Wavelet neural networks for stock trading

Tianxing Zheng, Kamaladdin Fataliyev, Lipo Wang
School of Electrical and Electronic Engineering, Nanyang Technological University, Block S1, 50 Nanyang Avenue, Singapore 639798

ABSTRACT

This paper explores the application of a wavelet neural network (WNN), whose hidden layer is comprised of neurons with adjustable wavelets as activation functions, to stock prediction. We discuss some basic rationales behind technical analysis, and based on which, inputs of the prediction system are carefully selected. This system is tested on Istanbul Stock Exchange National 100 Index and compared with traditional neural networks. The results show that the WNN can achieve very good prediction accuracy.

Keywords: stock prediction; Artificial Neural Networks; Adaptive Wavelet Neural Networks

1. INTRODUCTION

The financial market is one of the greatest inventions in human history. It creates a platform for various economic entities to meet their specific needs such as investing, financing, risk-sharing, etc., through exchanging a great variety of financial products. Amongst many kinds of markets, stock market is the most common and accessible one for small investors. However, in order to benefit from, instead of suffering loss in the stock market, correct predictions must be sought. Without sophisticated decision-making processes, such as fundamental analysis and technical analysis, small investors often find it difficult to survive in the stock market. Artificial Neural Network (ANN) is one of the most commonly used techniques in stock market prediction[27].

An ANN is a set of inter-connected processing units – neurons[1]. The applications of ANN can be found in a number of fields, such as face recognition, data-mining and financial decision-making. In financial decision-making, credit risk evaluation is well researched[2]. On the other hand, time series prediction in stock market is a relatively new application of neural networks and is not yet commonly adopted in real world. Stock markets are highly uncertain and easily influenced by news, the intrinsic unpredictability of market participants, and many other complex issues, all contributing to the challenges in this application.

Wavelet neural network (WNN)[3] has been shown to achieve better prediction accuracy. In this report we will apply WNN in stock market prediction. To show how well the constructed system performs, the results will be compared against traditional ANN as well as the WNN built by previous researchers.

This paper is organized as follows: In Section 2, detailed explanations of the rationale behind traditional ANN and WNN will be given. In Section 3, we will review network construction and discuss how the design is selected. Section 4 is devoted for the applications in stock market prediction. The prediction result will be compared with results achieved by previous researchers. The main conclusions are presented in Section 5.

2. WAVELET THEORY AND ARTIFICIAL NEURAL NETWORKS

2.1 Wavelet theory

Wavelet is a set of functions that can localize an input function using two parameters, i.e., translation and dilation. The general form of wavelets is:

$$\psi_{\lambda,t}(x) = \frac{1}{\sqrt{\lambda}}\psi\left(\frac{x-t}{\lambda}\right). \tag{1}$$

In eq.1 λ represents dilation, t represents translation, and ψ is the original wavelet (mother wavelet). Translation represents how we would like to shift the wavelet along x-axis and dilation represents how wide or narrow we want the wavelet to be, if we put the independent variable on x-axis and dependent variable on y-axis.

Wavelet theory is widely applied in signal processing and time series analysis. The wavelet theory provides a unified framework for a number of techniques that had been developed independently for various signal-processing applications, e.g., multi-resolution signal processing in computer vision, sub-band coding in speech and image compression, and wavelet series expansions in applied mathematics[1]. Wavelet transforms are similar to Fourier transforms, which decomposes the input signal into different frequency components. The difference is that wavelet transforms can also cut the signal into different time ranges[4].

There are two main categories of wavelet transforms: Continuous Wavelet Transforms (CWT) and Discrete Wavelet Transforms (DWT), as in Fourier transforms. CWT decomposes the signal into infinite number of dilations and translations, providing a large amount of information. However, it is redundant and difficult for implementation. If t and λ in the wavelet transform are discrete, then this is called a discrete wavelet transform[5]. In fact, with the correct parameter set-up, DWT is powerful enough to preserve all the key features in the signal. The wavelet transform $\mathrm{W}(\lambda, \mathrm{t})$ of an input function $f(\mathrm{x})$ with respect to mother wavelet $\psi(\mathrm{x})$ is:

$$\mathrm{W}(\lambda, \mathrm{t}) = \langle \mathrm{x}, \psi_{\lambda,\mathrm{t}} \rangle = \int_{-\infty}^{\infty} \psi_{\lambda,\mathrm{t}}(\mathrm{x})\, f(\mathrm{x})\, \mathrm{dx}. \quad (2)$$

In DWT, for convenience, it is conventional that the dilations and translations of wavelets are sampled on a “dyadic” grid[6], meaning$\lambda = 2^{-\mathrm{j}}, \mathrm{t} = \mathrm{k}$ and $\mathrm{j}, \mathrm{k} \in \mathrm{Z}$, where Z is a set of integers.

