# Stock Market Index Prediction Using Artificial Neural Network

Falah Hassan Ali Al-Akashi, University of Kufa, Iraq*

https://orcid.org/0000-0002-2342-7643

## ABSTRACT

Often, nonlinearity exists in the financial markets while artificial neural network (ANN) could be used to expect equity market returns for the next years. ANN has improved its ability to forecast the daily stock exchange rate and to investigate several feeds using the back propagation algorithm. The proposed research utilized five neural network models, Elman network, multilayer perceptron (MLP) network, Elman network with self-optimizing map (SOM), MLP with SOM filter, and simple linear regression for estimating new values. Results were examined to investigate the predicting ability and to provide effective feeds for future values. The result of the proposed simulation showed that SOM could greatly improve the convergence of the neuron networks whereas Elman network did a better performance to capture the temporal pattern of the symbolic streams generated by SOM. A benchmark of linear regression model was also employed to show the ability of neural network models to generate higher accuracy in forecasting financial market index.

## KEYWORDS

E-Commerce, Future Forecasting, Linear Regression, Modelling Artificial Map, Time-Series

## 1. INTRODUCTION

The ability of artificial neural network (ANN) in forecasting the daily market stock exchange rate was investigated (Moghaddam et al., 2016; Hedayati et al., 2016). An international exchange market is one of the most invested markets in the world with an average daily trade volume of $1.8 trillion (Ahmed and Hokey, 2009). Researchers have shown that many commonly cited signals have had very weak and erratic correlations with actual subsequent returns, even at long investment horizons (Joseph et al., 2012). Large changes have taken place over the while in the market places, involving the use of significant communication and commercial platforms that empowered the number of investors entering the markets. Traditional capital market theory has also changed, and methods of financial analysis have improved (Laurent, 1979). Stock-return forecasting has drawn the attention of researchers for several years and typically conveys an idea that essential information generally available in the past has some predictive connections to future stock returns or indices. The samples of such information include: economic variables, exchange rates, industry and sector-specific information, and individual corporates financial statements. This is opposed to the general ideal of the efficient market hypothesis

*Corresponding Author

which states that all available information affecting the current stock value is constituted by the market before the general market builds trades based on it. Hence, it is hard to forecast future returns for a reason that stocks already reflect all known data. This is still an empirical problem because there is a contradictory evidence that markets are not totally efficient, and that is possible to guess the future returns with results that are better than random by means of generally available information such as time-series data on financial and economic variables (David, 1988).These studies assumed that variables e.g. interest rates, monetary-growth rates, changes in industrial production, and inflation rates are statistically necessary for predicting some of the stock returns. However, most of the studies simply mentioned that effort to capture the conjunction between the current information and also the stock returns depends on simple linear regression assumptions despite the fact that there's no proof for the linear connection between the stock returns and the monetary and economic variables. Since there existing significant residual variance of the actual stock return from the prediction of the regression function, it is possible that nonlinear models should be used to demonstrate this residual variance and turn out additional reliable predictions of the stock value movements. The stock probably leads to a downturn in the housing market because business' share prices have engaged very high relatives to their outcomes. Analysts who made that point of view in the past earned high price ratios have usually followed by not really fast growth in stock prices (Pu, 2000). Neural network is widely used to overcome such problems as a nonlinear modelling technique. Neural network offers a novel technique that does not require pre-specification during the modelling process since they independently learn the inherent relationship between the variables (Amin et al, 2018). This is particularly useful in protection investment and other financial fields where much was proposed and little was known about the type of the processes specifying asset prices. Neural networks propose different infrastructure models and learning platforms, in which, classification and level estimation in both multi-layer feed-forward neural networks and recurrent neural network were briefly reviewed. As a consequence, as shown in this study, using different learning models in artificial neural network would forecast time-series of future value more efficiently. Current studies and previous studies have had forecasted markets in specific periods or forecasting data in specific markets.

A business market index is a listing of stocks that reflects the index value of its elements. There are many major market indices in finance world, and each of those tracks the performance of a specific group of stocks considered to represent a particular market or sector of the business. The Dow Jones Industrial Average (DJIA) is example of the index of the almost 30 largest and most widely holds general organizations in the United States. The Index holds valuable industrial firms with a history of successful development and lots of investor interests. Also, the Index holds a wide range of firms from commercial services, to computer companies, to retail companies. A typical plot of daily information for Dow Jones Industrial Average looks like Figure 1. Like most data (such as asset prices, exchange rates, GDP, inflation and other macroeconomic indicators) in economics and finance, a stock market

**Figure 1. Time Series of the Dow Jones Industrial Average (DJIA)**
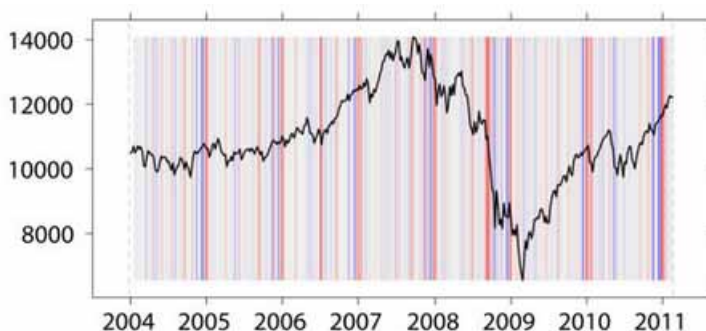
index comes in the form of time-series exhibiting very high noise and significant non-stationary and non-linearity. Stock prices have been followed by the high price-earnings ratios of slow long-run growth. Moreover, when high price-earnings ratios have minimized the earnings that yield stocks relative to return on other investments, short-run stock market performance has affected as well (Pu, 2000). Every aspect will be discussed later in this study and techniques will be employed to overcome those difficulties.

