2013 International Conference on Communication Systems and Network Technologies

Optimal Neural Network Architecture for Stock Market Forecasting

Gitansh Khirbat, Rahul Gupta and Sanjay Singh
Department of Information & Communication Technology
Manipal Institute of Technology, Manipal University, Manipal-576104, INDIA
gitanshkhirbat.91@gmail.com, rahul.gupta.engg@gmail.com, sanjay.singh@manipal.edu

***Abstract*—Predicting stocks accurately has always intrigued the market analysts. A possible forecast of stocks is done using trading parameters and Price/Earnings ratio. With the advances in Artificial Neural Networks, it has become possible to analyze a data set in temporal domain. The use of Time Series Forecasting empowers us to predict the value of an entity in the future based on the previously obtained outputs. The current best fit solution for stock forecasting produces a forecast result with 58 % accuracy using feed-forward backpropagation neural network. In this paper, we have represented the data set containing financial stock price as a time series. This time series is forecasted by feeding it to a multi layer back propagation neural network. In real world scenario, stock prices are influenced by many non deterministic factors such as national & international economy and public confidence. This paper takes into account factors like Earnings Per Share (EPS) and public confidence and introduces an empirically defined neural network architecture of the form $[m - m/2 - m/10 - 1]$ which gives an optimized structure for predicting the future value of a stock by extrapolating the near future value by the present value comparisons. The experimental results obtained after the training and testing of the financial data are very promising. This increase in accuracy of our financial prediction is due to the factors incorporated for forecasting which can give a clear binary classification for buying or withholding the stock in the current market scenario.**

I. Introduction

Stock markets have always been a center of attraction for all the human beings at one time or another. This intriguing field of money making creates multiple confusions regarding buying or selling of stocks. The factors on which buying or selling depend cannot be quantified. Since all these factors are highly stochastic in nature, it is nearly impossible to correctly predict about a stock's value going up or down in the near future, say tomorrow [1].

Stock transactions may vary with respect to the volume of transactions, previous closure value, current trend in national or international financial market and many other factors such as national financial policies and confidence of customers in a company. These factors will be discussed in detail in the following sections. It is highly desirable to formulate a correlation between these factors. Often these factors are analyzed separately and a decision is taken on the basis of the more dominating factor.

It has been found experimentally that the rate of prediction for a company stock is 58.52% while training and 54.73% while testing. This is achieved through Support Vector Machines (SVM) [2].

In this paper, we have tried to analyze the financial stocks of different companies based on a set of internal and external factors. The stock data is collected from Yahoo! Finance [3]. These stocks are represented as financial Time Series and fed into a multilayer back-propagation neural network. The results are then used to formulate a non-linear correlation among the factors and are analyzed to extrapolate the future value of the company stock. Based on the empirical results we have identified an optimal neural network architecture of the form $[m - m/2 - m/10 - 1]$, where m is the number of neurons in each layer.

Rest of the paper is organized as follows. Section II gives brief description about financial stock market. Section III introduce to the parameters used for stock market prediction. Section IV describes about various trading strategies. Section V briefly explains about the neural network and time series forecasting. Section VI describes about the methodology adopted to find the optimal neural network architecture. Section VII discusses about the results obtained and finally section VIII concludes this paper.

II. Financial Stock Market

Investing some part of the company's assets as shares for public holding is a vital trade strategy for various companies. This is achieved by launching a well calculated percentage of company's assets as shares. These shares are then purchased by the customers. With the growth or depreciation of the company, the share prices are affected correspondingly. So, the profit or loss of the company results in corresponding profit or loss of the share holder. Transactions for these stocks are affected by many factors and policies, and are carried out at the Stock Market or Equity Market.

There are many types of transactions occurring in the Market. Some of them include retail investors, institutional investors such as mutual funds, banks, insurance companies and hedge funds, and public trade corporations which trade their own shares [4].

III. Indicators for Prediction

We have seen the importance of stock market in the previous section and how important it is for a country's economic growth and development. This section gives an insight into the measures of stock market prices.

