EFFICIENT NEURAL STOCK MARKET PREDICTION SYSTEM

Arosha Senanayake, J. Janaththanan
Faculty of Science, University of Peradeniya, Peradeniya, Sri Lanka

ABSTRACT

Colombo stock market trading varies depending on the quantitative and qualitative factors. The intelligent system introduced in this research predicts the stock market trading in an efficient manner. The analysis of quantitative factors is primarily based on how effective data mining does from the parameters extracted in the Colombo stock market exchange. Initially, it was done based on the extraction of noisy data taking into consideration the principal component analysis and secondly, the feed-forward neural network is applied to optimize the same. The analysis of qualitative factors is based on how frequently the stock market trading is disturbed by the political situation in the country. Fuzzy logic approach is proposed to decide the degree of influence of political situation for the prediction. Finally, another feed-forward neural network is introduced to determine whether the stock market is fully, partially or not based on the quantitative factors. Based on the degree of influence obtained from the fuzzy analysis, the output is projected. The results show that the system can be effectively applied for Colombo Stock Market Exchange with less than 20% error.

1.0 INTRODUCTION

The Stock market is always one of the most popular investments due to its high profit. However higher profit tends to higher risk too. Thus, various research works intended to develop models in order to provide the investors an optimum prediction. Among the traditional research, time series analysis techniques and multiple regression models were used. Recently due to the computational speed, Artificial Neural Networks (ANN) has been also used in this area. Through various models have been proposed, they only concentrated quantitative factors. However, in developing countries, like Sri Lanka, sometime non quantitative factors are more important than qualitative factors. Therefore, proposed intelligent stock market prediction system intends the inclusion of both factors. Therefore, proposed intelligent stock market prediction system intends the inclusion of both factors such that right decision

In this project I consider only the quantitative factors. Apply the Neural Network concept to the quantitative factors this is the first phase of the whole project. In future remaining phase will be completed.
1.1 Intelligent Stock Market Prediction

Related factors collection for the stock market environment.

Quantitative factors

Artificial Neural Network (Prediction 1)

Qualitative factors

Events effect

Time effect

Artificial Neural Network (Prediction 2)

Artificial Neural Network (Prediction 3)

Fig 1: Structure of Stock Market Prediction
1.1.1 Factors Collection
In order to make right decision, collecting the effective information regarding the predicted object is crucial. The assumption that makes, is collected factors are good enough to support the prediction model.

1.1.2 Quantitative factors
The factors that can be considered as qualitative factors are volume, average volume, rate of volume, index in open, index in close, index fluctuation, rate of index change, financial buying, financial selling, remaining quota with financing, remaining quota with stock. Turn over rate, other related factors in the Stock market. These are the inputs to the ANN of quantitative factors. The output to the same specifies stock tendency performance.

1.1.3 Qualitative factors
These factors must be collected from the society in consideration. Depending on the factors, it is necessary to formulate a questionnaire. Answers to the questionnaire will get from the experts and determine the mean value of factors consider. In this way, several different questionnaires will be formulated and forced to learn the proposed ANN. Therefore, the output of this ANN is the effect of qualitative factors to the stock market.

1.1.4 Decision integration
From the above proposed two ANNs can be obtained general stock market tendency and special factors effect which are based on the quantitative and qualitative models respectively. The overall results can determine the integration of these outputs together with time effect through a third ANN as shown in figure 1.

1.2 The Colombo Stock Exchange (CSE)
The Colombo Stock exchange was established in 1984. It is the central, and only authorized market for shares listed for public transaction. These securities are mainly the shares of public listed companies. There is also a limited amount of trading in treasury bills and convertible debentures.

The CSE is not mushroom development of a market economy. It traces its history more than hundred years to 1896, when the Colombo share market was established under the administration of the Colombo share broker’s association, later called the Colombo brokers association. The CSE is now situated at World Trade Center, Colombo 01.
Like any other exchange the CSE does not itself buy, sell or set price of the share traded on its trading floor. The prices of these are strictly determined by supply and demand. It provides all facilities for convenient bidding by brokers, similar to a public auction, publishes a handbook of listed companies and other material useful to the investor. The CSE’s Central Depository System (CDS) is a fully computerized system of recording and processing all share transactions. The CDS has a comprehensive database of shares listed on the CSE, and a market information system making it one of the most modern integrated systems of any stock exchange worldwide which assists in efficient post-trade operation.