The rest of paper is organised as follows: Section 2 outlines some related works to our steady. Section 3 describes how the efficient market hypothesis. Section 4 highlights the system details. Section 5 shows the stationary in data and process. Section 5 details the experimental results, and finally, Section 6 will provide conclusion remarks and future direction.

## 2. LITERATURE SURVEY

Artificial Neural Networks (ANNs) are widely deployed in the domain of stock price prediction. Kuo et al., (2001) evolved a decision support system by combining a genetic algorithmic model based fuzzy neural network (GFNN) and ANN for stock exchange. The designed system was assessed using the information of Taiwan stock exchange. Chen et al., (2003) involved Probabilistic Neural Network (PNN) to forecast the direction of Taiwan index return. They investigated that PNN has higher performance available index than generalized strategies of moments-Kalman filter and stochastic process prediction models. Atsalakis and Valavanis (2009) evolved an adaptive neuro-fuzzy logical thinking controller to forecast next day's stock worth trend. They showed the potential ability of ANFIS in predicting the stock forecasting. Guresen et al., (2011) reported the performance of multi-layer perceptron (MLP), dynamic ANN, and hybrid ANN algorithms in prediction the market values. Qiu, Liu, and Wang (2012) developed a brand new prediction model on the idea of fuzzy statistic and C-fuzzy call trees to predict index number of shanghai stock index. Karlsson, S. and Nordberg, M. (2015) proposed artificial neural networks for training on foreign markets index prediction. The study aimed to compare with a domestic artificial neural network. The ANNs trained on five different stock market indices (Denmark, Germany, Japan, Sweden and USA) and then cross-tested on the other markets to investigate what impact on the forecasting performance. They suggested that more research is needed to be able to draw any definitive conclusions. Amin et al., (2016) Stock market index prediction using artificial neural network in forecasting the daily NASDAQ stock exchange rate. The methodology used in that study proposed the short term historical stock indices as well as daily input parameters (last four and nine working days). Their investigation outputs showed that there is no distinct difference between the prediction ability of the four and nine prior working days as input parameters.

## 3. EFFICIENT-MARKET HYPOTHESIS

The Efficient Market Hypothesis (EMH), alternatively known as the efficient market theory, maintains share prices that reflect all information and consistent alpha generation is impossible. Fama (1965) had developed such hypothesis which was an academic methodology of study that later became a cornerstone of modern financial theory. The EMH showed that it was impossible to consistently outperform the market value by using any information of market that already knew it. Stock market efficiency often collaborates and reflects all relevant information by sharing the existing prices. This means stocks always trade at their fair value on stock exchanges, and thus it is impossible for investors to profit from either purchasing undervalued stocks or selling stocks for inflated prices.

The EMH had produced a theoretical basis for much of the financial market research during the seventies and the eighties, but it has been put on trial recently. The potential attacks on the efficient market hypothesis and the belief that stock prices are partially predictable, there was a large evidence for supporting EMH and an equal amount of dissension also existed (Burton, 2003). For example,

the existence of apparently sophisticated professional investors like Warren Buffett is impossible according to the EMH. Critics of EMH (La et al., 1997) argued that the predictability of stock returns reflects the psychological factors, social movements, noise trading, and other irrational factors in a speculative market price. The crux of the EMH is that it should be impossible to forecast trends or prices in the firms through vital analysis or technical analysis. If the proponents of EMH posited, then a monetary time-series could be modelled like the accumulation of noisy element at each step:

$$y(k + 1) = y(k) + \epsilon(k) \tag{1}$$

where k denotes a zero mean Gaussian variable with variance σ.

The best estimate for the value is:

$$\bar{y}(k + 1) = y(k) \tag{2}$$

Similarly, if the series was a correct random walk, then the best estimation for the next time-series was equal to the current estimation. Now, if it is proposed that there was a predictable element of the series, then it was possible to use:

$$y(k + 1) = y(k) + f(y(k), y(k-1),…,y(k - n + 1)) + \epsilon(k) \tag{3}$$

where $\epsilon(k)$ denotes a zero mean Gaussian variable with variance σ, and f denotes a non-linear function in its arguments. The best estimation is given by:

$$\bar{y}(k + 1) = y(k) + f(y(k); y(k-1),…, y(k - n + 1)) \tag{4}$$

Estimation under such model was a problematic like the series usually contains trends. An alternative solution to this was to involve a method would comply with the first order differences instead of the raw time-series:

$$\delta(k + 1) = \delta(k) + f(\delta(k), \delta(k-1),…, \delta(k-n + 1)) + v(k) \tag{5}$$

where $\delta^{o} y(k + 1) - y(k)$ and v(k) is a zero mean Gaussian variable with variance σ, and in this case, the best estimation is:

$$\delta(k + 1) = \delta(k) + f(\delta(k), \delta(k-1),…, \delta(k-n + 1)) \tag{6}$$

## 4. THE PROPOSED METHODLOGY

To exploit the hypothesis in the previous section, we will start with data pre-processing, which consists of all the procedures taken prior the actual data analysis process. It is essential that T transforms the raw real world data vectors to a set of new data vectors . Figure 2 depicts the hybrid intelligent

Figure 2. High-Level Block Diagram of the System
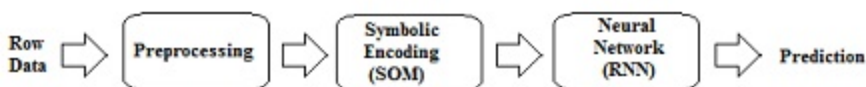
system model for stock market analysis. Thereafter, data is fed into a Self-Optimizing Map (SOM). The output of the Self-Organizing-Map is the topographical portion of the winning node, each node represents unique item. A brief description of the self-organizing map is outlined in a later section.

