Expert Systems with Applications 38 (2011) 14346–14355

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Short communication

Forecasting stock indices with back propagation neural network

Jian-Zhou Wang [a], Ju-Jie Wang [a], Zhe-George Zhang [b,c,*], Shu-Po Guo [a]

[a] School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, China
[b] Department of Decision Sciences, Western Washington University, Bellingham, WA 98225, USA
[c] Faculty of Business Administration, Simon Fraser University, Burnaby, British Columbia, Canada V5A 1S6

ARTICLE INFO

Keywords:
Wavelet de-noising
BP neural network
WDBP neural network
Stock prices

ABSTRACT

Stock prices as time series are non-stationary and highly-noisy due to the fact that stock markets are affected by a variety of factors. Predicting stock price or index with the noisy data directly is usually subject to large errors. In this paper, we propose a new approach to forecasting the stock prices via the Wavelet De-noising-based Back Propagation (WDBP) neural network. An effective algorithm for predicting the stock prices is developed. The monthly closing price data with the Shanghai Composite Index from January 1993 to December 2009 are used to illustrate the application of the WDBP neural network based algorithm in predicting the stock index. To show the advantage of this new approach for stock index forecast, the WDBP neural network is compared with the single Back Propagation (BP) neural network using the real data set.

1. Introduction

Stock price data are always one of the most important information to investors. Unfortunately, stock prices are essentially dynamic, non-linear, nonparametric, and chaotic in nature. This implies that the investors must handle the time series which are non-stationary, noisy, and have frequent structural breaks (Oh & Kim, 2002; Wang, 2003). In fact, stock prices' movements are affected by many macro-economical factors such as political events, company's policies, general economic conditions, commodity price index, bank rates, investors' expectations, institutional investors' choices, and psychological factors of investors. Thus forecasting stock price movement accurately is not only extremely challenging but also of great interest to investors.

Past works on forecasting stock prices can be classified into two categories: statistical and artificial intelligence (AI) models. The statistical approach includes autoregressive integrated moving average (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH) volatility (see Franses & Ghijsels, 1999), and the smooth transition autoregressive model (STAR) (see Sarantis, 2001). These models are mainly based on the assumption of linearity among normally distributed variables. However, the linearity and normality assumption may not be satisfied in modeling the stock price movements. On the other hand, the AI models without this restrictive assumption have been shown to outperform the statistical models empirically by many recent studies such as Enke and Thawornwong (2005), Hansen and Nelson (2002), Ture and Kurt (2006) and Zhang (2003). Therefore, AI approaches, such as neural network, fuzzy system, and genetic algorithm, have been utilized in predicting stock prices in Armano, Marchesi, and Murru (2005), Chen, Leung, and Daouk (2003), Chun and Kim (2004), Kim and Han (2000), Shen and Loh (2004), Thawornwong and Enke (2004), Vellido, Lisboa, and Meehan (1999) and Wang (2002a,b).

In this paper, we propose a hybrid forecasting model called Wavelet De-noising-based Back Propagation (WDBP) neural network. In such a model, the original data are first decomposed into multiple layers by the wavelet transform. Each layer has a low-frequency and a high-frequency signal component. Then a Back Propagation (BP) neural network model is established by the low-frequency signal of each layer for predicting the future value. To the best of our knowledge, this paper is the first attempt to utilizing the WDBP neural network based algorithm for forecasting the stock prices. The empirical data set of Shanghai Composite Index (SCI) closings prices from January 1993 to December 2009 are used to illustrate the application of the WDBP neural network. Furthermore, the superiority of our model is shown by comparing the WDBP neural network with a single BP neural network.

The rest of this paper is organized as follows. Section 2 focuses on the WDBP neural network algorithm for forecasting stock price. Section 3 presents the experimentation design and numerical results. Section 4 concludes.

* Corresponding author at: Department of Decision Sciences, Western Washington University, Bellingham, WA 98225, USA. Tel.: +1 360 650 2867; fax: +1 360 650 4844.

E-mail addresses: gzhang@sfu.ca, George.zhang@wwu.edu (Z.-G. Zhang).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.04.222

2. The proposed WDBP neural network

2.1. Artificial neural networks

In general, artificial neural networks (ANNs) possess attributes of learning, generalizing, parallel processing and error endurance. These attributes make the ANNs powerful in solving complex problems (see Azadeh, Ghaderi, & Sohrabkhani, 2008). Back-Propagation (BP) neural networks, a type of ANNs, take inputs only from the previous layer and send outputs only to the next layer (see Fig. 1).

Our study employs a BP neural network which is widely used in business situations. A three-layer BP neural network is shown in Fig. 2. The BP process determines the weights for the connections among the nodes based on data training, yielding a minimized least-mean-square error between the actual and the estimated values from the output of the neural network. The connection weights are assigned initial values first. The error between the predicted and actual output values is back-propagated via the network for updating the weights (see Wang, 2009). Theoretically, neural networks can simulate any kind of data pattern given sufficient training. The neural network must be trained before being applied for forecasting. Assume that there are n input neurons, m hidden neurons, and one output neuron, a training process can be described by the two stages (see Zhang & Wu, 2009):

(I) *Hidden layer stage:* The outputs of all neurons in the hidden layer are calculated by the following steps:

$$net_j = \sum_{i=0}^{n} v_{ij}x_i, \quad j = 1, 2, \ldots, m, \tag{1}$$

$$y_j = f_H(net_j), \quad j = 1, 2, \ldots, m. \tag{2}$$

Here net_j is the activation value of the jth node, y_j is the output of the hidden layer, and f_H is called the activation function of a node, usually a sigmoid function as follows:

$$f_H(x) = \frac{1}{1 + \exp(-x)}. \tag{3}$$

(II) *Output stage:* The outputs of all neurons in the output layer are given as follows:

$$O = f_o\left(\sum_{j=0}^{m} \omega_{jk}y_j\right), \tag{4}$$

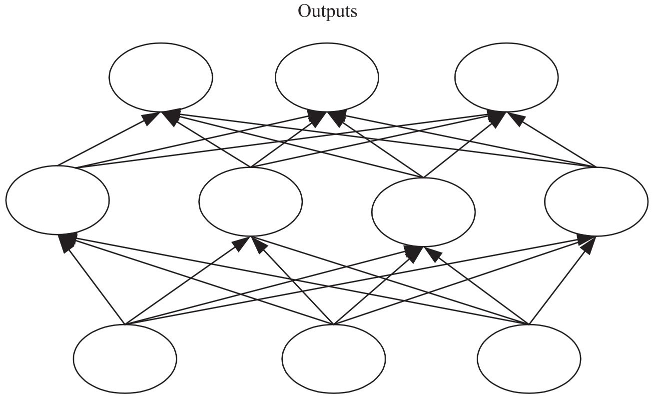

Fig. 1. Back-propagation neural network.