1.2.1 Share price indices
Share price index is an important datum to know about current situation of the CSE. The CSE calculates share price index for each and every sector called sectorial index. It also calculates the All Share Price Index (ASPI) and Milanka Price Index, which are main share indices.

1.2.2 All Share Price Index (ASPI)
This index reflects the share price fluctuations of all companies \((n = 240)\) in the stock market. The base year for ASPI is 1985 and base index was set at 100 points. The CSE computes ASPI using the following formula.

\[
\text{ASPI} = \frac{\sum_{i=1}^{n} \left( \frac{P_{i}}{P_{0}} \times \frac{Q_{i}}{Q_{0}} \right) \times 100}{n}
\]

Where
- \(P_{i}\) - Present price of the \(i^{th}\) company.
- \(Q_{i}\) - Total issued quantity of Shares of the \(i^{th}\) company.
- \(P_{0}\) - Base year price of a share of the \(i^{th}\) company.
- \(Q_{0}\) - Base year quantity of a share of the \(i^{th}\) company.
1.2.3 Sensitive Share Prize Index (SSPI) / Milanka Price Index (MPI)

The CSE introduced the Milanka Price Index (MPI) with effect from 4th January 1999. The base index was set at 1000 points as at 31st December 1998. The MPI replaced the Sensitive Price Index (SPI). The introduction of the MPI was timely for several reasons. Some of the companies included in the SPI did not continue to meet the evaluation criteria since the SPI did not continue to meet the evaluation criteria since the SPI was last revised in August 1994. Unlike the SPI, the MPI is not a blue chip index and the criteria taken in to its constructions are size and liquidity. The MPI is comprised of 25 companies representing 7 sectors. The index represents over 10% of the listed companies and accounts for over 50% of market capitalization of the CSE.

1.3 Quantitative Factors in CSE

The factors that can be considered as qualitative factors in CES are Turnover, Trades, Shares Traded, Companies Traded, Companies Listed and Market Capitalization. These are the inputs to the ANN of quantitative factors. The outputs to the System are ASPI and SSPI.

1.3.1 Turnover
Daily Total turnover in Colombo Stock Exchange, including the foreign and local

1.3.2 Trades
Number of trades done by Colombo stock exchange daily, which includes selling and buying

1.3.3 Shares Traded
Total number of shares traded per day in Colombo Stock Exchange

1.3.4 Companies Traded
How many companies trade their shares on particular day

1.3.5 Companies Listed
How many companies listed in the Colombo stock exchange for trades their shares.

1.3.6 Market Capitalization
Market Capitalization = \( \sum \) Price of Share * Quantity of shares issued

1.4 Qualitative Factors in CSE

The Share Price Indices mainly depends on the following Qualitative factors.

1. Socio Economic conditions of the country.
2. Political situations.
3. Performance of listed companies.
4. Activities of foreign investors.
5. International scenario.

Fluctuations in share price of the companies are mainly depending on the above factors. These factors are discussed briefly as follows.

1.4.1 Socio Economic conditions of the country

Economic growth of country plays an important role in CSE. Insufficient infrastructure facilities have been one of the factors that have affected the competitiveness of Srilankan products and constrained faster economic expansion in Srilanka. The conflict in North_East has curtailed the development of key sectors due to resource constraints. Inflation (depreciation of money) and weather conditions, which determine the Socio Economic Conditions, do have a direct impact on price fluctuations of shares.

1.4.2 Political Situations

The policy of ruling government makes a drastic change in Stock Market activity. Privatization of the sectors fiscal policy (public revenue), monetary policy and economy policy of the government play an important role in Stock Market activities.

1.4.3 Performance of listed Companies (16 sectors)

The performance of listed companies depends on many aspects such as values of shares traded, frequency of shares traded, and volume of shares traded, Market Capitalization, movement in sector indices etc.

1.4.4 Activities of Foreign investors

Foreign investor interest in the Colombo Stock Market has been inextricably linked to the Indian Market. The indirect investments of foreign investors through Colombo Stock Exchange make substantial price movements of the shares. Sri Lanka Market is over dependent on foreign investment.
1.4.5 International Scenario
The Colombo Stock Exchange is very much dependent on international scenario because it is a member of the FIBV (International Federation of Stock Exchanges). For example, the Colombo stock market commenced the year 1998 amidst the aftermath of the East Asian Contagion, and plunged almost immediately into a second dilemma precipitated by the nuclear explosions by India and Pakistan. As because of those reasons All Share Price index declined by 15% and the Sensitive Price Index by 14% in 1998. In 2001 October also ASPI and SPI are affected because of attack of American Buildings.