An Elman recurrent neural network was thereafter utilized to train on the series of outcomes from the Self-Organizing-Map. The Elman network was selected as a result of appropriation for the matter, and showed to perform well compared to different architectures. As shown in Figure 3, the Elman neural network has a connection from each of the hidden nodes to all of the hidden nodes.

In feed-forward *neural networks*, neurons are connected together in form of a network. Information flows from input neurons into the network in *one* direction Delay embedded series was changed into symbols using a Self-Organizing Map and a repeated neural network is trained on the series of neurons. The low level theory of Elman network is as follows:

## 4.1 Data Processing

*1- Differencing*: Large scale deterministic components, such as trends and seasonal variations, should be eliminated from the inputs whenever possible. For this case, the differences $[y_k - y_{(k-1)}]$ of

**Figure 3. The Pre-Processing Feeds of Elman Network**
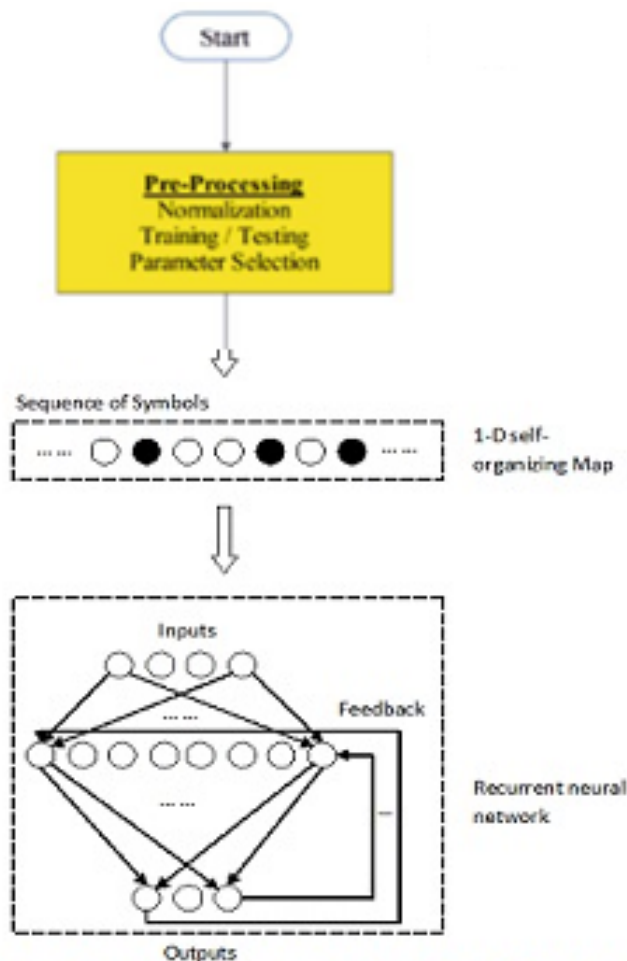
the variables were produced to the networks in which different inputs' variables could be compared in terms of relative changes to a while stock returns, as a result of the relative change of variables potentially more useful to the models than the original values if forecasted a financial time-series.

**2- *Log compression*:** In order to compress the dynamic range of the series and reduce the effect of outliers, a log transformation of the data is used: $x(k) = sign(\delta(k)) \log(|\delta(k)| + 1)$

## 4.2 Kohenen Network

A Self-Organizing Map (SOM) is a kind of Artificial Neural Network (ANN) that was trained using unsupervised-learning algorithm to produce a low dimensional value (typically two dimensions), discrete-representation of the input value of the training samples. The input value looks for to produce the topological properties of the map- space. The model was first showed as an artificial neural network and proposed by the Teuvo Kohonen and sometimes named a Kohonen map.

Like most artificial neural networks, Kohenen Map operates in two levels: Training and Mapping. Training builds the map using input examples. It is a challenge process or vector quantization. Mapping automatically classifies a new input vector. Consider the matter of charting associate n-dimensional space using a one-dimensional chain of Kohonen units. The units are all arranged in sequence and are numbered from 1 to m (Figure 4). Each unit is associated to the n-dimensional input x and counts the corresponding excitation. The n-dimensional weighting vectors $W_1$, $W_2$ ..., $W_m$ were involved for the computation. When an input from such a region is fed into the network, the corresponding unit could compute the maximum excitation. The n-dimensional weighting vectors $W_1$, $W_2$ ... $W_m$ were involved for the computation. When an input from such a place is fed into the network, the certain unit could be computed by the maximum excitation.

A Kohonen map computes the Euclidian distance between an input y and its weighted vector w. This new definition of excitation was more suitable for specific applications and also easier to visualize. In the Kohonen map of one-dimensional network, the neighbourhood of radius r of unit k involves all units reached up to r locations from k to the left or to the right of the chain. Values at both ends of the chain had asymmetrical neighbourhoods. The Kohonen learning hold a neighbourhood function Ø, whose value Ø(i, k) denotes the strength of the pairing between unit i and unit k during the training process. The learning algorithm for the Kohonen network is as follows:

**Figure 4. Lattice Reduction M-Dimension**



(a)          (b)
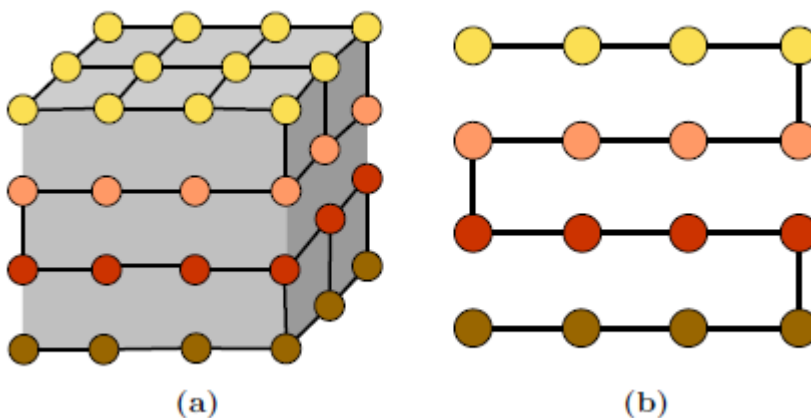
***Start:*** The n-dimensional weight vectors $(W_1, W_2, \ldots, W_m)$ of the *m* computing criterion were chosen at random. An initial radius *r*, a learning constant ŋ, and a neighbourhood function Ø were selected.

***Step* 1:** Select an input vector using the desired probability distribution over the input space.

***Step* 2:** The unit k with the maximum excitation is selected (that is, for which the distance between $W_i$ and Ɛ is minimal, $i = 1 \ldots m$).

***Step* 3:** The weight vectors were updated using the neighbourhood equation and the updated rule

$$W_i = W_i + \text{Ø}(\text{i, k})(\text{Ɛ} - W_i) \text{ , for i} = 1,.., m$$

***Step* 4:** Stop if the maximum number of iterations has been reached; otherwise modify ŋ and Ø as scheduled and continue with step 1.

After quantizing real-valued time series into symbolic streams, there were many ways to interpret the resulting symbolic. In particular, it performs quantization into symbolic streams over two and four symbols, respectively. The sequence $S_t$ over the binary alphabet {1, 2} is as follows:

$$S_t = \begin{bmatrix} 1 \left(down\right) & \delta_t < 0 \\ 2 \left(up\right) & \delta_t \geq 0 \end{bmatrix}$$

Quantization using four symbols is more complicated, since it has to determine the positions of the cut values Ø1 and Ø2 separating normal from external differences. The sequence $S_t$ over the alphabet {1, 2, 3, 4} is as follows:

$$S_t = \begin{bmatrix} 1(extremely\ down) & if\ \delta_t < \theta_1 < 0 \\ 2(normal\ down) & if\ \theta_1 < \delta_t < 0 \\ 3(normal\ up) & if\ 0 < \delta_t < \theta_2 \\ 4(extremely\ up) & if\ 0 < \theta_2 < \delta_t \end{bmatrix}$$

## 4.3 Recurrent Neural Network

Recurrent Neural Networks (RNNs) are not like Multi-Layer Perceptron (MLP). For time series forecasting, Neural Network architectures can be trained to predict the future values of the dependent variables. Wang (2011) assumed a new method called HLP as data prediction using preprocessing algorithm to process the stock data. HLP method predicts the stock high and low point with different frequency and amplitude. The extracted data describes the feature of stock price movement. Thereafter, we built ANN models to reflect the stock movement running and price. The HLP algorithm and ANN methods give motivation to investors. What required is the design of the network paradigm and its parameters, and the back propagation algorithm had been used for assessing several feed forward ANNs. The methodology used by researchers (Moghaddam et al., 2016; Hedayati et al., 2016) considered the short-term historical stock prices as well as the day of week as inputs. The multi-layer feed-forward neural network approach (Figure 5) consists of an input layer, one or several hidden layers and an output layer.

Another approach is known as the partially recurrent neural network that can learn sequences as time evolves and responds to the same input pattern at different times, depending on the previous input patterns as well. Some argue that the use of a recurrent neural network instead of a Multilayer

**Figure 5. Non-Recurrent Network (Multilayer Perception)**
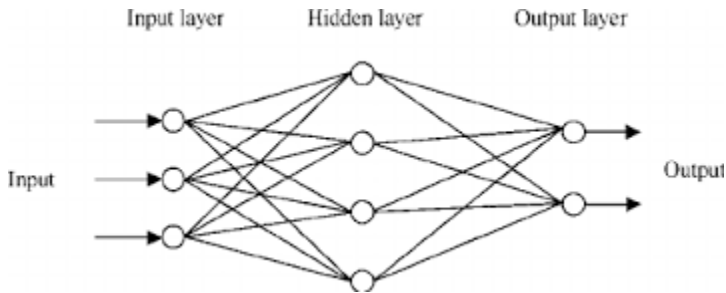


Perceptron (MLP) neural network with a window of time delayed inputs introduces another assumption by externally addressing the temporal connection of the inputs via the addressing of an internal state (Giles et al., 2001). None of these approaches is superior to another in all cases (Horne and Giles, 1995). However, in our perspective, an additional dampened feedback that possesses the characteristics of a dynamic memory will improve the performance of both approaches. It can convert a non-repeated network into an Elman network to remember the previous state. This done by adding a new set of inputs that fully connected between the repeated links and the hidden-layer outputs (but delayed by one time unit). Two forms of delayed neural networks are proposed in this study, as follows:

1. **Input Delayed Neural Network**: Time-Delay Neural Network (TDNN) or Time Series Time-Delay Neural Network, also known as the neural network finite impulse response architecture, has been used successfully in a number of practical algorithms including speech recognition and time-series prediction. A TDNN (Figure 6) is similar to a multi-layer perception in that all connections feed forward. With the TDNN, The difference is that the inputs to any node $i$ could involve the outputs of earlier nodes not only within the present time frame but during some numbers $d$ of previous time steps (t -1, t - 2… t - d) as well. This is generally implemented using tap-delay lines. The actual problem of the general TDNN topology was