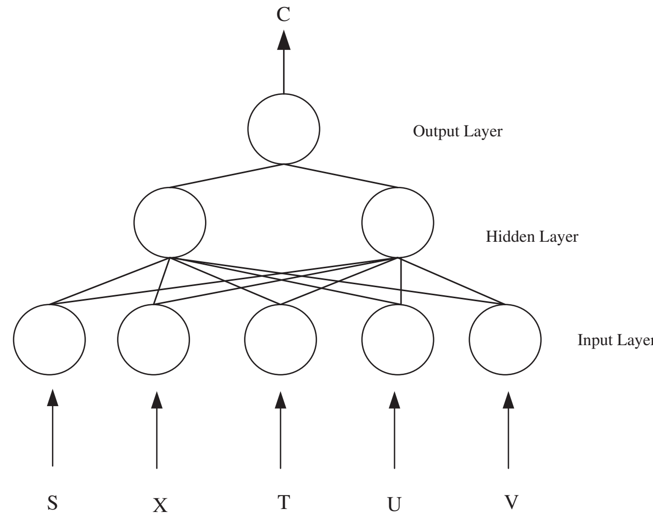

Fig. 2. The architecture one hidden layer back propagation neural network.

where f_o is the activation function, usually a linear function. All weights are assigned with random values initially, and then modified by the delta rule according to the learning samples.

2.2. Wavelet transform

Wavelet transform is used in analyzing non-stationary time series for generating information in both the time and frequency domains. It may be regarded as a special type of Fourier transform at multiple scales and decomposes a signal into shifted and scaled versions of a "mother" wavelet. The continuous wavelet transform, denoted by *CWT*, is defined as the convolution of a time series $x(t)$ with a wavelet function $w(t)$ (Goswami & Chan, 1999):

$$CWT_x^{\psi}(b,a) = \phi_x^{\psi}(b,a) = \frac{1}{\sqrt{|a|}} \int x(t) \cdot \psi^{\bullet}\left(\frac{t-b}{a}\right) dt, \tag{5}$$

where a is a scale parameter, b is the translational parameter and $\bullet$ is the complex conjugate of $\psi(t)$. Let $a = 1/2^s$ and $b = k/2^s$, where s and k belong to the integer set Z. The CWT of $x(t)$ is a number at $(k/2^s, 1/2^s)$ on the time-scale plane. It represents the correlation between $x(t)$ and $\psi^{\bullet}(t)$ at that time-scale point. A discrete version of Eq. (5) is thus obtained as

$$DWT_x^{\psi}(k,s) = \phi_x^{\psi}\left(\frac{k}{2^s}, \frac{1}{2^s}\right) = \int_{\infty}^{\infty} x(t) \cdot \psi^{\bullet}\left(\frac{t-k/2^s}{1/2^s}\right) dt, \tag{6}$$

which separates the signal into components at various scales corresponding to successive frequencies.Note that DWT corresponds to the multi-resolution approximation expressions for analysis of a signal in many frequency bands (or at many scales). In practice, multi-resolution analysis is carried out by starting with two channel filter banks composed of a low-pass and a high-pass filter, and then each filter bank is sampled at a half rate of the previous frequency. The number of steps of this de-composition procedure will depend on the length of data. The down sampling procedure keeps the scaling parameter constant (1/2) throughout successive wavelet transforms (Li & Kuo, 2008).

Over the past decade, DWT has been well developed and applied in signal analysis of various fields (Mörchen, 2003). In this study, DWT is utilized to remove noise from the stock prices data for prediction purpose.

2.3. The WDBP neural network model algorithm

The WDBP neural network algorithm first decomposes the original data into several layers via the wavelet transform, then establishes a BP neural network model using the low-frequency signal of every layer for prediction. The algorithm is described as follows and the flowchart is shown in Fig. 3.

1. *Wavelet decomposition:* This step decomposes the signal into low-pass filter B and high-pass filter D by the wavelet transform. The low-pass filter B reflects the main features of the signal, and the high-pass filter D represents random factors often called the noise. The main purpose of the wavelet de-composition is to separate the basic characteristics from the noise of the signal.