978-0-7695-4958-3/13 $26.00 © 2013 IEEE
DOI 10.1109/CSNT.2013.120

A. Earnings Per Share (EPS)

The company stocks vary based on the Earnings Per Share (EPS) ratio. EPS is the main information that affects a company's stock price. EPS is a factor of company's credibility and profitability. It is defined as [4]:

$$EPS = \frac{\text{Net Income Dividends on Preferred Stocks}}{\text{Average Outstanding Shares}} \quad (1)$$

where, Net income is the total income of the specified company, dividends on preferred stocks gives the profit value of the stocks the buyer is interested in, and average outstanding shares means the shares which are not yet sold, i.e. they are in the state of liquid at the moment.

B. Price Earnings Ratio (P/E)

An estimate of current price of a company share with respect to its per-share earnings is given by the Price Earnings Ratio. It is defined as [4]:

$$P/E = \frac{\text{Market value per share}}{\text{Earnings per share (EPS)}} \quad (2)$$

where Market value per share is the market price of the concerned share and EPS ratio is the one discussed in the previous sub-section.

In the market terms, a higher and consistent upward P/E ratio means a positive future growth for a company's financial stock. Similarly, a low P/E ratio means that the company's stock prices are going low. This is a good time to buy stocks for the buyers.

C. Price Dividend Ratio (P/D)

Dividends are the profit which a company makes and distributes it to its shareholders. The price dividend ratio gives a measure of a company's stability in the financial market. Even if the stock prices do not grow over time, the dividends are a measure of retaining the shareholders by a company. The dividends are distributed equally among the company's shareholders from time to time. A constantly growing dividend amount indicates that the company is at profit. This ensures a confidence in the buyers as a positive indicator to buy the company's shares.

IV. Trading Strategies

This section elaborates upon the market analysis from a buyer's perspective and the strategies adopted by the buyers and sellers to efficiently perform the buying or selling operation, thereby optimizing the profit on the stocks. We have categorized various trading strategies as follows [5]:

- **Buy and Hold (B&H)**
  The Buy and Hold strategy is a strategy which is built around the concept of static investments. A company's stock is bought by the buyer for investment purpose and then kept for a long period of time, say 12 months. After this stocking period, the stock is sold. This strategy is typically used by those investors who want to earn a constant profit and who want to play safe by not dynamically buying and selling the stocks.
- **Stop and Objective (S&O)**
  The Stop and Objective strategy, unlike its previous counterpart, focuses on profit maximization of the stock in hand and the stock to be purchased. The investor observes and analyze the market thoroughly by applying some parameters such as those discussed in the section III. After analyzing the current trend and P/E value of the stock, the investor buys it and then keeps it for a short duration, after which he might sell it.
- **Neural Network based Buy and Hold (NN-B&H)**
  The B&H strategy is augmented using Neural Network. Similar to the B&H strategy, a company stock is purchased without much analysis, to be kept for a period of say 12 months. The NN decides which stock to buy based on the previous results for that stock with respect to time. The NN plays the role of a recommender for the NN-B&H strategy.
- **Neural Network based Stop and Objective (NN-S&O)**
  Similar to the simple S&O mechanism, the stocks are purchased based on the results obtained by applying some trading strategy and are kept in a dynamic state, i.e. buying and selling is dynamic based on demand and supply to maximize the profit. The selection of stock is done by a Multilayer Backpropagation Neural Network.

V. Neural Networks and Time Series Forecasting

In this section, we focus on the second part of the paper, i.e. Artificial Neural Network and Time Series Forecasting.

A. Artificial Neural Network (ANN)

An ANN is a massively parallel distributed processor made up of simple processing units. Basically it acts as a machine to store experiential knowledge of the system and makes use of it for making decisions for the future outputs. A typical ANN consists of three layers, i.e. an input layer, a hidden layer(s) and an output layer. A layer is made up of perceptron, which are the fundamental building blocks of an ANN [6].

The input data is fed into the ANN through the input layer. This is transitioned to the hidden layer(s) which may vary based on different scenario. Finally it is sent to the output layer, which gives the final output. The different inter-neuron links are called as synapses, and introduce a multiplier called synaptic weight, which is multiplied to the input pattern to give result to the immediate next layer of an ANN.