**Figure 6. Input Delay Neural Network**
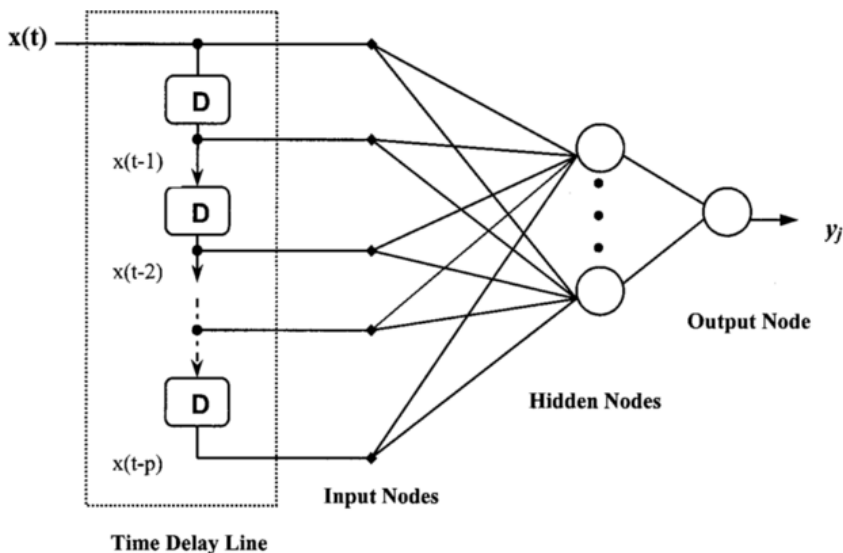
the class of infrastructures which had time delays specifically on the input units. The IDNN has a single output unit and d-1 delay on its input that computes a function of the most recent d inputs including the current input:

$$y(n) = f\left[\sum_k w_k \cdot x(n-k+1)\right] \tag{7}$$

where $f$ is a sigmoid function and the weights are corrected by the Back-Propagation algorithm (BP).

2. **Elman Networks:** They are a form of recurrent Neural Networks (Figure 7) which have connections from their hidden layer back to a special copy layer. That means the function learned by the network could be used the present inputs plus a case of the previous state(s) and outputs of the network. As shown below, a new set of inputs connected completely by delay element to the hidden layer outputs. More precisely, the Elman network is a finite-state machine that predicts what state in memory (i.e., what is relevant). The special copied layer is treated as just another bunch of inputs and hence standard back-propagation learning algorithms could be used:

$$y(n) = f\left[\sum_k w_k \cdot x_k(n) + \sum_m w_m \cdot y_m(n-1)\right] \tag{8}$$

## 5. STATIONARY PROCESS

A common assumption in many time series techniques is that the data are stationary, but in reality, data points are often non-stationary. A stationary process has the firm that the mean, variance and entire structure do not vary over time. Non-stationary behaviours can be trends, cycles, random walks

**Figure 7. Elman Neural Network**
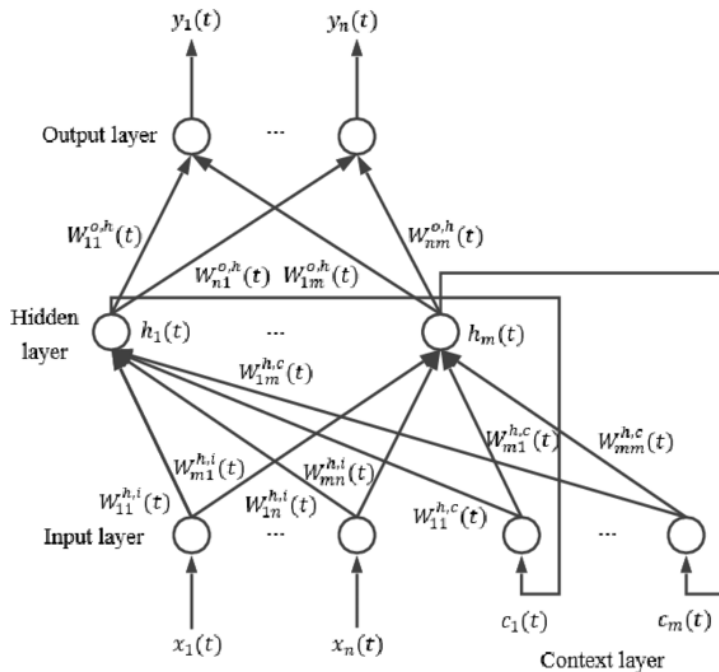
or combinations of the three processes. Using non-stationary time-series information in business models produces non-relevant and false results and presents poor perception and forecasting. If the time series is not stationary, it often transforms it to stationary with one of the following techniques: a random-walk can be transformed to a stationary process by differentiating and then the process becomes difference-stationary. That is, given the series $Y_t$, it create the new series $Zi = Y_i - Y_{i-1}$, the drawback of differentiating is that the operation loses one observation each time the difference is applied. Although it can differ the data more than once, first order differentiating is usually sufficient. A non-stationary process with a deterministic run becomes stationary after striping the run, or deterring the end. It can produce some type of curve to the data and then model the backwards. For example, $Y_t = \alpha + \beta t + \epsilon t$ is transformed into a stationary process by subtracting $\beta t$: $Y_t - \beta t = \alpha + \epsilon t$. Because the purpose of the training is to strip long-term orientation, a simple firm, such as a straight line, is typically used. No process is lost when deter-ending is used to move from a non-stationary process to a stationary one.