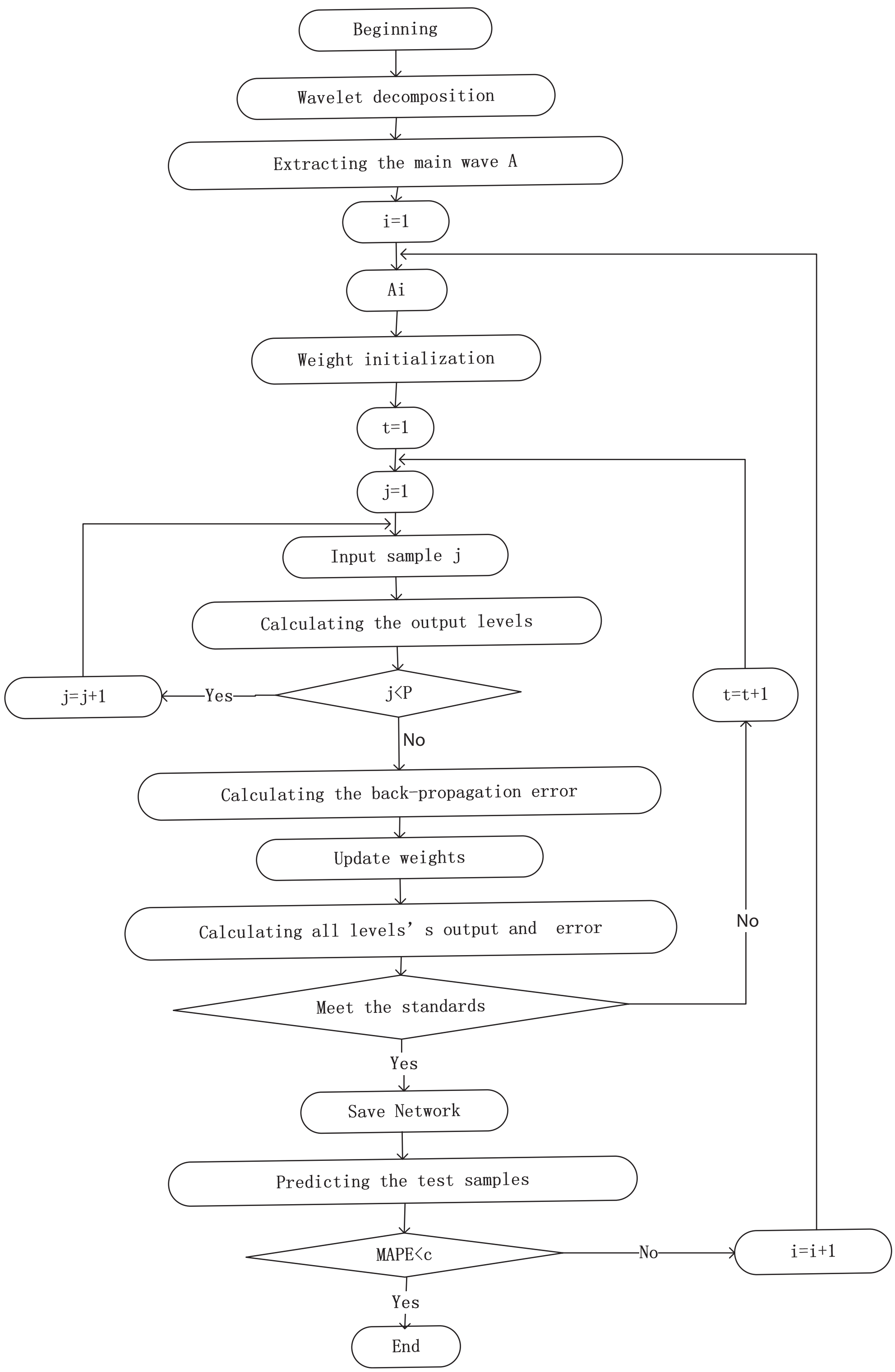

Fig. 3. The algorithm of the proposed WDBP neural network model.

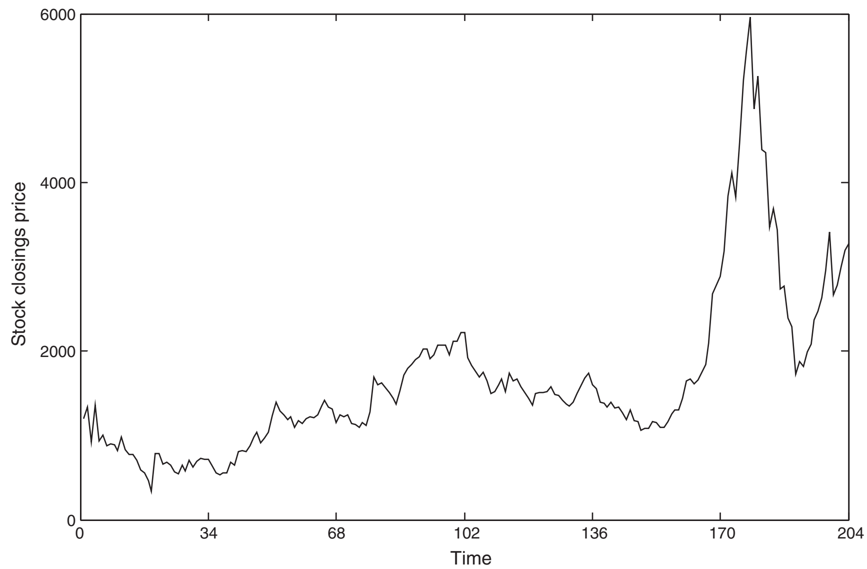

Fig. 4. The original data.

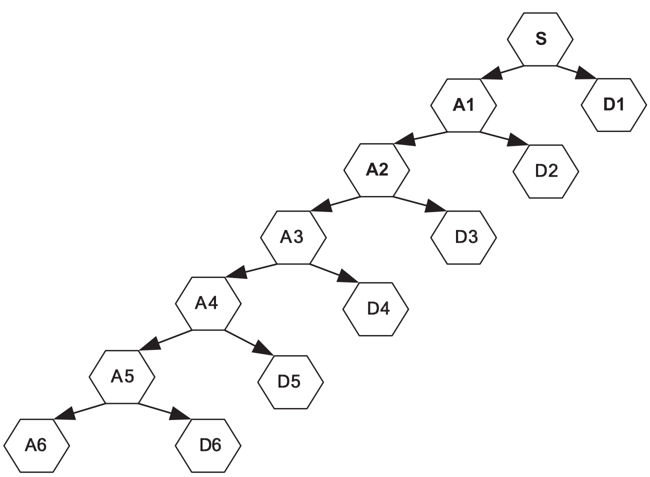

Fig. 5. The six levels of db3 wavelet de-composition.

2. *Main wave extraction:* This step extracts the low-pass filter B_i ($i = 1, 2, \ldots, n$) and remove the high-pass filter D_i ($i = 1, 2, \ldots, n$) in the data. Here n is the number of layers of the wavelet de-composition.
3. *Normalization*: Low frequency data B_i is normalized into A_i.
4. *Training and testing data sets determination:* Divide the low-pass filter data A_i into two subsets: training $A_i(1)$ and testing $A_i(2)$ data. A validation must be performed by using $A_i(2)$ to test how well the network is able to generalize for unknown data. To cover wide ranges of outcomes, it is necessary to achieve a balance between the training and validation data set sizes.
5. *Relation estimation:* Estimate the relation between input(s) and output(s) through training the BP network using the training set $A_i(1)$. To find the appropriate number of hidden nodes, repeat these steps using different training parameters for networks with 1 to q nodes in their hidden layer. Training continues until the estimation error is below a threshold.
6. *Validation*: Validate the network using $A_i(2)$ and make forecasts.
7. *De-normalization*: De-normalize the predicted values.
8. *Evaluation*: Check if $P^{(i)}_{MAPE}$, the mean absolute percentage error (MAPE) is no more than a threshold. If $P^{(i)}_{MAPE} \leqslant c$, the training is stopped and the algorithm ends; otherwise, set $i = i + 1$ and go to the Step 4.

2.4. Evaluation criteria

Furthermore, three accuracy measures are utilized to evaluate the forecasting performance relative to the actual price C_n^{MP}. These are mean absolute error (MAE), root mean-square error (RMSE) and mean absolute percentage error (MAPE) which are given as follows:

$$MAE = T^{-1}\sum_{n=1}^{T}\left|C_n^{MP} - C_n\right|, \tag{7}$$

$$RMSE = \left(T^{-1}\sum_{n=1}^{T}(C_n^{MP} - C_n)^2\right)^{1/2}, \tag{8}$$

$$MAPE = T^{-1}\sum_{n=1}^{T}\left|(C_n^{MP} - C_n)/C_n^{MP}\right|, \tag{9}$$

where C_n^{MP} and C_n are the actual value and predicted value, respectively, and T is the sample size.