2.2 Artificial Neural Networks

The processing power of ANN is achieved through adjusting the weight on the connections between neurons. By feeding the network with training data, which are a set of inputs and desired outputs, the network weights automatically adjust to match the inputs to the outputs – so called training process.

Each neuron receives inputs through the weights and, via an activation function, sends the output to the recipients. There are many kinds of activation functions, ranging from simple summation (or linear neuron) to the sigmoid function. There may be a bias in a neuron. The architecture of the network may also vary, such as 2-layer, 3-layer, recurrent (with output feeding back to input), etc...

2.3 Wavelet Neural Networks

Neural networks have very large potential in solving many complicated problems, including non-linear time-series prediction[7]. However, financial market movements is more difficult to catch and highly non-stationary, demonstrating many chaotic characteristics[8]. Since wavelets have demonstrated excellent capabilities in non-stationary signal analysis and nonlinear function modeling, the combination of the wavelet theory and the ANN, i.e., the Wavelet Neural Networks (WNN), has become a new research area[3, 9].

A WNN contains neurons and normally has one hidden layer. The hidden layer neurons use wavelets as their activation functions. These neurons are known as wavelons, with the activation function given by eq.1 for one-dimensional (1 input) cases.

It is commonly seen that the wavelon takes multiple inputs, in which case it is called multi-dimensional wavelon. The output of this wavelon then becomes the tensor product of the 1-dimensional wavelets[10], which has expression:

$$\mathrm{z} = \prod_{\mathrm{i}=1}^{\mathrm{n}} \psi_{\lambda,\mathrm{t}}(\mathrm{x_i}). \quad (3)$$

The architecture of an example of WNN is illustrated as:

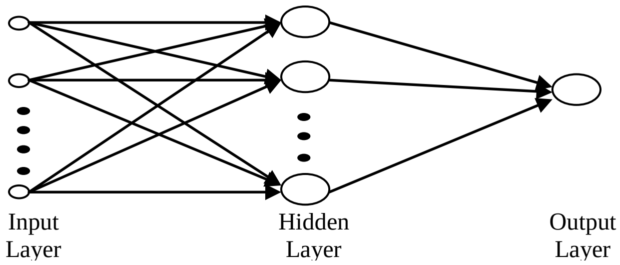

Figure1. Basic structure of WNN

The WNN in the diagram above has i inputs, j wavelons and the output function is:

$$y = \sum_j w_j z_j. \tag{4}$$

In the equation above, w_j is the weight on connection between hidden layer wavelon and output node, and z_j is the output of wavelon:

$$z_j = \prod_{i=1}^{n} \psi_{\lambda_j, t_j}(x_i). \tag{5}$$

Moreover, depending on the determination of dilations and translations on wavelons, WNNs are separated into two categories: adaptive wavelet neural networks and fixed grid neural networks[10]. Adaptive WNNs use wavelets from the CWTs. The dilation and translation for each wavelons are adjustable parameters in the network, just like weights. On the other hand, fixed grid WNNs use wavelets from DWTs and the dilations and translations are all fixed to predetermined levels, following the "dyadic" rule.

3. NETWORK CONSTRUCTION

Geva[11] cascades wavelet-based signal processing techniques and artificial neural networks to perform time-series prediction. In Geva's work, the discrete wavelet transform is applied to decompose the time-series signal into separate scales of wavelets. The resulted components are then put into separate ANNs to perform prediction in respective scales and the prediction results are then re-combined to form the prediction of original time series. This approach involves separate initialization and training in multiple ANNs. As a result, higher degree of complexity is expected in this model.