For variable variance, having the logarithm or square root of the series may organize the variance. For negative data, it can add a suitable constant to make all data positive before applying the transformation. This constant can then be removed from the model to obtain predicted (i.e., the fitted) values and forecasts for future points. The approach used to deal with the non-stationary of the signal in this work and to differ original data followed by a logarithm transformation and build models based on a short time period only. This is an intuitively appealing approach because one would expect that any inefficiency found in the market value would typically not last for a long time.

## 6. RESULTS AND DISCUSSION

In order to show the effectiveness of the symbolic encoding and recurrent neural network, the following five systems have been investigated:

1.  The system as presented in Figure 3 with SOM and Elman network. The configurations of SOM and Elman network are described in Table 2 and Table 3 respectively.
2.  The system above but without the symbolic encoding, i.e. the pre-processed data is entered directly into the recurrent neural network without the SOM stage.
3.  The system with the recurrent network replaced by a standard MLP network. The configuration of MLP is described in Table 1.
4.  Standard MLP network only.
5.  Simple linear regression model.

Linear regression can be easily implemented and converged quickly[1]. The traditional predictive regression methodology implies stock returns on lagged predictors:

**Table 1. Configuration of IDNN**

| Parameters | Value |
|---|---|
| Number of hidden nodes | 5 |
| Learning rule | BP |
| Learning rate | 0.01 |
| Momentum constant | 0.9 |
| Transfer function of hidden layer | Tansig |
| Transfer function of output layer | Purelin |

**Table 2. Configuration of SOM**

| Parameters | Value |
|---|---|
| Dimension | 1 |
| Number of nodes | 2 |
| Learning rule | Kohonen |
| Learning rate | 0.01 |
| Epochs | 50 |

**Table 3. Configuration of Elman Network**

| Parameters | Value |
|---|---|
| Number of hidden nodes | 10 |
| Learning rate | 0.01 |
| Momentum constant | 0.9 |
| Transfer function of recurrent layer | Tansig |
| Transfer function of output layer | Purelin |

$$rt+1 = \alpha + \beta xt + \epsilon\ t+1 \tag{9}$$

But even test it against training data it can only predict direction of changing 75% of time. It gets $R^2 = 0.1733$, the F statistic $= 0.6290$ and p$= 0.6511$ is a value for the full model, and an estimate of the error variance $= 22.4130$. Every number is suggesting that linear regression model failed to capture the nature of the problem which has been pointed out earlier in this manuscript[2]. Algorithm of SOM, as shown in Figure 8, is quite simple that did a very good performance in terms of classification. In this particular case, only two neurons are used (one means stock index will go up and another means it will go down). Table 4 shows the Error test rate at each model.

Two types of recurrent neural networks have been used here, both Elman networks and MLP. For Elman networks, after less than 100 epochs training, it can perfectly simulate training data almost all the time (Figure 9). That is not surprising since it can approximate any function with a finite number of discontinuities and with an arbitrary accuracy if the hidden layer have enough neurons. When presented with testing data to simulate a real prediction or forecasting case, the best performance was obtained with an embedding dimension of 8 and 10 hidden neurons where the error rate was 58.0%, as shown in Figure 10. A t-test indicates that this result is significantly different from the null hypothesis corresponding to a random walk at P=5.3854^-3. Sensitive analysis was done for both embedding dimension number and hidden neuron number as shown in Figure 11 and 12.

**Table 4. The results of the systems**

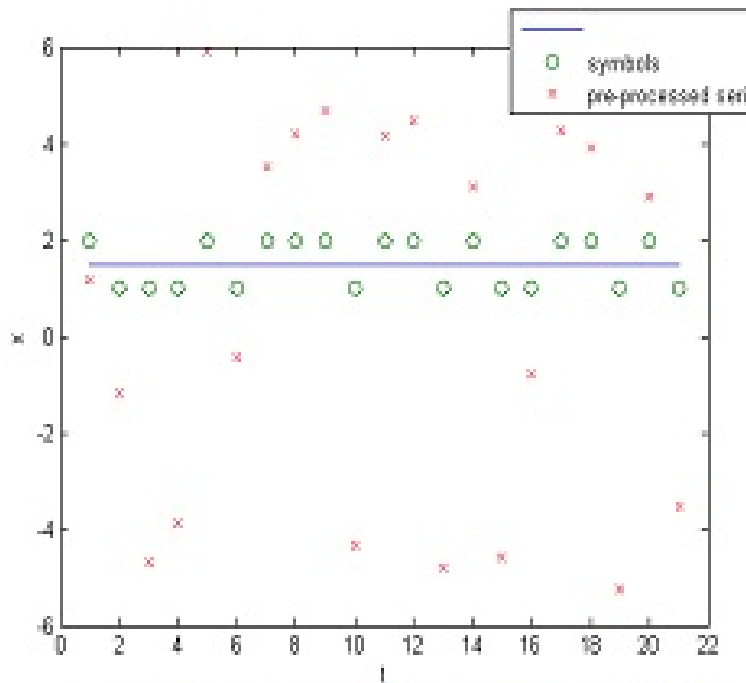| Model | Test Error % |
|---|---|
| SOM + Elman | 0.525 |
| Elman | 0.530 |
| SOM + MLP | 0.451 |
| MLP | 0.445 |