Smaller values of these measures indicate more accurate forecasted results and if the results are not consistent among three criteria, we choose the relatively more stable MAPE, as suggested by Makridakis (1993), to be the benchmark. In this paper, we use all three measures to evaluate the forecasting performance.

3. Experimentation design

3.1. Data preparation

The data for our experiments are SCI closing prices, collected on the Shanghai Stock Exchange (SSE). The total number of values for the SCI closing prices is 204 trading months, from January 1993 to December 2009. Fig. 4 shows the original data series. The data set is partitioned into a training set (80%) and a testing set (20%) for validation.

As the network's output value is between 0 and 1 (a characteristics of the transfer function), we need to normalize the original data series. A stock closings price p is normalized to p' by

$$p'_i = \frac{p_i - p_{\min}}{p_{\max} - p_{\min}}, \quad i = 1, 2, 3, \ldots, 204, \tag{10}$$

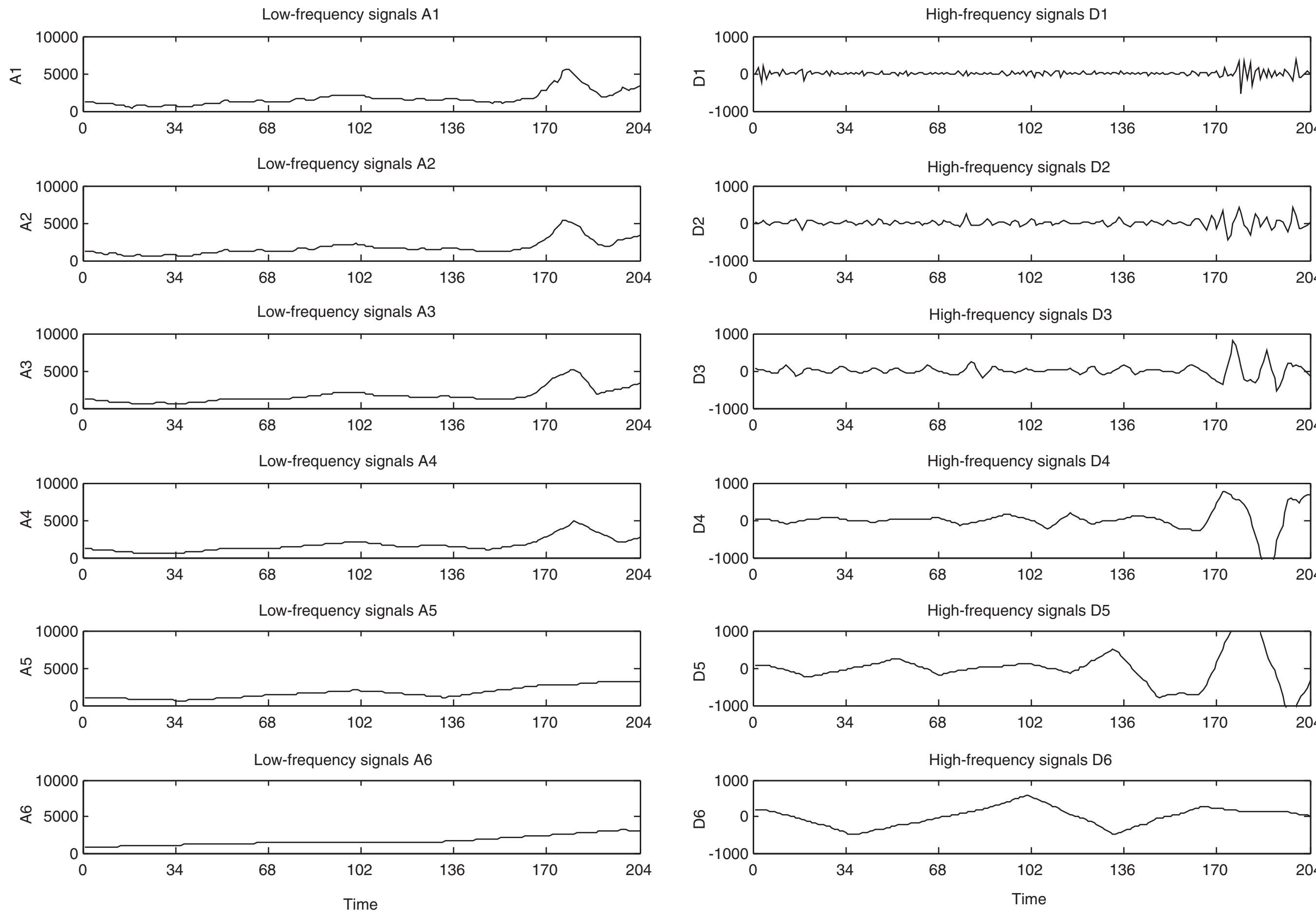

Fig. 6. The de-composition process of the experimental data.

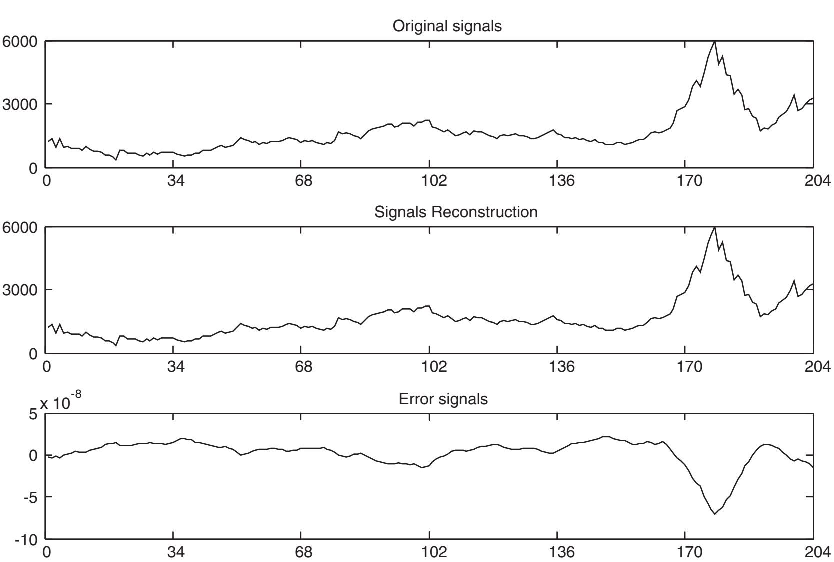

Fig. 7. The re-construction process of the experimental data.