In this paper, we have considered a four layer network architecture with 2 hidden layers, details of which are mentioned in the next section. The results which we have got are very promising.

B. Time Series Forecasting (TSF)

Time Series Forecasting is the process of analyzing the times series by the facts we have obtained from the past and based on those facts forecast the future actions [7]. TSF is done by preparing models which are highly stochastic in nature.

There are various models for TSF namely Auto-regressive Moving Average, Auto-regressive Integrated Moving Average, Generalized Auto-regressive Conditional Heteroskedasticity model and Kalman Filters. However, there are problems associated with all these models in terms of optimization of forecasting due to either function approximation technique or difficulty in analyzing dynamic data. This is the reason why ANN model is used for Time Series Forecasting.

VI. Methodology

This section provides an in-depth discussion of the NN model used and the concepts used for our experiment.

A. Formation of the ANN

The ANN which we have used consists of one input layer, one output layer and two hidden layers. The purpose behind choosing two hidden layers is to improve the accuracy of prediction without compromising time complexity [8]. We are considering a maximum number of epochs to be 5000, the reason for which is described in section VII-A. The various layers of the Neural Network are elaborated below:

- **Input Layer**
  This layer contains the daily price of a company's stock considered for analysis. The stock prices for a period of 44 days have been considered. The financial data quotations are accurate and are taken from the Yahoo! Finance [3].
- **Hidden Layer**
  There are two hidden layers in our network. The first hidden layer consists of 22 neurons. The number of neurons considered for the input layer analysis is restricted to half of the input data parameters which are being fed into the neural network. In the second hidden layer, we have restricted the number of neurons to be equal to almost one-tenth of the number of neurons in the first hidden layer. Plots for different values in the range of [0, 22] was taken, and the results for the number of neurons equal to one-tenth of the first hidden layer, i.e. 2, was found to be most consistent. These results are based on our experimental observations which are described in the section VII-A with required plots.
- **Output Layer**
  This layer shows the result of the final calculations performed in the hidden layers. The result obtained from this layer is compared with the desired output and the error is back propagated to the hidden layers and input layers, to be corrected and to get final output as close to the desired output as possible.

The complete NN architecture can be visualized as $[m-m/2-m/10-1]$, where m is the total number of neurons in the first layer corresponding to the number of inputs provided to the NN.

B. Training of the ANN

The ANN thus formed in sub-section 6.1 is used to forecast the financial time series we have. For forecasting, the system must be trained first in order to provide some basis for comparisons to the real data of a company's stock. We have obtained the data for a company's stock from Yahoo! Finance and have used this data of a period of 44 days to analyze the stock prices. These variations of stock prices are stored in the system as training set. We have maintained indices for all these variations in our source code as training set. These training sets are then used to compare the stock value after each epoch and the error calculated is then back propagated in the hidden layers.

C. Synaptic Weights and Bias Assignments

The synaptic weights are initialized to values ranging from [0, 1]. The bias is set to unity for both hidden layers. The weights are initialized arbitrarily, which are later organized once the learning starts.

D. Binary Classification for Stock Analysis

The NN is trained using the financial data obtained from the training set. Once the training is done, an extensive testing of the financial data takes place. The mean for the actual data as well as the predicted data is calculated next and a comparison is made. If the mean of the predicted data from the NN exceeds the actual data from the training set, then the system suggests the user to purchase a stock. Otherwise, the user is discouraged to buy that stock and wait for the appropriate time to buy it.

VII. Experimental Results and Discussion

This section gives details about the results obtained by applying the above ANN to the financial data. The important observations are described in the further sub-section.

A. Regression Analysis of Actual Trend of Stock Prices v/s Predicted Price

In the experiment, we have set up a feed-forward backpropagation neural network which is given a data set of current stock market prices as an input. For analysis purpose, we are focusing on two values viz market open value and market close value respectively.