2.0 DESIGN ANN FOR STOCK MARKET PREDICTION

The specification and design of an ANN application should aim to produce the best system and overall performance. Much of the work is done in the initial data preprocessing and feature extraction as possible to reduce the task of the network. A number of different types of ANN can be used in the same application.

ANN design involves five main tasks:
1) Data collection
2) Selecting Inputs and Outputs
3) Raw data preprocessing
4) Selection of an ANN type and topology
5) ANN training, testing and validation

Firstly the problem is specified and an ANN model is chosen. Next suitable data are collected and preprocessed before the features are extracted to make the input/output vector pair sets. These vector sets are then used to train and test the ANN. If the results are satisfactory the ANN is finally validated and the design is complete. However, more likely than the results will not be satisfactory on the first design pass, In that case it is necessary to repeat one or more of the process steps repeatedly until the problem is solved. In some cases the problem may have to be re-specified or even abandoned if no adequate solution can be achieved (Figure 2).
Fig 2: ANN Design Procedure for Stock market prediction
2.1 Data collection
All necessary data are collected from Colombo Stock Exchange. They gave daily based data for ALSH, SSPI, Daily Turnover, Number of Shares Traded, Number of Trades, Market Capitalization, Turnover ratio and Sectorial indices. Also gave monthly based data for Domestic and Foreign Total Turnover, Domestic and Foreign Trades, Domestic and Foreign shares traded, Companies Traded, Companies Listed, All Share Price Index, Sensitive / Milanka (since 1999), Market Capitalization, Market PER, Dividend Yield and Market PBV.

2.2 Selecting Inputs and Outputs
The ability of neural networks to discover nonlinear relationships in input data makes them ideal for modeling nonlinear dynamic systems such as the stock market. Various neural network configurations have been developed to model the stock market. Often these networks use raw data and derived data from technical and fundamental analysis. This section will overview the use of neural networks in financial markets including a discussion of their inputs, outputs, and network organization.

One of the most important factors in constructing a neural network is deciding on what the network will learn. The challenge is determining which indicators and input data will be used, and gathering enough training data to train the system appropriately. The input data consider here is daily change. A neural network must be trained with large amount of pattern. Each example pattern consist set of inputs and outputs. These patterns are collected from Colombo Stock Exchange.

Inputs from CSE → Number of Shares Traded (X1), Number of Trades (X2), Market Capitalization (X3)

Outputs from CSE → All Share Price Index (Y1), Milanka Price Index (Y2)

2.3 Raw data preprocessing

2.3.1 Modify the Inputs and Outputs
Quantitative factors are modified for the Stock market Prediction. New Inputs and outputs are formulated by following steps.

Step1
Calculate the different between Today’s values and Yesterday’s values.

\[
\begin{align*}
  dX_1 &= X_1 - X_{1,-1} \\
  dX_2 &= X_2 - X_{2,-1} \\
  dX_3 &= X_3 - X_{3,-1} \\
  dY_1 &= Y_1 - Y_{1,-1} \\
  dY_2 &= Y_2 - Y_{2,-1}
\end{align*}
\]

Step2
Assign new values to the inputs of the training data. These values in the range of -1.0 to 1.0 are called normalized values of input.

Spanned and normalize the inputs for each output.

<table>
<thead>
<tr>
<th>Input</th>
<th>New Input Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>1 ←→ 0.5</td>
</tr>
<tr>
<td>+</td>
<td>0.5 ←→ 0</td>
</tr>
<tr>
<td>-</td>
<td>0 ←→ -0.5</td>
</tr>
<tr>
<td>-</td>
<td>-0.5 ←→ -1.0</td>
</tr>
</tbody>
</table>
Case 1\[ dy_{i} \Rightarrow + \quad dx_{i} \Rightarrow - \]
\[
\begin{align*}
  dy_{i}/dx_{i} & \quad \text{New Weight Range} \\
  10.0 - 1.0 & \quad 1.0 - 0.9 \\
  1.0 - 0.1 & \quad 0.9 - 0.7 \\
  0.1 - 0.0 & \quad 0.7 - 0.5
\end{align*}
\]