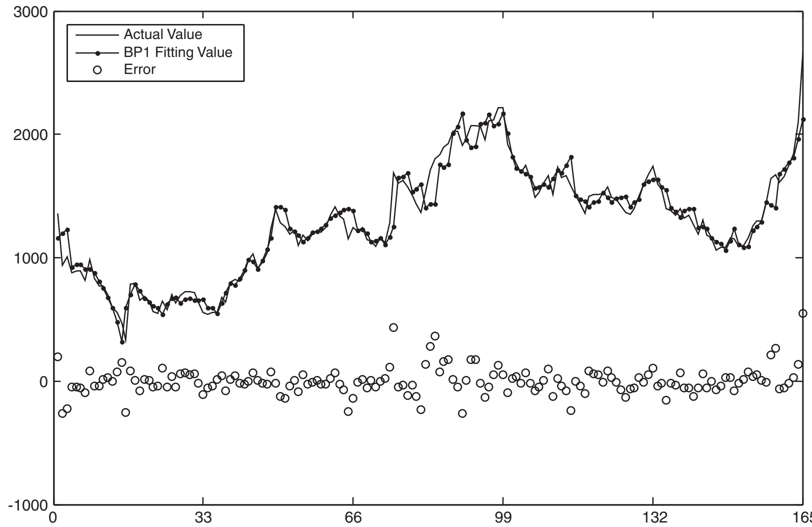

Fig. 8. The comparison chart of the actual value and BP1 fitting value.

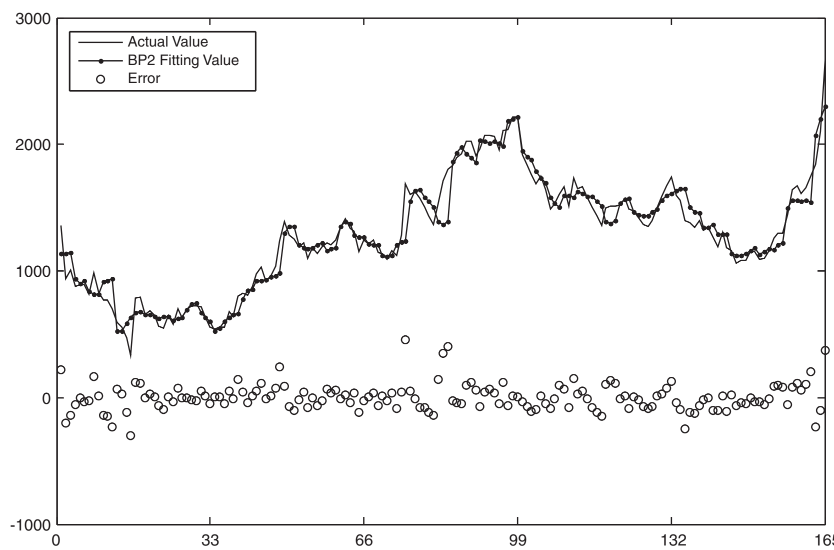

Fig. 9. The comparison chart of the actual value and BP2 fitting value.

where $p_{\max}$ and $p_{\min}$ are the maximum and minimum value of the original series, respectively.

3.2. Wavelet de-noising

From Fig. 4, we can see that the observed data are contaminated by a lot of noise. The noise can be removed from the observed data by DWT.

There are a variety of wavelets proposed in the literature for performing DWT. Each has its own application domain with unique resolution capability, efficiency, and computational cost etc. In this study, the computationally efficient Daubechies (db3) wavelet is used. The Daubechies (db3) wavelet is the common wavelet and its decomposition process is shown in Fig. 5.In Fig. 5, A_i and D_i (i = 1, 2, 3, 4, 5, 6) are the approximation and detail components, respectively. A_i's (i = 1, 2, 3, 4, 5, 6) represent the high-scale and low-frequency components of the time series and D_i's (i = 1, 2, 3, 4, 5, 6) the low-scale and high-frequency components. These approximation and detail records are reconstructed from the wavelet coefficients. The first high-pass filter provides the detail D_1. The first low-pass filter is approximation A_1, which is obtained by superimposing detail D_2 on approximation A_2. Furthermore, A_2, A_3, A_4 and A_5 are obtained by the similar iterations.

The appropriate number of levels of wavelet decomposition can be determined by the nature of a time series, according to dominant frequency components (Mörchen, 2003), entropy criterion (Coifman & Wickerhauser, 1992), or application's characteristics (Li & Shue, 2004). Generally speaking, the number of levels of decomposition depends on the length of the time series. For example, a six level decomposition is used in this study.

The noise of SCI closings prices sample data in our study has been reduced by the db3 wavelet's decomposition. Decomposition and reconstruction processes are shown in Figs. 6 and 7, respectively.

3.3. WDBP neural network

Although many different neural network models have been applied in finance, we have chosen to use the BP neural network in this paper due to its popular use in the short-term forecasting situations. Unlike other BP models directly constructed by the original data, we first decompose the original data into many layers by the wavelet

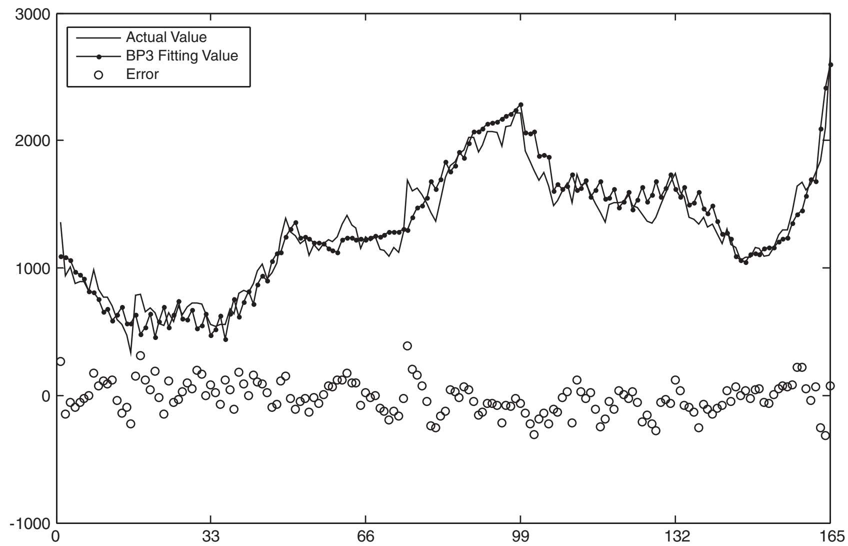

Fig. 10. The comparison chart of the actual value and BP3 fitting value.