In comparison, Wang[12, 13] introduced a similar cascading architecture but with only one neural network in the model. The model – wavelet packet multilayer perceptron (WP-MLP) includes the wavelet packet transform which decomposes the input signal into different wavelet components and the resulted components are fed into a traditional MLP neural network. This approach reduced the complexity and improved the efficiency compared to Geva's approach[11]. The present paper focuses on traditional WNN (i.e. wavelet as activation function), instead of wavelet packets.

Zhao et al.[14] adopted a fixed WNN in predicting stock market returns, with fixed and pre-determined translation and dilations. The dilations and translations are set on a "dyadic grid".

Pindoriya et al.[9] and Garcia-Trevino[15] applied modified versions[17] of the so-called (1+1/2) - layer neural network proposed by Zhang and Benveniste[3]. Besides being connected to the hidden layer, the input layer has additional direct connections to the output layer, as in the random vector functional-link net proposed by Pao et al.[26]. The purpose of introducing this direct input-output connection is to help the learning of linear relationships between output and inputs. In addition, a bias term is added to output node introduced to help the learning of nonzero mean functions [15].

Selection of the baseline structure for the neural network is only the first step. Construction of a WNN includes not only the building architecture, but also involves issues regarding the selection of wavelets, learning algorithm, training method, initialization method of the networks, etc., to be discussed below.

3.1 Selection of mother wavelet

A number of mother wavelets are available, among which a commonly used one is the Mexican hat and is differentiable. We shall use the Mexican hat is selected as the mother wavelet:

$$\psi(x) = (1 - x^2)e^{-0.5x^2}. \tag{6}$$

3.2 Learning algorithm

In Zhang and Benveniste's paper[3], a learning algorithm – stochastic gradient algorithm, which is similar to back-propagation (BP) algorithm in the traditional ANN, is introduced. Firstly, the cost function is defined as:

$$C(\theta) = \frac{1}{2}(g_\theta(x) - y)^2. \tag{7}$$

In eq.7, $g_\theta(x)$ is the output of the WNN and y is the desired output. The final goal is to minimize the cost function, in another word, making predict outputs to be as close to desired outputs as possible. The minimization is realized using the stochastic gradient algorithm, which recursively modifies θ (i.e. the representative of WNN parameters, such as weights, biases, dilations and translations) along the opposite direction of the gradient of cost function. The gradient is calculated as using the partial derivatives of cost function with respect to each parameter. The actual adjustment of each parameter is then calculated as the negative of respective partial derivative times a learning rate α.

Training can be carried out either by batch training or on-line training. Batch training adjusts network parameters every epoch when all the training samples are fed into the network, whereas on-line training adjusts parameters immediately after each sample is fed into the network. Wilson and Martinez[16] have conducted an empirical test to compare these two methods. It is found that "batch training is much slower than on-line training especially on large training sets, as on-line training can follow the error surface closely and allow using larger learning rate". Hence, on-line training method is used in this paper.

In order to avoid over-fitting, validation is implemented in training. In the beginning of training process, 5% of training samples are randomly drawn out as validation samples. The mean square error of these samples is monitored in every epoch and if it starts to increase, training is stopped.

3.3 Initialization

Adaptive WNN includes wavelets whose dilations and translations can be adjusted in a continuous plane in order to achieve the best localization results, indicating the CWT nature in this design. On the other hand, since only a limited number of wavelons are included, the network is actually based on the DWT to decompose and reconstruct the signals. Therefore, dilations and translations should be initialized on the "dyadic grid" and fine-tuned during network training. Zhang[17] has proposed an initialization algorithm, which is adopted in this paper:

- Biases are initialized to the average of outputs in the training samples
- Weights are all initialized to zeros.
- Dilations and translations are initialized as follows recursively until all the wavelons are initialized:

$$t_1 = 0.5 \times (\mathrm{input}_{\min} + \mathrm{input}_{\max}), \lambda_1 = 0.5 \times (\mathrm{input}_{\max} - \mathrm{input}_{\min}) ;$$
$$t_2 = 0.5 \times (\mathrm{input}_{\min} + t_1), \lambda_2 = 0.5 \times (t_1 - \mathrm{input}_{\min});$$
$$t_3 = 0.5 \times (t_1 + \mathrm{input}_{\max}), \lambda_3 = 0.5 \times (\mathrm{input}_{\max} - t_1)$$

3.4 Numbers of input nodes, wavelons and output nodes

The number of input nodes depends on the dimension of input vector (i.e. how many inputs, the number of delays, etc.) and the number of output nodes depends on the dimension of output vector (i.e. how many steps the network is required to predict, etc.). From the initialization algorithm for dilations and translations, the number of wavelons should be $2^L - 1$ where L is the highest level of details. Again, as with input and output layers, this number depends on the nature of specific prediction problems and will be determined in later parts.

