle

4.5 THE PREDICTION RESULTS OF THE PROPOSED APPROACHES

This experiment uses the ANN combined with FA algorithm and Wavelet Transform to predict the stock prices for three companies including Google, Yahoo, and Apple. For the Google Inc. in the period of 2014-2015, the sub-datasets for the first twelve-month period are used for the training-validating process, while those from 3/2015 to 6/2015 are chosen for testing. Fig 8 shows the forecasting and real results of the Google Inc. in the period from 3/2015 to 6/2015. As can be seen that the prediction and actual results displayed the same trends in the time period shown, and the deviations among these figures are relatively small. As for Yahoo Inc., the dataset from 2013 to 2014 was used for the experiment. We used the data of seven months before to provide the predicted results for the next two months. For example, the dataset from 2/2013 to 8/2013 was adopted to estimate the stock prices for 9/2013 and 10/2013. The experimental results of the Yahoo Inc. are presented in Fig.9. It can be found that in general the trends of actual and predicted prices are the same, only having some fluctuations in the period from September 28, 2013 to October 8, 2013. However, the predicted results are quite close to the actual stock prices in the time period shown. With regard to the Apple Inc., the dataset from 2008 to 2013 used for the experiment, and we adopted the data of ten months before to estimate the stock prices for the next two months. For instance, the actual stock price from 4/2008 to 1/2009 was employed to predict the results for 2/2009 and 3/2009. The actual and predicted stock prices of Apple Inc. are represented in Fig.10. Looking at the graphs in Fig.10, it can be obvious that the trends between the actual stock prices and the predicted results are quite similar apart from the several small oscillations in the time periods between March 7, 2009 and March 11, 2009 as well as from October 18, 2012 to October 24, 2012 and between November 2, 2012 and November 9, 2012. These errors can occur due to the volatility of many factors in the stock market such as economic crisis, business policy changes. These factors are inconstant, irregular, and the current proposed method has not yet been handled. In general, nevertheless, the deviations between predicted and real stock prices are relatively small, and these figures still ensure the accuracy of estimated results and contribute to assistance for traders. In reality, to perform successful trade we must only know the price direction and experimental results indicated that our study has done this requirement with the predicted trends being relatively close to the real trends as well as the predicted values being quite close to the real values for all firms. These results show that the hybrid method of data preprocessing and optimized algorithm with multi-layer feed-forward neural network trained by back-propagation algorithm can provide the positive predicted value for stock price problem.