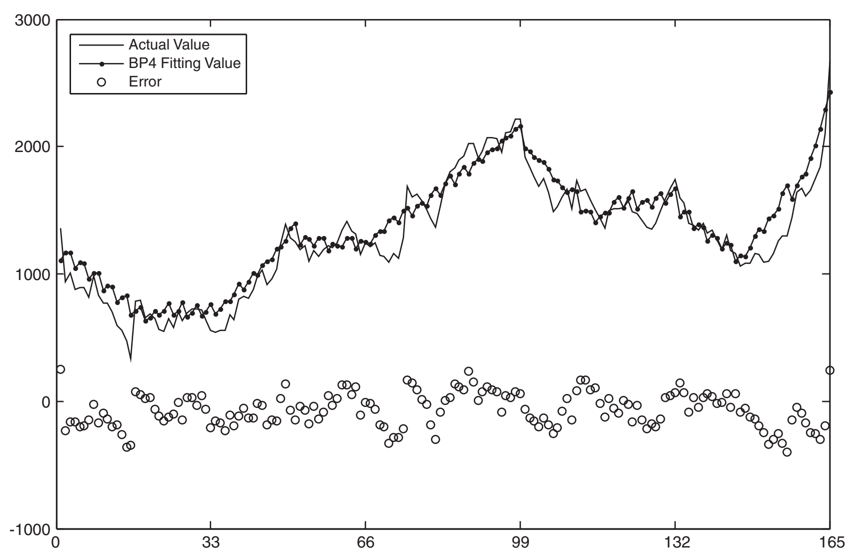

Fig. 11. The comparison chart of the actual value and BP4 fitting value.

transform. Every layer has a low-frequency signal and high-frequency signal. Then we establish a BP neural network model by the low-frequency signal of each layer and predict its future. In all BP models, we choose three values of every quarter as an input sample, and the total observations (the stock closings prices) are divided into 68 groups. The former 55 groups of every layer's low-frequency data is used for training and the remaining groups for testing.

In this paper, we chose a three-layer BP neural network which has a 3-neuron input layer, a 10 neuron hidden layer, and a 3-neuron output layer. Decomposing the original data into six layers, we establish the six models shown in Figs. 8–13.

In these Figs. 8–13, we show how BPi values fit the actual values, where BPi (i = 1, 2, 3, 4, 5, 6) denotes the model based on the ith layer of data. The errors are also shown in these figures.

3.4. Results and discussion

In order to verify the proposed WDBP method, we have conducted a forecasting experiment with the SCI closings prices. In this experiment, we compare our models with the BP network established by the original data without de-noising processing. This BP without de-noising values versus actual values are shown graphically in Fig. 14. In addition, the comparisons between the basic BP and our six models are presented in Table 1 and are graphically shown in Fig. 15. Table 2 shows accuracies of these models based on MAE, RME and MAPE and indicate that the BP4 model has the smallest errors among these models. Hence, BP4 should be chosen as our WDBP model. The WDBP model selected in this process outperforms the conventional BP model significantly.

4. Concluding remarks

This paper proposes an improved way of forecasting the stock closings price based on the neural network. In previous studies, neural networks are established with the original data for prediction. However, the stock market data are highly random and non-stationary, thus contain much noise. The prediction accuracy of traditional neural networks without the de-noising process is

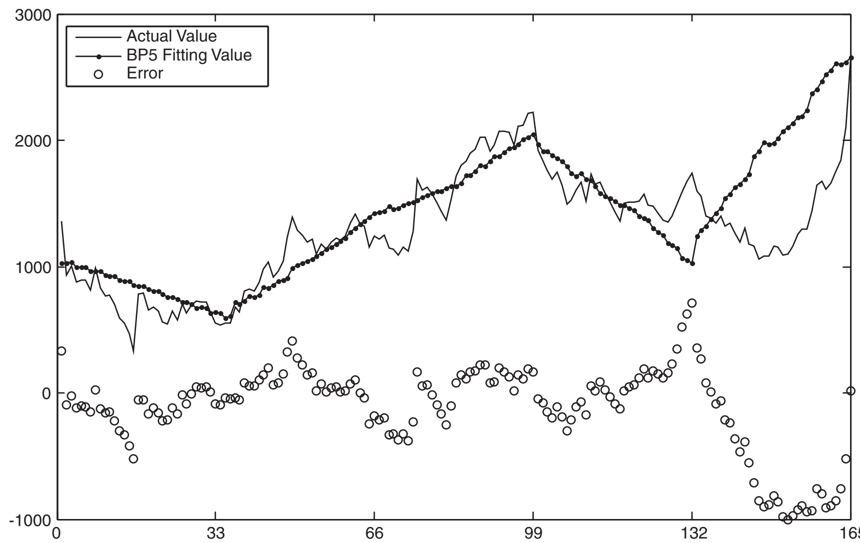

Fig. 12. The comparison chart of the actual value and BP5 fitting value.

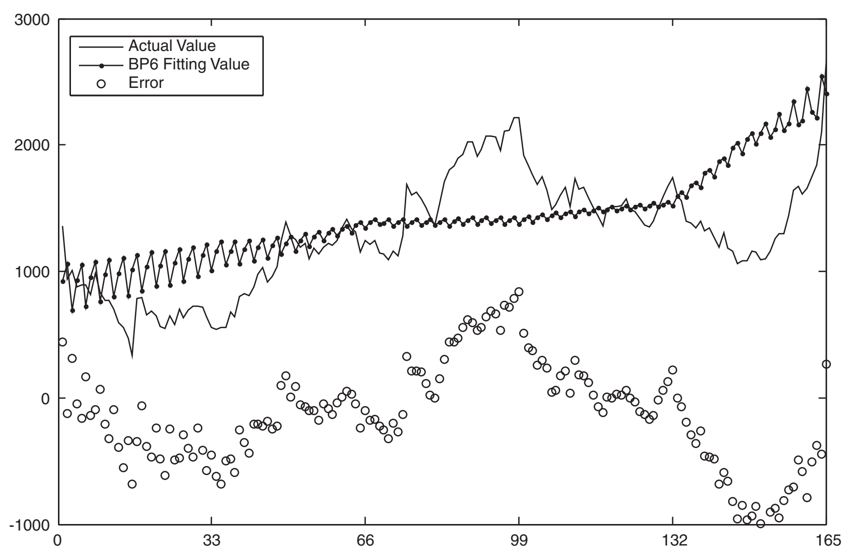

Fig. 13. The comparison chart of the actual value and BP6 fitting value.