4. APPLICATION IN STOCK MARKET PREDICTION

The book "The Dow Theory"[18], written by Robert Rhea based on Charles H. Dow's theory and William Hamilton's refinement, is one of the most influential works in financial markets and laid the foundations for technical trading. The Dow Theory states that there exist trends of different levels in stock market, namely primary trend, secondary trend and minor (tertiary) trend.

Based on the Dow Theory, technical analysis tries to predict market movements using a variety of indicators, trend lines, etc.. The most important information in technical analysis is the moving average of past prices and trading volume. In fact, most of the commonly used technical analysis indicators are derived from this information. For example, OSCP (price oscillator) is calculated using moving averages, whereas "Disparity 5 Days" is calculated using daily price and moving average:

$$\mathrm{OSCP} = \frac{\mathrm{MA}_5 - \mathrm{MA}_{10}}{\mathrm{MA}_5} \tag{8}$$

$$\text{Disparity 5 days} = \frac{C_t}{\mathrm{MA}_5} \times 100. \tag{9}$$

4.1 Construction of Adaptive WNN stock prediction system

Before going into details of constructing the stock prediction system, it is necessary to clearly define how performance is measured. The most commonly used performance measure for neural networks is mean square error (MSE). Using N to denote the number of prediction values, the MSE of prediction is:

$$\mathrm{MSE} = \frac{1}{\mathrm{N}}\sum_{\mathrm{n}}\left(\mathrm{g}_{\mathrm{predict}} - \mathrm{y}_{\mathrm{target}}\right)^2. \tag{10}$$

Egeli et al. [19] used ANNs to predict ISE National 100 Index and an mean relative percentage error is used to measure performance, i.e., by how many percent the predicted output deviates from the desired output. This measure complements MSE and eliminates the biases toward smaller index values. In another paper[20], the same measure is used, but in a different name: mean absolute deviate in percentage (MAD %):

$$\mathrm{MAD}\ \% = \frac{1}{\mathrm{N}}\sum_{\mathrm{n}}\frac{\left|\mathrm{g}_{\mathrm{predict}} - \mathrm{y}_{\mathrm{target}}\right|}{\mathrm{y}_{\mathrm{target}}} \times 100\%. \tag{11}$$

A network's performance is often measured on how well the system predicts market direction[8]. While MSE and MAD measure by how much error occurs, they do not take falsely predicted directions into account. Therefore, a new measure needs to be introduced, i.e., success rate in detecting directions of movement, referred to as success rate:

$$\text{Success Rate} = \frac{\text{No.of Directions Correctly Detected}}{\mathrm{N}} \times 100\%. \tag{12}$$

4.1.1 Whether actual value or daily changes should be predicted

In some stock prediction methods, daily changes were used. Zhao et al.[14] used daily changes in Shanghai Stock Composite Index as the time-series to be predicted. In Zhao et al.[14], MSE is used as performance measure and the value achieved is 0.0139. However, as the range of daily changes in Shanghai Stock Exchange is within -10% ~ +10% and have relative small absolute values, the MSE might not represent the true performance of the system. Therefore, the test in Zhao et al.[14] is re-done to measure the success rate, using both the daily changes and actual values of the same data set (Shanghai Stock Composite Index from Jan 10th 2006 to July 18th 2008), in the network constructed in the previous chapter. The results are shown in Table 1.

Table1. Comparison between predicting daily changes and index values

	MSE	**Success Rate**
Daily changes	0.006875	46.2%
Actual index value	62291.2	51.3%

From the results shown in Table 1, it can be seen that even though test on daily changes yields much smaller MSE, the success rate is smaller than test on actual index value. The cause of this is possibly the high volatility and randomness of daily changes in stock index. Therefore, the system designed for this project will use actual index value as the output.