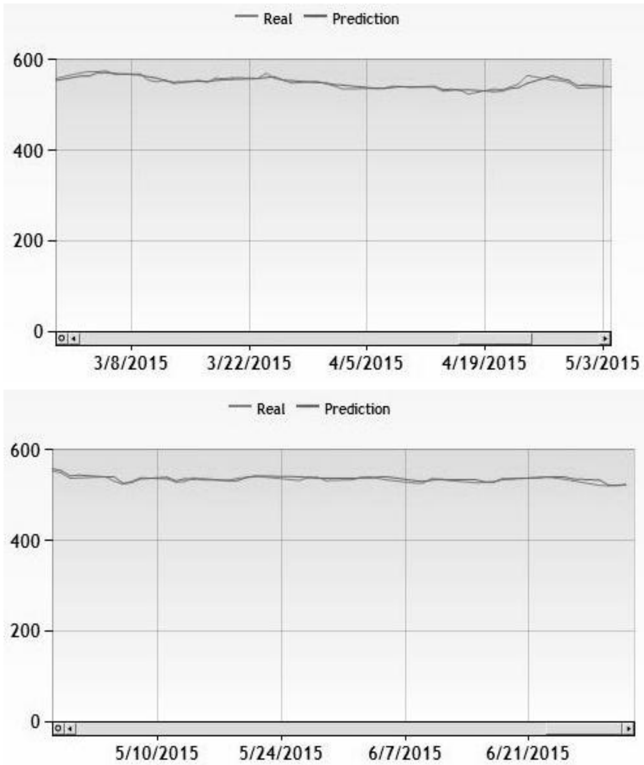


Fig.8. Testing results of FA-ANN-WT for Google Inc. for period 3/2015-7/2015

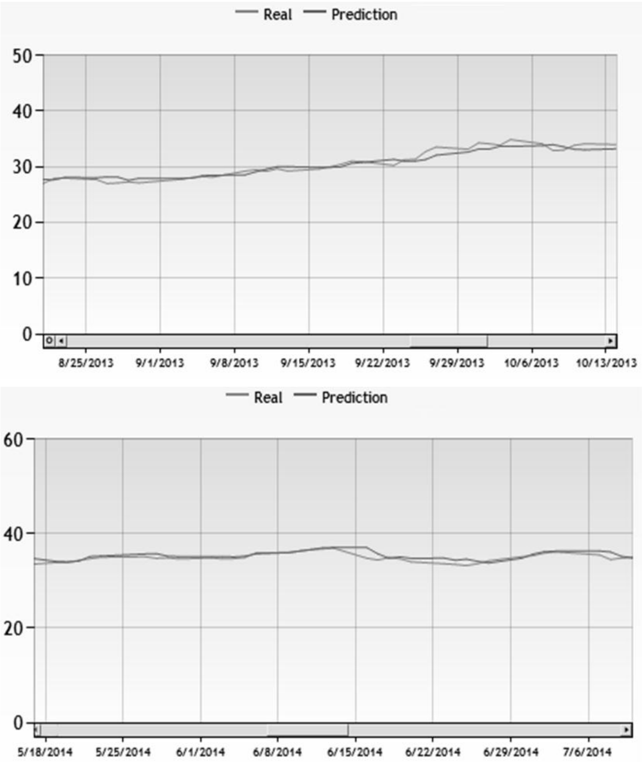
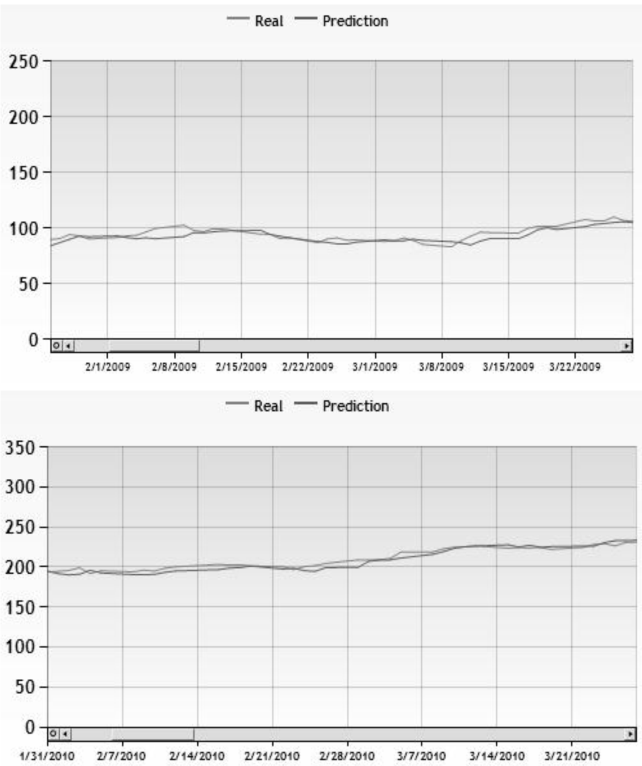