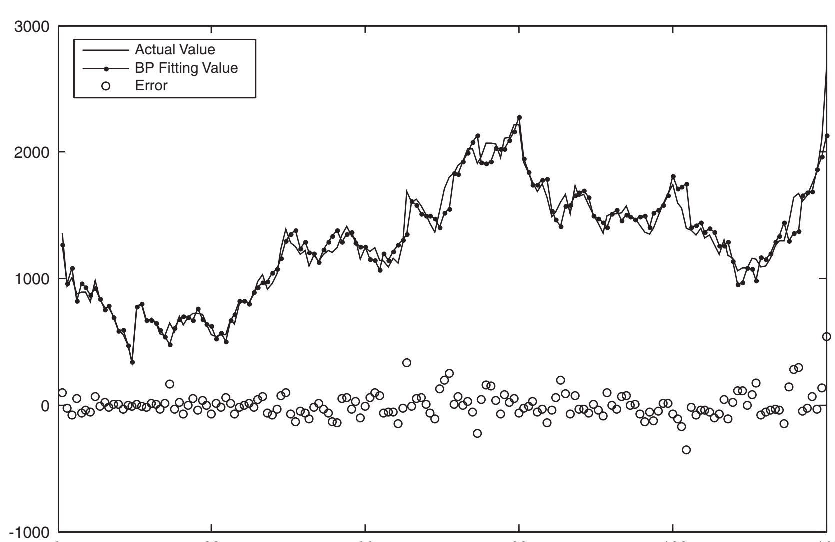

Fig. 14. The comparison chart of the actual value and BP fitting value.

Table 1
Comparing the predicted values of seven models (the BP*i* (*i* = 1, 2, 3, 4, 5, 6) network and the BP network).

Number	Time	Actual	BP	BP1	BP2	BP3	BP4	BP5	BP6
1	Jan-07	2786.34	2372.008	2190.09	2426.758	3129.364	2669.715	2592.685	2220.476
2	Feb-07	2881.07	2396.536	2426.313	2497.576	3505.848	2843.667	2580.487	2611.874
3	Mar-07	3183.98	2539.259	2689.835	2578.715	4082.62	3109.071	2613.389	2598.046
4	Apr-07	3841.27	2863.294	2085.804	2462.571	3546.475	3121.006	2619.506	2233.963
5	May-07	4109.65	2970.428	1739.596	2514.216	3985.493	3261.657	2624.032	2706.219
6	Jun-07	3820.7	3287.309	1531.385	2576.392	4743.108	3636.844	2649.681	2783.059
7	Jul-07	4471.03	2978.664	2058.575	2466.067	3262.927	3416.227	2660.398	2246.888
8	Aug-07	5218.82	2974.182	1600.961	2516.305	3541.268	3547.855	2707.301	2808.414
9	Sep-07	5552.3	3077.43	1302.705	2575.313	4092.433	4006.215	2715.889	2993.747
10	Oct-07	5954.77	3160.7	2059.712	2465.43	3261.144	3658.971	2674.954	2253.656
11	Nov-07	4871.78	3191.587	1601.633	2515.893	3532.715	3818.983	2717.764	2913.31
12	Dec-07	5261.56	3345.321	1302.91	2576.907	4078.095	4370.94	2726.515	3283.371
13	Jan-08	4383.39	2671.548	2047.312	2445.434	3246.611	3813.24	2681.93	2257.932
14	Feb-08	4348.54	2324.962	1588.765	2503.397	3509.434	3997.667	2725.288	3012.803
15	Mar-08	3472.71	1995.324	1290.585	2622.052	4043.666	4613.772	2732.402	3578.82
16	Apr-08	3693.11	2700.148	2019.745	2177.668	3697.985	2766.339	2762.913	2269.352
17	May-08	3433.35	2554.911	1560.347	2332.732	5348.569	2851.501	2892.879	3132.847
18	Jun-08	2736.1	2536.928	1263.514	3206.021	8782.262	3328.44	2866.599	3855.249
19	Jul-08	2775.72	2262.053	1545.676	2328.798	3375.344	2562.423	2807.252	2272.906
20	Aug-08	2397.37	2031.621	1063.971	2399.028	4942.449	2589.192	2969.296	3239.813
21	Sep-08	2293.78	1900.214	779.6376	2689.265	8234.721	2918.935	2929.419	4179.23
22	Oct-08	1728.79	2098.349	1479.176	1919.298	2835.693	2267.147	2848.619	2277.322
23	Nov-08	1871.16	2159.389	963.0936	1900.923	3887.044	2256.494	3039.129	3334.65
24	Dec-08	1820.81	2139.749	605.1982	2091.826	6005.922	2514.231	2991.506	4449.536
25	Jan-09	1990.66	1651.648	1989.116	1742.71	2760.343	2211.379	2872.987	2281.169
26	Feb-09	2082.85	1509.184	2158.615	1679.256	3039.229	2149.05	3077.423	3408.811
27	Mar-09	2373.21	1419.823	2351.878	1653.744	3425.232	2244.137	3030.119	4654.961
28	Apr-09	2477.57	2186.201	2117.907	2410.279	2676.114	1761.886	2915.987	2287.577
29	May-09	2632.93	2213.058	2060.366	2490.442	2819.038	1709.009	3159.782	3473.601
30	Jun-09	2959.36	2358.663	2084.589	2579.015	3089.402	1876.426	3102.085	4796.619
31	Jul-09	3412.06	2739.148	2088.145	2409.382	2732.674	2219.615	2865.352	2272.974
32	Aug-09	2667.74	2836.968	1753.818	2454.061	2871.431	2330.705	3011.47	3467.348
33	Sep-09	2779.43	3127.578	1555.223	2527.309	3156.292	2443.499	3025.843	4956.111