4.1.2 Determine the number of delays

We follow the recommendation made by Zhao, Zhang and Qi[14] and use five past prices. Further adjustments shall be based on trial-and-error in experiments.

In this test, in order to help making comparisons with other research, Istanbul Stock Exchange National 100 Index (later referred to as ISE) from June 2nd, 2001 to February 28th, 2003 will be used. The first 90% samples are used for training and validation while the last 10% are used as testing samples. Table 2 illustrates the effect of selecting different numbers of delays as input to the system.

From Table 2 it can be learnt that, as the number of delays increases, all of the performance measures improve. This suggests that as more information is fed into the system, a better-informed decision can be made. However, selecting too many delays is not necessary, since using five and six delays reduces the system performance. The possible reason is that as irrelevant information becomes input, the prediction system is distracted from focusing on truly important factors. Therefore, from this test, it can be determined that the index values on the last four days should be used as input.

Table2. Effect of changing the number of delays.

Number of Past Values	**1**	**2**	**3**	**4**	**5**	**6**
MSE	121677.1	63380.27	62218.50	61489.64	64231.66	62040.24
MAD %	2.72	1.86	1.79	1.79	1.89	1.84
Success Rate	48.78	56.10	58.54	65.85	63.41	63.41

4.1.3 Other inputs

There are some technical indicators that can be included as input. But it should be noted that the goal is to construct a system that uses as few inputs as possible while still achieving desired accuracy. It is of great important to understand that using too many inputs does not necessarily improve the prediction abilities of the system. Therefore, careful examination via empirical test should be conducted to select the correct input[28-32].

One of the well-known technical analysis methods is the moving average (MA). Moving average - as the name suggests - is the average of past index values during a certain period in the past[23]. Traders use moving averages in various ways: as trend line, or to find the most popular moving average crossovers. Moving average crossovers are the points when a moving average with shorter period cross a moving average with longer period, it suggests possible trend reversal. Since moving average is so important in technical analysis, it should be tested in our system to show its effect on prediction performance.

Hence, 10-day and 20-day moving averages are calculated for ISE National 100 Index and treated as inputs for testing purposes. The result are presented in Table 3.

Table 3. Effect of adding moving averages.

	Original System	**10-Day MA**	**20-Day MA**	**10-Day & 20-Day MA**
MSE	61489	64054	61942	60242
MAD %	1.8	1.9	1.7	1.7
Success Rate %	65.9	65.9	73.2	73.2

According to this test, 10-day moving average itself is not able to improve the prediction accuracy while 20-day moving average alone can significantly improve the success rate but increase Mean absolute Deviate in Percentage. On the other hand, when combined, 10-day and 20-day moving averages can achieve both smaller Mean absolute Deviate in Percentage (MAD %) and higher success rate. This is possibly because that the interacting effect of two different moving averages is captured by the network. Hence, they are both selected as input.

4.1.4 Number of prediction steps

In almost every country, investors are required to pay stamp tax, which is levied as portion of the value of trade. Moreover, most of the retail investors trade via brokers, who charge commissions. As a result, short-term speculation might not be suitable for most of the investors as the transaction costs are too high, unless the stock price changes are very large. So longer-terms would allow investors to achieve better gain, since capital gain can accumulate to cover transaction costs. Therefore, it is preferable for the prediction system to predict more steps if accuracy is not significantly compromised. In this sub-section, we study how far in the future the constructed system can predict the stock index value. The result is shown in Table 4.

Table 4. Effect of predicting steps.