The number of input data sequence considered are 44 in number, which are fed into the NN through the input layer. The NN architecture set up after empirical testing of various values was finalized to $[m - m/2 - m/10 - 1]$, value of m being 44 in this case.

By comparing the plots for the mentioned number of neurons, we came to a conclusion to choose the total number of neurons in the first hidden layer to be equal to half of the input data set. The results are even found to be consistent with the mean proportion for all the considered values. Fig 1,3 and 5 show the learning with lesser neurons for 50,500 and 5000 epochs. Figures 2, 4 and 6 show the learning with neurons equal to half the input data set for 50, 500 and 5000 epochs. The same methodology was chosen to decide the number of neurons in the second hidden layer to be equal to almost one-tenth of the number of neurons in the first hidden layer. Comparisons are shown in Fig 1,3 and 5.

The maximum number of epochs are taken as 5000 for an improved result. It is well known that learning becomes better with the number of epochs. Initially the number of epochs was taken as 50 and the regression was plotted. Then the epoch was increased 10 times to plot the regression. A further ten-fold increase in the number of epochs resulted in the best fit data. A comparison is shown in Figs 1 to 6.

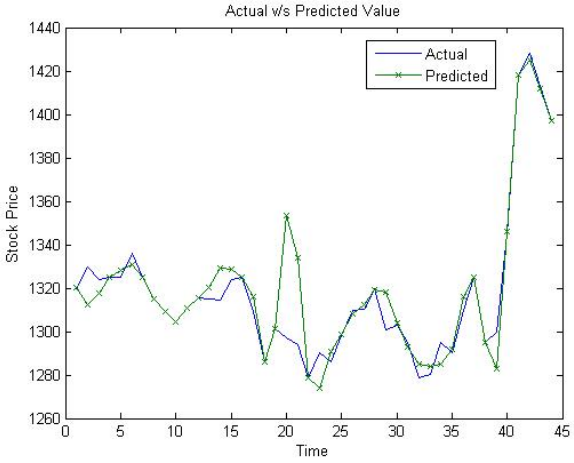

Fig. 1. Less number of neurons in hidden layer with 50 epochs

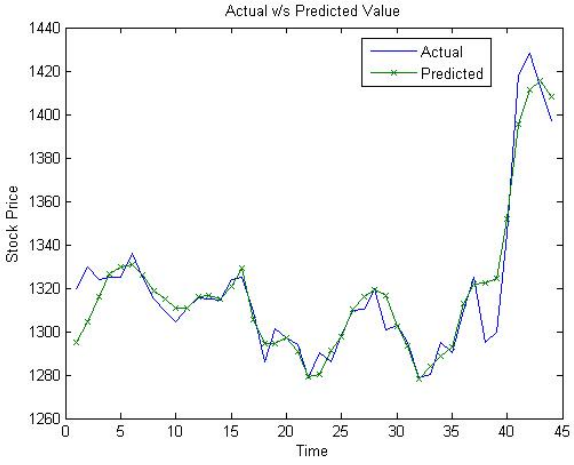

Fig. 2. More number of neurons in hidden layer with 50 epochs

B. Best Fit Results With Varying Epoch

Figures 7 to 9 show the results obtained from the training of the NN, followed by testing. The plots are compared and finally means are calculated for actual and predicted values. Finally system makes a decision based on the comparison of means regarding whether to buy a stock or withhold it. It is observed that the rate of prediction increases with the number of epochs. The NN used is able to forecast the stock values more accurately in 5000 epochs as compared to 50 epochs.