Fig.9. Testing results of FA-ANN-WT for Yahoo Inc. for period 2013-2014



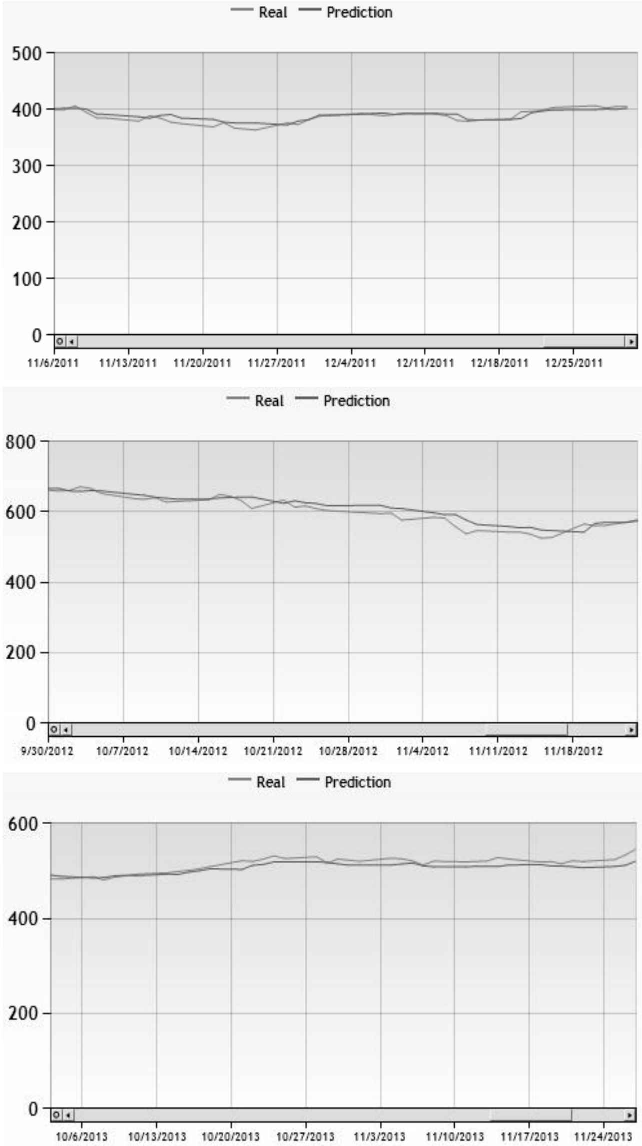


Fig.10. Testing results of FA-ANN-WT for Apple Inc. for period 2009-2013

5. CONCLUSION AND FUTURE WORK

This study suggested a hybrid method of the data preprocessing techniques and optimized algorithms with the multilayer feed-forward neural network trained by back-propagation algorithm to produce a predictive model for raising the rigor of the stock estimation. It can be seen that the use of the Haar Wavelet Transforms helps to reduce the noise in the data sets and enhance the accuracy of obtained results. The FA increases the speed of convergence of the ANN as well as diminishing the relative errors of the training process. We can find that the predictive model with the combination of the Haar Wavelet Transforms and FA into the ANN produces the acceptable results of the stock prediction and significantly supports for traders in making decisions.

Future work focuses on identifying the important influence of particular fundamental analysis variables on the quality of the stock price prediction. In addition, a more advanced pattern selection techniques might be integrated into the system to retrieve significant patterns from the data.

REFERENCES

[1] Gerasimos Marketos, Konstantinos Pediaditakis, Yannis Theodoridis and Babis Theodoulidis, "Intelligent Stock Market Assistant using Temporal Data Mining", MSc Thesis, University of Manchester Institute of Science and Technology, 2004.

[2] Tae Hyup Roh, "Forecasting the Volatility of Stock Price Index", *Expert Systems with Applications*, Vol. 33, No. 4, pp. 916-922, 2007.

[3] Qiang Yang and Xindong Wu, "10 Challenging Problems in Data Mining Research", *International Journal of Information Technology and Decision Making*, Vol. 5, No. 4, pp. 597-604, 2006.

[4] A. Adebiyi Ayodele, K. Ayo Charles, O. Adebiyi Marion and O. Otokiti Sunday , "Stock Price Prediction using Neural Network with Hybridized Market Indicators", *Journal of Emerging Trends in Computing and Information Sciences*, Vol. 3, No. 1, pp. 1-9, 2012.

[5] I.M El-Henawy, A.H Kamal, H.A Abdelbary and A.R Abas, "Predicting stock index using Neural Network Combined with Evolutionary Computation Methods", *Proceedings of 7th International Conference on Informatics and Systems*, pp. 1-6, 2010.

[6] Tsung Jung Hsieh, Hsiao Fen Hsiao and Wei Chang Yeh, "Forecasting Stock Markets using Wavelet Transforms and Recurrent Neural Networks: An Integrated System based on Artificial Bee Colony Algorithm", *Applied Soft computing*, Vol. 11, No. 2, pp. 2510-2525, 2011.