Number of Future Values	**1**	**2**	**3**	**4**	**5**	**6**
MSE	60242	76512	88539	102490	72584	112886
MAD %	1.7	1.9	1.9	2.1	1.8	2.4
Success Rate %	73.2	73.2	75.6	71.8	64.1	58.5

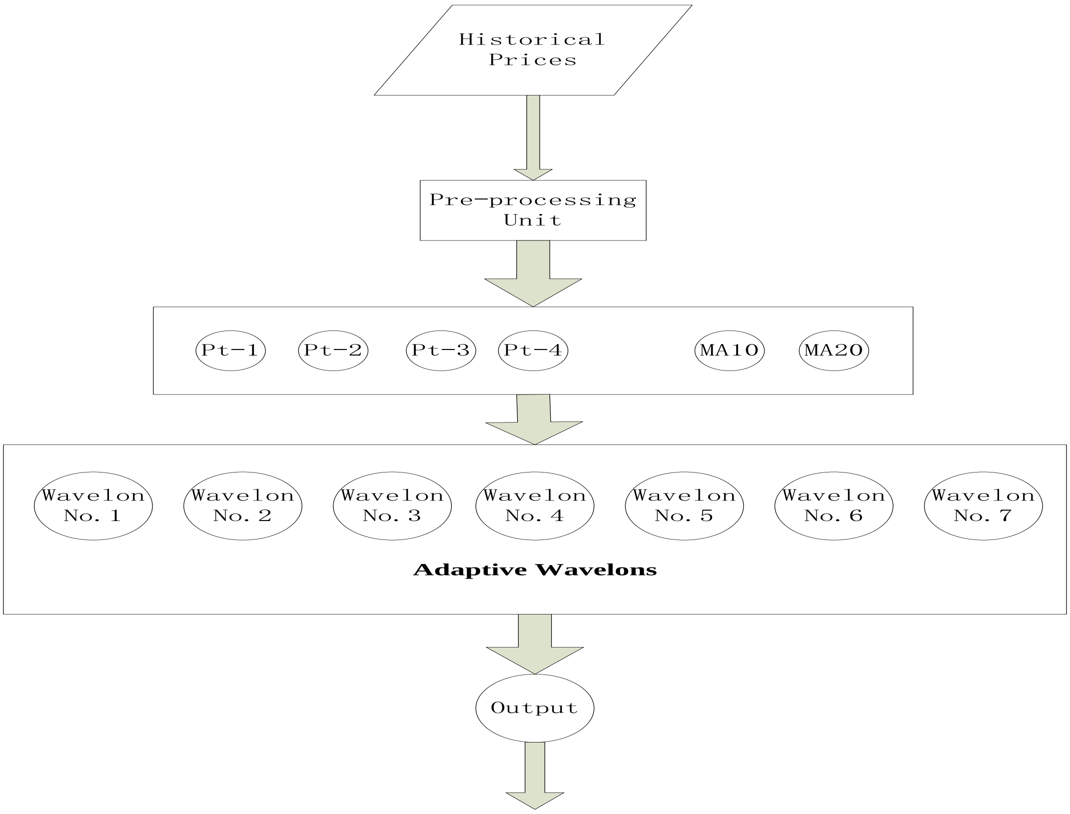

Figure 2. Final Configuration of Adaptive Wavelet Neural Network Prediction System

The result in Table 3 shows that using the system to predict 3 steps (i.e. predicting the next three days' index values) yield the highest success rate. The possible reason is that the information captured can serve up to three days, and on the third day the actual value is closest to the expected value calculated by the system. This is actually consistent with Adaptive Expectation Theory [24]: it takes time for market participants to adapt their expectations towards correct value. However, three-step prediction does increase the MAD% compared with single-step prediction. It is possibly because, in order to be able to predict three days, some small movements in shorter period is ignored by the system. So there exists a trade-off between success rate and MAD %, and which one to focus on depends on the objective of the investor.

4.2 Final system

After adjustment, the WNN stock prediction system has its final configuration. Prices or index values on the previous four days, 10-day and 20-day moving averages are taken as inputs. When performing single-step prediction, the system can achieve smallest error, and when performing three-step prediction, better success rate of detecting direction of movements is optimized. The schematic diagram is presented in Fig.2.

This final system is tested on Istanbul Stock Exchange National 100 Index. The final plot of the response is in Fig.3. In the plot, solid curve represents the prediction output and the dashed curve represents the actual value. This final result is compared with Egeli et al.'s[19] MLP design. Also, the prediction result using 5-day Moving Average Prediction Model is also retrieved from that research[19] for comparison (Table 5).