Case 2\[ dy_{i} \Rightarrow + \quad dx_{i} \Rightarrow + \]
\[
\begin{align*}
  dy_{i}/dx_{i} & \quad \text{New Weight Range} \\
  10.0 - 1.0 & \quad 0.4 - 0.5 \\
  1.0 - 0.1 & \quad 0.4 - 0.2 \\
  0.1 - 0.0 & \quad 0.2 - 0.0
\end{align*}
\]

Case 3\[ dy_{i} \Rightarrow - \quad dx_{i} \Rightarrow - \]
\[
\begin{align*}
  dy_{i}/dx_{i} & \quad \text{New Weight Range} \\
  10.0 - 1.0 & \quad -0.4 - -0.5 \\
  1.0 - 0.1 & \quad -0.4 - -0.2 \\
  0.1 - 0.0 & \quad -0.2 - 0.0
\end{align*}
\]

Case 4\[ dy_{i} \Rightarrow - \quad dx_{i} \Rightarrow + \]
\[
\begin{align*}
  dy_{i}/dx_{i} & \quad \text{New Weight Range} \\
  10.0 - 1.0 & \quad -0.9 - -1.0 \\
  1.0 - 0.1 & \quad -0.7 - -0.9 \\
  0.1 - 0.0 & \quad -0.5 - -0.7
\end{align*}
\]

**Outputs**
Assign new values to the output of the training data. It is important that the target values (desired response) be chosen within the range of sigmoid activation function. These values in the range of -1.0 to 1.0 are called normalized value of output.

New Value = \( dy_{i} / \text{maximum value of last ten values} \)

\[ = dy_{i} / \max(dy_{1}, dy_{i-1}, dy_{i-2}, \ldots \ldots \ldots dy_{i-28}, dy_{i-29}, dy_{i-30}) \]

2.3.2 Training Data and Test Data
The patterns are divided into two Training Set and Testing set. Comparatively training set is larger than the testing set. Training set is used to train the network. Testing set is used to test the performance of the network.

Patterni\[ \longrightarrow \text{(Input1, Input2, Input3, Input4, Input5, Input6, Output1, Output2)} \]

3.0 SELECTION OF AN ANN TYPE AND TOPOLOGY

3.1 Artificial Neural Network Type
The most common network architecture used is the backpropagation network. However, stock market prediction networks have also been implemented using genetic algorithms, recurrent networks, and modular networks.
Backpropagation networks are the most commonly used network because they offer good generalization abilities and are relatively straightforward to implement. Although it may be difficult to determine the optimal network configuration and network parameters, these networks offer very good performance when trained appropriately.
3.2 Artificial Neural Network Topology

3.2.1 Optimal Number of Hidden Neurons

To find the optimal number of neurons in the hidden layer, network architecture is changed for each training and observe the mean square error on the network. For the analysis (6-8-2) and (6-6-2) architectures are selected. Both are trained with 1750 patterns and 0.2 learning rate. Training outputs are shown below. When comparing two outputs, MSE of (8-8-2) architecture is higher than that of architecture (6-6-2). Second one is more suitable for training because it’s minimize error with minimum number of neurons. Smallest number of hidden neurons that increase the performance of the network. So optimal number of hidden neuron is six.

**Architecture (6-8-2)**

<table>
<thead>
<tr>
<th>Epoch</th>
<th>SSE</th>
<th>MSE</th>
<th>SS/1000 units</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>1031.92</td>
<td>0.5897</td>
<td>515.99615</td>
</tr>
<tr>
<td>9000</td>
<td>553.28</td>
<td>0.31616</td>
<td>276.63840</td>
</tr>
<tr>
<td>8000</td>
<td>543.49</td>
<td>0.3057</td>
<td>271.74545</td>
</tr>
<tr>
<td>7000</td>
<td>540.63</td>
<td>0.3089</td>
<td>270.31641</td>
</tr>
<tr>
<td>6000</td>
<td>539.58</td>
<td>0.3063</td>
<td>269.79263</td>
</tr>
<tr>
<td>5000</td>
<td>540.18</td>
<td>0.3068</td>
<td>270.09232</td>
</tr>
<tr>
<td>4000</td>
<td>540.44</td>
<td>0.3082</td>
<td>270.22177</td>
</tr>
<tr>
<td>3000</td>
<td>539.48</td>
<td>0.3082</td>
<td>269.74295</td>
</tr>
<tr>
<td>2000</td>
<td>538.57</td>
<td>0.3076</td>
<td>269.17853</td>
</tr>
<tr>
<td>1000</td>
<td>538.85</td>
<td>0.3079</td>
<td>269.29416</td>
</tr>
</tbody>
</table>