**Figure 8. Symbolic Encoding by SOM**
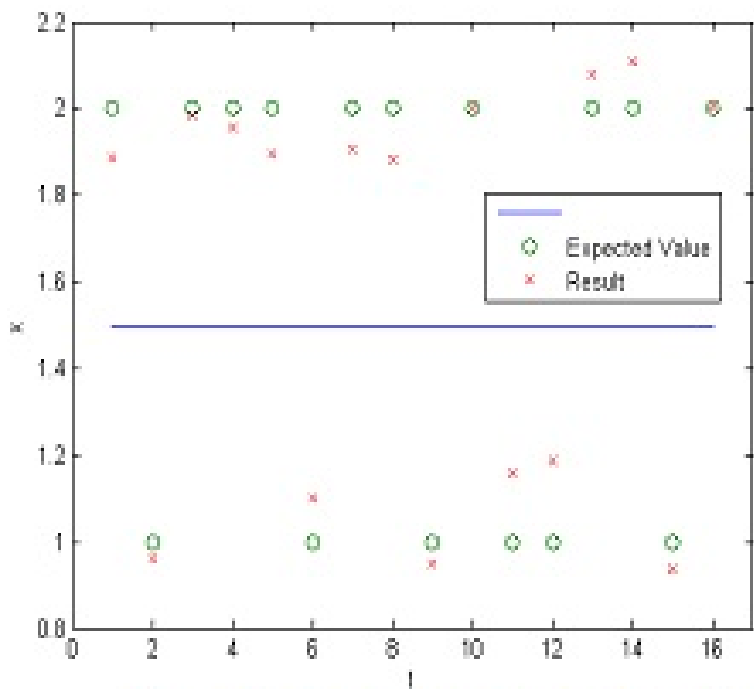


**Figure 9. Simulation using Training Data**
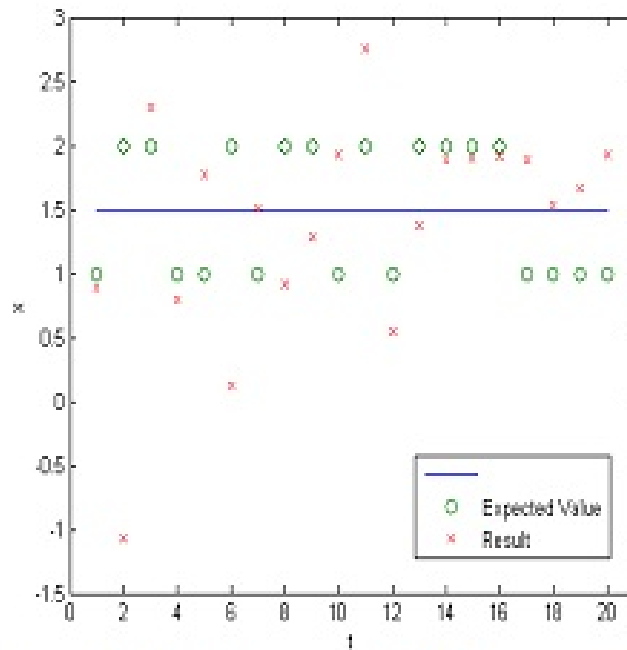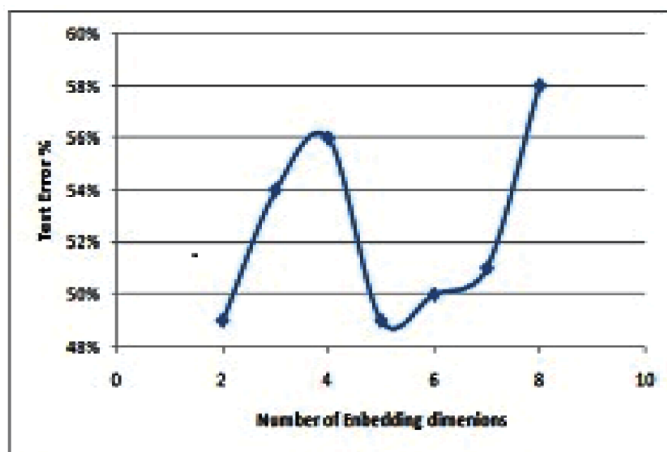
**Figure 10. Simulation using Test Data**



**Figure 11. Sensitive Analysis of Embedding Dimension**



## 7. CONCLUSION

Forecasting stock market value or stock index is a substantial financial matter that has relevant researchers' attention for several years. There exist a vast number of articles addressing the predictabilities of stock market return, and many of them are claiming that the simulation can benefit greatly from quantization of the original data into symbolic streams. And also there is no conclusion whether the use of a recurrent neural network, including the temporal relationship of the series explicitly in the model, is significant for this type of study. We did a comparison of five

**Figure 12. Sensitive Analysis of Number of Hidden Neurons**
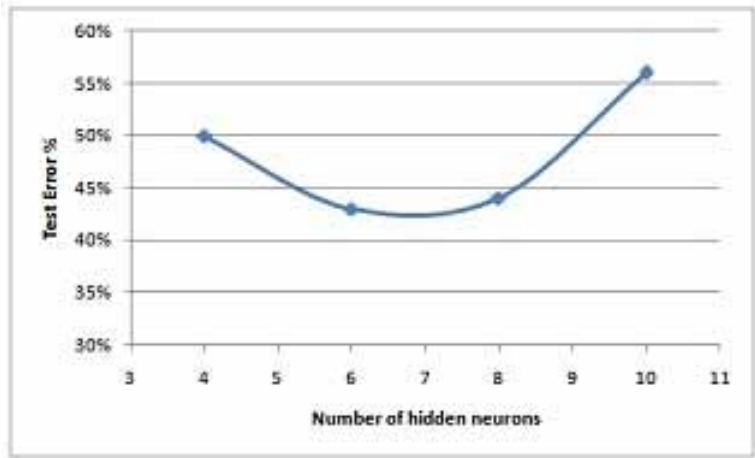


different models: Elman network, MLP network, Elman network with SOM filter, MLP with SOM filter and simple linear regression. The result of the simulation shows that SOM can greatly improve the convergence of the neuron networks; whereas Elman network do a better performance to capture the temporal pattern of the symbolic streams generated by SOM. Simple linear regression is almost unusable in this case. Some sensitivity study was done with respect to neural network parameters. A sensitivity study of SOM parameters, which could be significant here, is left to future work.