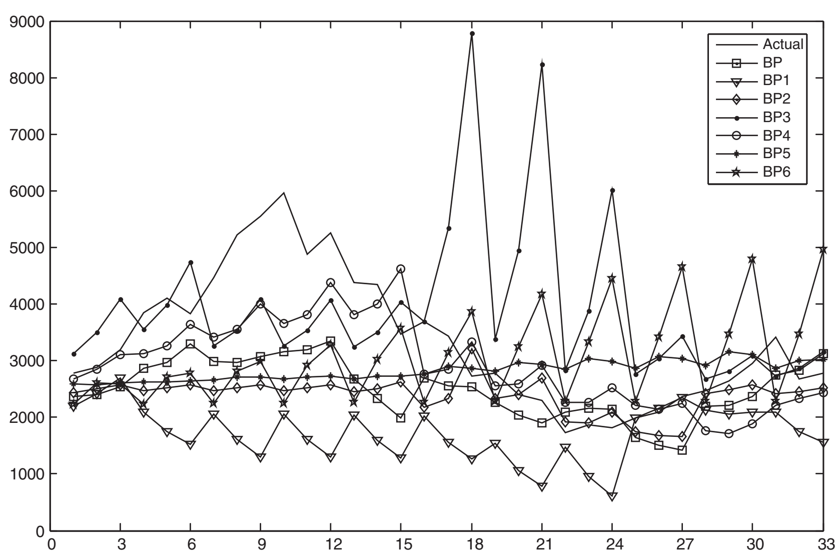

Fig. 15. The comparison chart of the actual value and the predicted value of the seven models.

Table 2
Comparing the three criteria of the seven models (the BP*i* (*i* = 1, 2, 3, 4, 5, 6) network and the BP network).

Errors	*BP*	*BP1*	*BP2*	*BP3*	*BP4*	*BP5*	*BP6*
MAE	930.2404	1620.6	1038.4	1340.3	675.9543	1067.1	1377.7
RMSE	1174.3	2006.2	1412.7	1984.4	847.5841	1349.7	1619.1
MAPE	0.2492	0.4364	0.2584	0.4831	0.1948	0.3056	0.4276

not satisfactory. The lack of a good forecasting model motivates us to find an improved method of making forecasts called WDBP model. In this method, we first decompose the original data into multiple layers by the wavelet transform, and every layer has a low-frequency signal and high-frequency signal. Then we establish a BP neural network model based on the low-frequency signal which is used to predict the future value respectively. Real data are used to illustrate the application of the WDBP neural network and show the improved accuracy of using the WDBO model.

Wavelet transform is an effective data pre-processing tool which may be combined with other forecasting methods, such as statistical and other AI models. Investigating these forecasting approaches could be a future research topic.

Acknowledgments

The research was supported by the Ministry of Education overseas cooperation "Chunhui Projects" under Grant Z2007-1- 62012 and the Fundamental Research Fund for Physics and Mathematics of Lanzhou University under Grant (LZULL200910). The third and corresponding author wishes to thank the support from NSERC Grant RGPIN197319 of Canada.

References

Armano, G., Marchesi, M., & Murru, A. (2005). A hybrid genetic-neural architecture for stock indexes forecasting. *Information Sciences, 170*(1), 3–33.

Azadeh, A., Ghaderi, S. F., & Sohrabkhani, S. (2008). Annual electricity consumption forecasting by neural network in high energy consuming industrial sectors. *Energy Conversion and Management, 49*, 2272–2278.

Chen, A. S., Leung, M. T., & Daouk, H. (2003). Application of neural networks to an emerging financial market: Forecasting and trading the Taiwan stock index. *Computers and Operations Research, 30*, 901–923.

Chun, S. H., & Kim, S. H. (2004). Data mining for financial prediction and trading: Application to single and multiple markets. *Expert Systems with Applications, 26*, 131–139.

Coifman, R. R., & Wickerhauser, M. V. (1992). Entropy-based algorithms for best-basis selection. *IEEE Transactions on Information Theory, 38*, 713–718.

Enke, D., & Thawornwong, S. (2005). The use of data mining and neural networks for forecasting stock market returns. *Expert Systems with Applications, 29*, 927–940.

Franses, P. H., & Ghijsels, H. (1999). Additive outliers, GARCH and forecasting volatility. *International Journal of Forecasting, 15*(1), 1–9.

Goswami, J. C., & Chan, A. K. (1999). *Fundamentals of Wavelets: Theory, Algorithms, and Applications*. Wiley Publishers, pp. 149–152.

Hansen, J. V., & Nelson, R. D. (2002). Data mining of time series using stacked generalizers. *Neurocomputing, 43*(1–4), 173–184.

Kim, K. J., & Han, I. (2000). Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Systems with Applications, 19*(2), 125–132.

Li, S. T., & Kuo, S. C. (2008). Knowledge discovery in financial investment for forecasting and trading strategy through wavelet-based SOM networks. *Expert Systems with Applications, 34*, 935–951.

Li, S. T., & Shue, L. Y. (2004). Data mining to aid policy making in air pollution management. *Expert Systems with Applications, 27*(3), 331–340.

Mörchen. F. (2003). *Time series feature extraction for data mining using DWT and DFT*. Department of Mathematics and Computer Science Philips-University Marburg. Technical report, 33.

Oh, K. J., & Kim, K.-J. (2002). Analyzing stock market tick data using piecewise nonlinear model. *Expert System with Applications, 22*(3), 249–255.

Sarantis, N. (2001). Nonlinearities, cyclical behavior and predictability in stock markets: International evidence. *International Journal of Forecasting, 17*(3), 459–482.

Shen, L., & Loh, H. T. (2004). Applying rough sets to market timing decisions. *Decision Support Systems, 37*, 583–597.

Thawornwong, S., & Enke, D. (2004). The adaptive selection of financial and economic variables for use with artificial neural networks. *Neurocomputing, 56*, 205–232.

Ture, M., & Kurt, I. (2006). Comparison of four different time series methods to forecast hepatitis A virus infection. *Expert Systems with Applications, 31*, 41–46.

Vellido, A., Lisboa, P. J. G., & Meehan, K. (1999). Segmentation of the on-line shopping market using neural networks. *Expert Systems with Applications, 17*, 303–314.

Wang, Y. (2002a). Predicting stock price using fuzzy grey prediction system. *Expert System with Applications, 22*, 33–39.

Wang, Y. (2003). Mining stock prices using fuzzy rough set system. *Expert System with Applications, 24*(1), 13–23.

Wang, Y. F. (2002b). Predicting stock price using fuzzy grey prediction system. *Expert Systems with Applications, 22*(1), 33–39.

Wang, Y. H. (2009). Nonlinear neural network forecasting model for stock index option price: Hybrid GJR-GARCH approach. *Expert Systems with Applications, 36*, 564–570.

Zhang, G. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing, 50*, 159–175.

Zhang, Y. D., & Wu, L. N. (2009). Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network. *Expert Systems with Applications, 36*, 8849–8854.