Input layer: 6 Neurons
Output layer: 2 Neurons
Hidden layer: one hidden layer with 8 Neurons
Number of Training patterns: 1750 pattern
Training Algorithm: Std. Backpropagation
Number of epochs: 10000
Learning Rate: 0.2

**Architecture (6-6-2)**

<table>
<thead>
<tr>
<th>Epoch</th>
<th>SSE</th>
<th>MSE</th>
<th>SS/1000 units</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>1130.76</td>
<td>0.6465</td>
<td>565.37869</td>
</tr>
<tr>
<td>9000</td>
<td>536.30</td>
<td>0.3046</td>
<td>268.15454</td>
</tr>
<tr>
<td>8000</td>
<td>524.60</td>
<td>0.2978</td>
<td>262.30365</td>
</tr>
<tr>
<td>7000</td>
<td>523.10</td>
<td>0.2989</td>
<td>261.55200</td>
</tr>
<tr>
<td>6000</td>
<td>522.41</td>
<td>0.2985</td>
<td>261.20667</td>
</tr>
<tr>
<td>5000</td>
<td>521.31</td>
<td>0.2980</td>
<td>259.75390</td>
</tr>
<tr>
<td>4000</td>
<td>521.47</td>
<td>0.2979</td>
<td>259.73950</td>
</tr>
<tr>
<td>3000</td>
<td>519.92</td>
<td>0.2971</td>
<td>259.96231</td>
</tr>
<tr>
<td>2000</td>
<td>520.28</td>
<td>0.2973</td>
<td>260.14233</td>
</tr>
<tr>
<td>1000</td>
<td>521.33</td>
<td>0.2979</td>
<td>260.66727</td>
</tr>
<tr>
<td>1</td>
<td>519.62</td>
<td>0.2969</td>
<td>259.81082</td>
</tr>
</tbody>
</table>

Input layer: 6 Neurons
Output layer: 2 Neurons
Hidden layer: one hidden layer with 6 Neurons
Number of Training patterns: 1750 pattern
Training Algorithm: Std. Backpropagation
Number of epochs: 10000
Learning Rate: 0.2
3.2.2 Optimal Learning Rate

To find the optimal learning-rate parameter ($\eta$), network architecture is trained with various learning-rate parameter. It varies from 0 to 1.0. Network architecture (6-6-2) is trained with 1750 patterns. Learning rate parameter starts from 0.2 and each time it increased by 0.2 up to 1.0. MSE of three networks is observed, training network with learning-rate 0.2 is converged more than network with learning-rate 0.4, as well as network with learning-rate 0.8. Also network with learning-rate 0.4 is converged little bit more than network with learning-rate 0.8. Learning rate 0.2 is more suitable for the network. Increasing the learning-rate lead to converge the network quickly but it is not stable.

<table>
<thead>
<tr>
<th>Epoc</th>
<th>SSE</th>
<th>MSE</th>
<th>SSeo-units</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>1130.75378</td>
<td>0.64651</td>
<td>565.37689</td>
</tr>
<tr>
<td>9000</td>
<td>536.30908</td>
<td>0.30646</td>
<td>268.15454</td>
</tr>
<tr>
<td>8000</td>
<td>524.00730</td>
<td>0.29978</td>
<td>262.30365</td>
</tr>
<tr>
<td>7000</td>
<td>523.10400</td>
<td>0.29892</td>
<td>261.20667</td>
</tr>
<tr>
<td>6000</td>
<td>522.41333</td>
<td>0.29852</td>
<td>261.05200</td>
</tr>
<tr>
<td>5000</td>
<td>521.51880</td>
<td>0.29801</td>
<td>260.75940</td>
</tr>
<tr>
<td>4000</td>
<td>521.47180</td>
<td>0.29798</td>
<td>260.73590</td>
</tr>
<tr>
<td>3000</td>
<td>519.92462</td>
<td>0.29710</td>
<td>259.96231</td>
</tr>
<tr>
<td>2000</td>
<td>520.28467</td>
<td>0.29731</td>
<td>260.14233</td>
</tr>
<tr>
<td>1000</td>
<td>521.33453</td>
<td>0.29791</td>
<td>260.66727</td>
</tr>
<tr>
<td>1</td>
<td>519.62164</td>
<td>0.29693</td>
<td>259.81082</td>
</tr>
</tbody>
</table>