[7] Esmaeil Nourani, Amir Masoud Rahmani and Ahmad Habibizad Navin, "Forecasting Stock Prices using a Hybrid Artificial Bee Colony based Neural Network", *Proceedings of International Conference on Innovation Management and Technology* Research, pp. 486-490, 2012.

[8] Ying Tan and Yuanchun Zhu, "Fireworks Algorithm for Optimization", *Proceedings of 1st International Conference on Swarm Intelligence*, pp. 355-364, 2010.

[9] James B. Ramsey, "The Contribution of Wavelets to the Analysis of Economic and Financial Data", *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 357, pp. 2593-2606, 1999.

[10] Ying Tan, "*Fireworks Algorithm A Novel Swarm Intelligence Optimization Method*", Springer, 2015.

[11] Steve H. Kim and Se Hak Chun, "Graded Forecasting using an Array of Bipolar Predictions: Application of Probabilistic Neural Networks to a Stock Market Index", *International Journal of Forecasting*, Vol. 14, No. 3, pp. 323-337, 1998.

[12] Hua Ning Hao, "Notice of Retraction Short-Term Forecasting of Stock Price based on Genetic-Neural Network", *Proceedings of 6th International Conference on Natural Computation*, Vol. 4, pp. 1838-1841, 2010.

[13] Haoming Huang, Michel Pasquier and Chai Quek, "Financial Market Trading System with a Hierarchical Coevolutionary Fuzzy Predictive Model", *IEEE Transactions on Evolutionary Computation*, Vol. 13, No. 1, pp. 56-70, 2009.

[14] Mustafa E Abdual Salam, Hatem M Abdul Kader and Waiel F Abdel Wahed, "Comparative Study between Differential Evolution and Particle Swarm Optimization algorithms in Training of Feed-Forward Neural Network for Stock Price Prediction", *Proceedings of 7th International Conference on Informatics and Systems*, pp. 1-8, 2010.

[15] An-Sing Chen, Mark T Leung and Hazem Daouk, "Application of Neural Networks to an Emerging Financial Market: Forecasting and Trading the Taiwan Stock Index", *Computers and Operations Research*, Vol. 30, No. 6, pp. 901-923, 2003.

[16] K. Parasuraman and A. Elshorbagy, "Wavelet Networks: an Alternative to Classical Neural Networks", *Proceedings of IEEE International Joint Conference on Neural* Networks, Vol. 5, pp. 2674-2679, 2005.

[17] Kyoung. Jae Kim and Ingoo Han, "Genetic Algorithms Approach to Feature Discretization in Artificial Neural Networks for the Prediction of Stock Price Index", *Expert systems with Applications*, Vol. 19, pp. 125-132, 2000.

[18] Jianjun Lu, Yoshinori Kishikawa and Shozo. Tokinaga, "Realization of Feature Descriptive Systems for Clusters by using Rule Generations based on the Genetic Programming and its Applications", *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. 89, No. 12, pp. 1142-1152, 2006.

[19] G. Cybenko, "Approximation by Superpositions of a Sigmoidal Function", *Mathematics of Control, Signals and Systems*, Vol. 2, No. 4, pp. 303-314, 1989.

[20] Shunrong Shen, Haomiao Jiang and Tongda Zhang, "Stock Market Forecasting using Machine Learning Algorithms", Stanford University, 2012.

[21] Kur Hornik, Maxwell Stinchcombe and Halbert White, "Multilayer Feedforward Networks are Universal Approximators", *Neural Networks*, Vol. 2, No. 5, pp. 359-366, 1989.

[22] Hiroshi Takaho, Takayuki Arai, Tsuyoshi Otake and Mamoru Tanaka, "Prediction of the Next Stock Price using Neural Network for Data Mining", *Proceedings of International Symposium on Non-Linear Theory and its Applications*, pp. 411-414, 2002.

[23] Amir Omidi, Esmaeil. Nourani and Mehdi Jalili, "Forecasting Stock Prices using Financial Data Mining and Neural Network", *Proceedings of 3rd International Conference on Computer Research and* Development, pp. 242-246, 2011.

[24] Yahoo Finance, Available at: http://finance.yahoo.com/.