According to Table 5, it can be shown that Adaptive WNN has achieved the similar level of mean absolute deviation in percentage as the MLP model proposed by Egeli et al.[19]. The WNN system constructed in this paper is considered to be more efficient. It is to be noted that the MLP model includes not only information related to historical price, but also interest rate, exchange rate and five variables representing Monday to Friday. So Adaptive WNN has demonstrated its superior power to extract useful information from input, as it uses fewer inputs to achieve similar accuracy.

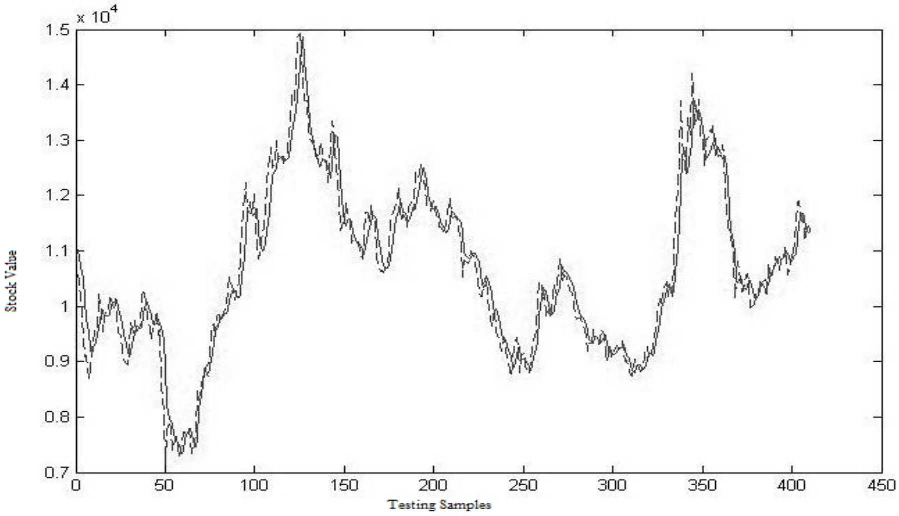

Figure 3. Final Response of applying Adaptive WNN on Istanbul Stock Exchange National 100 Index.

5. CONCLUSION

In conclusion, a stock prediction system based on an Adaptive Wavelet Neural Network has been constructed and used in stock prediction. It has been shown that Adaptive WNN has advantages over the traditional MLP neural network. Index values on the previous four days, 10-day and 20-day moving averages are carefully selected as inputs to maximize the network's predicting power. A final test is performed on Istanbul Stock Exchange National 100 Index and compared with Egeli et al.'s[19] MLP design, demonstrating a good and efficient performance.

Table 5.Final Comparison of MLP design, MA-5 day model, and the Adaptive WNN in this paper.

Model	**MAD %**
Adaptive WNN	1.65
MLP	1.62
MA – 5 days	2.17

REFERENCES

[1] Wang, L. P. and Fu, X. J., [Data Mining with Computational Intelligence], Berlin, Springer (2005).