Input layer: 6 Neurons
Output layer: 2 Neurons
Hidden layer: one hidden layer with 6 Neurons
Number of Training patterns: 1750 pattern
Training Algorithm: Std. Backpropagation
Number of epochs: 10000
Learning Rate: 0.2
The learning should be stopped in the minimum of the validation set error. At this point the net generalizes best. When learning is not stopped, overtraining occurs and the performance of the net on the whole data decreases, despite the fact that the error on the training data still gets smaller. After finishing the learning phase, the net should be finally checked with the test set. SNNS performs one validation cycle every n training cycle. Just like training, validation is controlled from the control panel of the simulator.

3.3 The Neural Network Architecture

The NSSP-system was a backpropagation network designed using a standard backpropagation algorithm. The algorithm allowed the automated design of the neural network, and determined that the optimal network configuration was one hidden layer with 6 nodes. Optimal learning-rate of Network Architecture is 0.2 with 10000 cycles. After execute the cycle training examples should be shuffle. The initial weights and the threshold values are chose with random values. Network is trained with 1750 patterns using SNNS simulator.
Prediction system is based on the backpropagation network [6, 6, 2]. Figure shows the architectural graph of multilayer perceptron with a hidden layer and an output layer. The network shown here is fully connected. This means that a neuron in any layer of the network is connected to all the nodes/neurons in the previous layer.

Two kinds of signals are identified forward signal and backward signal. A forward signal that comes in at the input end of the network, propagates forward (neuron by neuron) through the network, and emerges at the output end of the network as an output signal. Signal flow through the network progresses in a forward direction, from left to right and on a layer-by-layer basis. A backward signal (Error signal) originates at an output neuron of the network, and propagates backward (layer by layer) through the network.

### 3.4 Implementation

This system was developed and implemented using Java Language (JDK1.4) as the front end and MS Access 2000 as the back end.

Java provides user friendly environment to implement the applications. Java swing provides GUI to their applications so that user can easily understand and handle the system. Since it allows keeping the database separately apart from the programming codes, it is possible to switch to the best latest database technology available in the market. Java provides object oriented capabilities to the implementation.
4.0 CONCLUSIONS

Three feedforward neural models can be used to analyze these factors. Input data to the neural network proposed are quantitative factors. Input data to the neural network proposed for qualitative factors can be factors related to the political effect considered. Third neural network consists of decision integration in which input data will be the outputs of above-mentioned neural networks. This facilitates to make right decision whether stock market is influenced by quantitative or qualitative factors. This project considers first part only, that prediction using quantitative factors such as Daily Average Turnover, Number of Shares traded per day, Number of Trades and Market Capitalization. Other two phases will be completed in future.

Network pruning and training optimization are two very important research topics which impact the implementation of financial neural networks. Financial neural networks must be trained to learn the data and generalize, while being prevented from overtraining and memorizing the data. Also, due to their large number of inputs, network pruning is important to remove redundant input nodes and speed-up training and recall. The major research thrust in this area should be determining better network architectures. The commonly used backpropagation network offers good performance, but this performance could be improved by using recurrence or reusing past inputs and outputs. The architecture combining neural networks and expert systems shows potential. Currently, implemented neural network have shown that the Efficient Market Hypothesis does not hold in practice, and that stock markets are probably chaotic systems. Until we more fully understand the dynamics behind such chaotic systems, the best we can hope for is to model them as accurately as possible. Neural networks appear to be the best modeling method currently available as they capture nonlinearities in the system without human intervention. Continued work on improving neural network performance may lead to more insights in the chaotic nature of the systems el. However, it is unlikely a neural network will ever be the perfect prediction device that is desired because the factors in a large dynamic system, like the stock market, are too complex to be understood for a long time.

ACKNOWLEDGMENTS

It is important to note that all neural networks used in this publication are extracted from Stuttgart Neural Networks Simulator (SNNS). Hence, copyright statement is applicable as per stated by SNNS authors. We would also like to appreciate Mr. Rajive Bandara, Senior Manager, Colombo Stock Exchange, for giving us the information about CSE and necessary data.