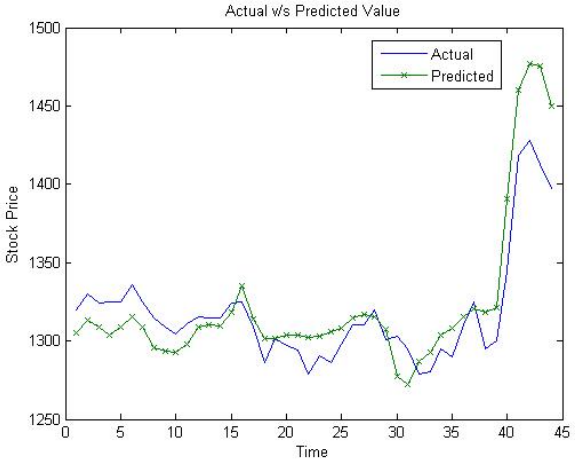

Fig. 3. Less number of neurons in hidden layer with 500 epochs

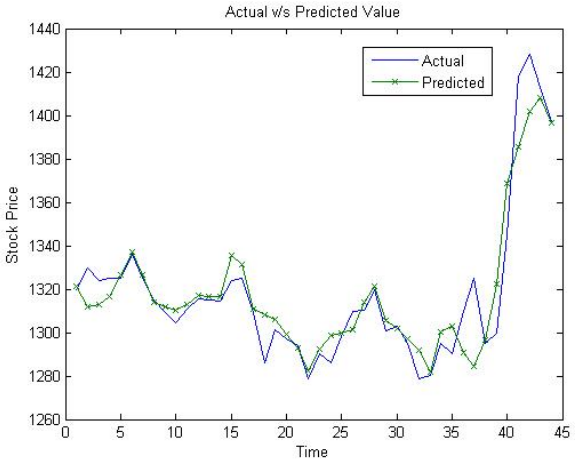

Fig. 4. More number of neurons in hidden layer with 500 epochs

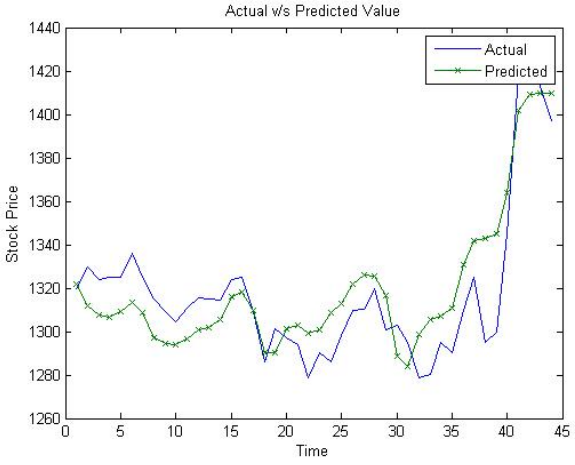

Fig. 5. Less number of neurons in hidden layer with 5000 epochs

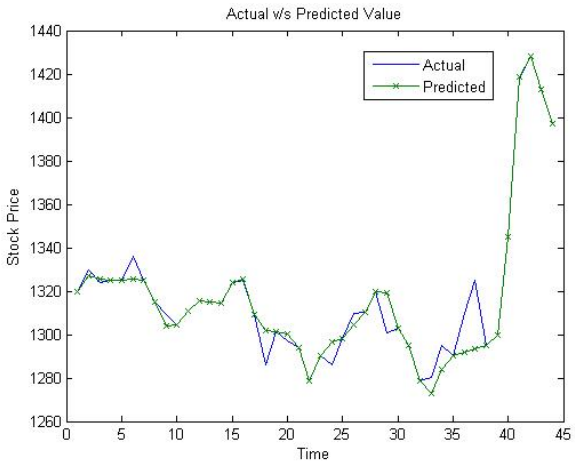

Fig. 8. Best fit learning curve with 500 epochs

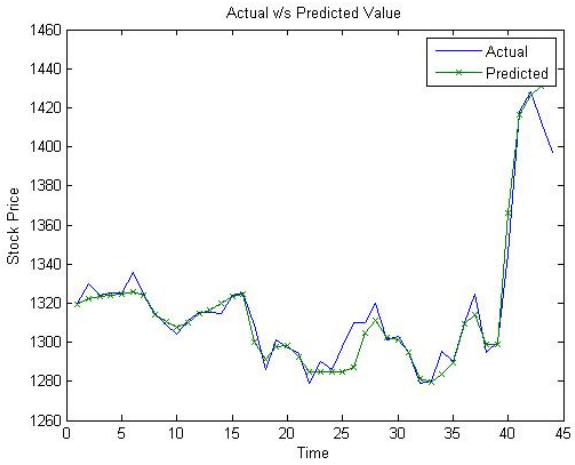

Fig. 9. Best fit learning curve with 5000 epochs

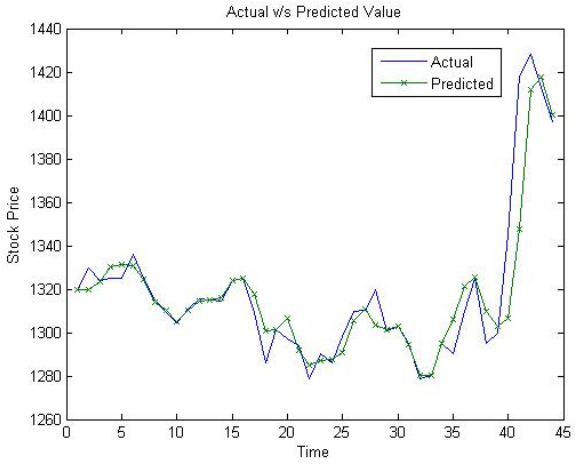

Fig. 6. More number of neurons in hidden layer with 5000 epochs

Finally, the probability with which the user can perform his transactions increases with an increase in the number of epochs.

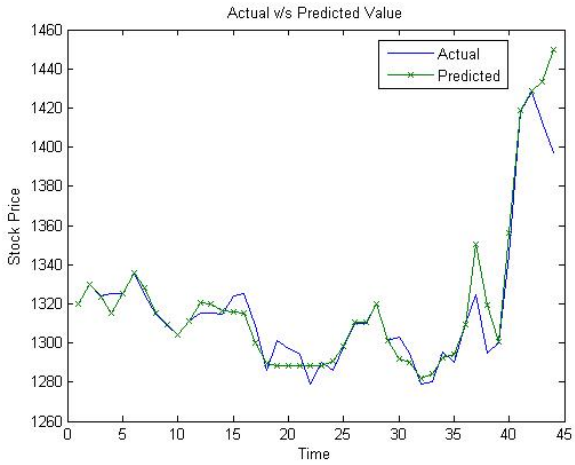

Fig. 7. Best fit learning curve with 50 epochs

VIII. Conclusion

Use of ANN is a well known technique for stock market forecasting. In this paper we have proposed an optimal neural network architecture for stock market prediction based on empirical observations. The generalized form of this architecture can be written as $[m - m/2 - m/10 - 1]$. This architecture holds good for the financial stock of a company. We have acquired financial stock data of companies and expressed it temporally as a time series, upon which the analytics have been performed. The final system is able to compare the actual market value of a stock with the predicted value. The system further correctly predicts whether to buy or withhold the stock based on a mean based binary classification. So, this paper provides a measure to select the optimized parameters in terms of neurons and epochs to provide an accurate forecasting.

References

[1] K. Kim and I. Han, "Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index," in *Expert System Application*, Vancouver, Canada, April 2000, pp. 125–132.

[2] K. jae Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, pp. 307 – 319, March 2003.

[3] Yahoo!, "Stock analysis at yahoo! finance," [Available Online]http://in.finance.yahoo.com/, November 2012.

[4] D. Goodwin, "Price-earnings ratio - p/e ratio," [Available Online] http://www.investopedia.com/terms/p/price-earningsratio.asp, July 2011.

[5] T. Kimoto, K. Asakawa, and M. Yoda, "Trading s & p 500 stock index futures using a neural network," in *Proceedings of the Annual International Conference on Arti-cial Intelligence Applications*, Wall Street, New York, June 1995, pp. 63–72.

[6] G. Zhang, B. Patuwo, and M. Hu, "Forecasting with artificial neural networks: the state of the art," *International Journal of Economics and Finance*, vol. 4, pp. 139–151, March 2012.

[7] T. Kamijo and T. Tanigawa, "Stock price pattern recognition: a recurrent neural network approachk," in *Proceedings of the International Joint Conference on Neural Networks*, San Diego, CA, June 1990, pp. 215–221.

[8] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. London: Cambridge University Press, 2007.