As future directions and implications, it is clear that our research shows some promising results but sensitive study on numbers of dimensions and neurons would have to be conducted.

## FUNDING AGENCY

## REFERENCES

Ahmed, E., & Hokey, M. (2009). The artificial neural network for forecasting foreign exchange rates. Proceedings of Int. J. Services and Operations Management, 5(6).

Amin, M., Moein, M., & Morteza, E. (2016). Stock Market Index Prediction Using Artificial Neural Network. *Journal of Economics, Finance and Administrative Science*, *21*(41), 89–93.

Anthony, P. (2013). A review of stock market prediction with Artificial Neural Network (ANN). *IEEE International Conference on Control System, Computing and Engineering*.

Atsalakis, G., & Valavanis, K. (2009). Forecasting stock market short-term trends using a neuro-fuzzy based methodology. *Expert Systems with Applications*, *36*, 10696–10707.

Burton, M. (2003). *The Efficient Market Hypothesis and Its Critics*. CEPS Working Paper, No. 91, Princeton University, Department of Economics, Centre for Economic Policy Studies.

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

David, E. (2006). Neural Network Based Stock Market Return Forecasting Using Data Mining for Variable Reduction. University of Missouri.

Fama, E. (1965). The Behaviour of Stock Market Prices. *The Journal of Business*, *38*(1), 34–105. doi:10.1086/294743

Giles, C., Steve, L., & Ah, T. (2001). Noisy Time Series Prediction using Recurrent Neural Networks and Grammatical Inference. *Machine Learning*, *44*(1), 161–183.

Guresen, E., Kayakutlu, G., & Daim, T. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*, 10389–10397.

Hedayati, A., Hedayati, M., & Esfandyari, M. (2016). Stock Market Index Prediction Using Artificial Neural Network. *Journal of Economics, Finance and Administrative Science*, *21*, 89–93.

Horne, G., & Giles, L. (1995). An experimental comparison of recurrent neural networks. MIT Press. *Advances in Neural Information Processing Systems*, 697–704.

Joseph, D., Roger, A., & Charles, T. (2012). *Forecasting stock returns: What signals matter, and what do they say now?* Vanguard Research.

Karlsson, S., & Nordberg, M. (2015). *Stock market index prediction using artificial neural networks trained on foreign markets*. Degree Project in Computer Science, DD143X.

Kuo, R., Chen, C., & Hwang, Y. (2001). An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy Sets and Systems*, *118*, 21–45.

La, R., Lakonishok, J., Shliefer, A., & Vishny, R. (1997). Good news for value stocks: Further evidence on market efficiency. *The Journal of Finance*, *52*, 859–874.

Laurent, C. (1979). Improving the efficiency and effectiveness of financial ratio analysis. *Journal of Business Finance & Accounting*, *6*(3), 401–413. doi:10.1111/j.1468-5957.1979.tb01099.x

Moghaddam, A., Moghaddam, M., & Esfandyari, M. (2016). Stock market index prediction using artificial neural network. *Journal of Economics, Finance and Administrative Science*, *21*(41), 89–93. doi.org/10.1016/j.jefas.2016.07.002

Pavlidis, G., Plagianakos, P., Tasoulis, K., & Vrahatis, N. (2006). Financial forecasting through unsupervised clustering and evolutionary trained neural networks. *Operational Research Journal*, *6*(2), 103–127.

Peter, T., Christian, S., & Georg, D. (2000). *Temporal pattern recognition in noisy non-stationary time series based on quantization into symbolic streams. Lessons learned from financial volatility trading. Report Series SFB. Adaptive Information Systems and Modelling in Economics and Management Science*. University of Economics and Business.

Qiu, W., Liu, X., & Wang, L. (2012). Forecasting shanghai composite index based on fuzzy time series and improved C-fuzzy decision trees. *Expert Systems with Applications*, *39*, 7680–7689.

Sakarya, S., Yavuz, M., Karaqglan, A., & Özdemir, N. (2015). Stock market index prediction with neural network during financial crises: A review on BIST-100. *Financial Risk and Management Reviews*, *1*(2), 53–67.

Shen, P. (2000). The P/E Ratio and Stock Market Performance. *Economic Review (Federal Reserve Bank of Atlanta)*.

Wang, L., & Wang, Q. (2011). Stock Market Prediction Using Artificial Neural Networks Based on HLP. *3rd International Conference on Intelligent Human-Machine Systems and Cybernetics*. doi:10.1109/IHMSC.2011.34

## ENDNOTES

[1]     Alternatives to predictive regressions based on Bayesian methods, latent variables, analyst forecasts.
[2]     To be more rigorous, the estimated coefficients of the regression should be indexed as they change with the expanding sample.

*Falah Al-Akashi received his PhD in Computer Science from the University of Ottawa, Canada in 2014. In 2010-2014 he was worked as the Teaching and Research Associative by the Faculty of Engineering at the University of Ottawa and is currently he is a Doctor Professor at the University of Kufa. He is a keen interest in Neural Networks, Information Retrieval, Data Mining, Web Technology, Multi-agent Models, Search Engines, and Big or Ocean Data on which he publishes many research papers.*