[2] Baesens, B., et al., “Using Neural Network Rule Extraction and Decision Tables for Credit-Risk Evaluation”, Management Science 49 , 312-329 (2003).
[3] Zhang, Q. and Benveniste, A., “Wavelet Networks”, IEEE Trans. Neural Networks 3(6), 889-898 (1992).
[4] Daubechies, I., [Ten Lectures on Wavelets], SIAM (1992).
[5] Lin, C. L., Shieh, N. C., Tung, P. C., "Robust wavelet neuro control for linear brushless motors", IEEE Transactions on Aerospace and Electronic Systems 38(3), 918-932 (2002).
[6] Iyengar, S. S., Cho, E. C. and Phoha, V. V., [Foundation of Wavelet Networks and Applications], Boca Raton, FL, Taylor and Francis (2002)
[7] Lapedes, A. and Farber, R. , “Nonlinear Signal Processing Using Neural Network: Prediction”, Los Alamos Nat. Lab Tech. Rep, (1987)
[8] Lawrence, R., “Using Neural Networks to Forecast Stock Market Prices”, Manitoba, BC, Canada, Univ. of Manitoba, (1997).
[9] Pindoriya, N. M., Singh, S. N. and Singh, S. K., “An Adaptive Wavelet Neural Network-Based Energy Price Forecasting in Electricity Markets”, IEEE Transactions on Power Systems 23, 1423-1432 (2008)
[10] Billings, S. A. and Wei, H. L., “A new class of wavelet networks for nonlinear system identification”, IEEE Trans. 16, 862-874 (2005).
[11] Geva, A. B., “ScaleNet - MultiScale Neural Network Architecture for Time Series Prediction”, IEEE Trans. on Neural Networks 9 (6), 1471-1482 (1998).
[12] Wang, L. P., et al., "Predicting time series using wavelet packet neural networks", Proc. IJCNN, 1593-1597 (2001).
[13] Teo, K.K., Wang, L.P., Lin, Z.P. "Wavelet packet multi-layer perceptron for chaotic time series prediction: effects of weight initialization," Computational Science -- ICCS 2001, Proceedings Pt 2, Volume: 2074, 310-317, 2001.
[14] Zhao, Y., Zhang, Y. and Qi, C., “Prediction Model of Stock Market Returns Based on Wavelet Neural Network”, IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application (2008).
[15] Garcia-Trevino, E. S. and Alarcon-Aquino, V., “Single-Step Prediction of Chaotic Time Series Using Wavelet-Networks”, IEEE Electronics, Robotics and Automotive Mechanics Conference (2008).
[16] Wilson, D. R., and Martinez, T. R., “The general inefficiency of batch training for gradient descent learning”, Neural Networks 16, 1429-1451 (2003).
[17] Zhang, Q., “Wavelet network: the radial structure and an efficient initialization procedure”, European Control Conference, Groningen (1993).
[18] Rhea, R. , [The Dow Theory], Daniel-Smith Co. (1932)
[19] Egeli, B., Ozturan, M., Badur, B., “Stock Market Prediction Using Artificial Neural Networks”, Proceedings of the 3rd Hawaii International Conference on Business, Hawai, USA. (2003)
[20] Guresen, E., Kayakutlu, G. and Daim, T.U., “Using artificial neural network models in stock market index prediction”, Expert Systems with Applications (2011)
[21] Aguirre, L.A., “A nonlinear correlation function for selecting the delay time in dynamical reconstructions”, Physics Letters A. 293, 88-94 (1995).
[22] Gurley, K., Kijewski, T. and Kareem, A., “First and Higher-Order Correlation Detection Using Wavelet Transforms”, Journal of Engineering Mechanics, 188-201 (2003).
[23] Dagnino, G., “Technical Analysis, the Markets and Moving Averages”, The Peter Dag Portfolio Strategy & Management (2013).
[24] Evans, G. W. and Honkapohja, S., [Learning and Expectations in Macroeconomics], Princeton University Press (2001).
[25] Zhang, J., et al., “Wavelet Neural Networks for Function Learning”, IEEE Transactions on Signal Processing 43, 1485-1497 (1995).
[26] Pao, Y. H., Park, G. H. and Sobajic, D. J., “Learning and generalization characteristics of random vector functional-link net”, Neurocomputing 6, 163-180 (1994).
[27] Gupta, S., Wang, L. P., “Stock Forecasting with Feedforward Neural Networks and Gradual Data Sub-Sampling,” Australian Journal of Intelligent Information Processing Systems 11, 14-17 (2010).
[28] Wang, L. P., Zhou, N., and Chu, F., “A general wrapper approach to selection of class-dependent features,” IEEE Transactions on Neural Networks 19 (7), 1267-1278 (2008).
[29] Zhou, N., and Wang, L. P., “Effective selection of informative SNPs and classification on the HapMap genotype data,” BMC Bioinformatics 8:484 (2007).

[30] Wang, L. P., Chu, F., and Xie, W., “Accurate cancer classification using expressions of very few genes,” IEEE-ACM Transactions on Computational Biology and Bioinformatics 4 (1), 40-53 (2007).
[31] Fu, X.J., and Wang, L. P., "Data dimensionality reduction with application to improving classification performance and explaining concepts of data sets", International Journal of Business Intelligence and Data Mining (invited paper) 1 (1), 65-87 (2005).
[32] Fu, X.J., and Wang, L. P., "Data dimensionality reduction with application to simplifying RBF network structure and improving classification performance", IEEE Trans. System, Man, Cybern, Part B-Cybernetics 33, (3), 399-409 (2003).