Hybrid Intelligent Systems for Stock Market Analysis

Ajith Abraham[1], Baikunth Nath[1] and Mahanti P K[2]

[1]School of Computing & Information Technology
Monash University (Gippsland Campus), Churchill 3842, Australia
Email: {Ajith.Abraham, Baikunth.Nath}@infotech.monash.edu.au

[2]Department of Computer Science and Engineering
Birla Institute of Technology, Mesra-835 215, India
Email: deptcom@birlatech.org

Abstract: The use of intelligent systems for stock market predictions has been widely established. This paper deals with the application of hybridized soft computing techniques for automated stock market forecasting and trend analysis. We make use of a neural network for one day ahead stock forecasting and a neuro-fuzzy system for analyzing the trend of the predicted stock values. To demonstrate the proposed technique, we considered the popular Nasdaq-100 index of Nasdaq Stock Market[SM]. We analyzed the 24 months stock data for Nasdaq-100 main index as well as six of the companies listed in the Nasdaq-100 index. Input data were preprocessed using principal component analysis and fed to an artificial neural network for stock forecasting. The predicted stock values are further fed to a neuro-fuzzy system to analyze the trend of the market. The forecasting and trend prediction results using the proposed hybrid system are promising and certainly warrant further research and analysis.

1. Introduction

During the last decade, stocks and futures traders have come to rely upon various types of intelligent systems to make trading decisions. Several hybrid intelligent systems have in recent years been developed for modeling expertise, decision support, complicated automation tasks etc. We present a hybrid system fusing neural networks and neuro-fuzzy systems aided with a few well-known analytical techniques for stock market analysis.

Nasdaq-100 index reflects Nasdaq's largest companies across major industry groups, including computer hardware and software, telecommunications, retail/wholesale trade and biotechnology [1]. The Nasdaq-100 index is a modified capitalization-weighted index, which is designed to limit domination of the Index by a few large stocks while generally retaining the capitalization ranking of companies. Through an investment in Nasdaq-100 index tracking stock, investors can participate in the collective performance of many of the Nasdaq stocks that are often in the news or have become household names. In this paper we attempt to forecast the values of six individual stocks and group index as well as the trend analysis of the different stocks. Individual stock forecasts and group trend analysis might give some insights of the actual performance of the whole index in detail. To demonstrate the efficiency of the proposed hybrid system we considered the two years stock chart information (ending 13 March 2001) of six major industry groups listed on the national market tier of the Nasdaq Stock Market (Nasdaq-100 index).

Neural networks are excellent forecasting tools and can learn from scratch by adjusting the interconnections between layers. Fuzzy inference systems are excellent for decision making under uncertainty. Neuro-fuzzy computing is a popular framework wherein neural network training algorithms are used to fine-tune the parameters of fuzzy inference systems. For the stock forecasting purpose we made use of a neural network trained using scaled conjugate gradient algorithm. However, the forecasted stock values might deviate from the actual values. We modeled the deviation of the predicted value from the required value as a fuzzy variable and used a fuzzy inference system to account for the uncertainty and decision-making. In section 2 we explain the details of the proposed hybrid system and its components followed by experimentation setup and results in section 3. Conclusions and further work are provided towards the end.

2. A Soft Computing Framework for Stock Market Analysis

Soft computing introduced by Lotfi Zadeh is an innovative approach to construct computationally intelligent hybrid systems consisting of neural network, fuzzy inference system, approximate reasoning and derivative free optimization techniques [4]. In contrast to conventional artificial intelligence, which only deals with precision, certainty and rigor the guiding principle of soft computing is to exploit the tolerance for imprecision, uncertainty, low solution cost, robustness, partial truth to achieve tractability, and better rapport with reality.

Figure 1 depicts the hybrid intelligent system model for stock market analysis. We start with data preprocessing, which consists of all the actions taken before the actual data analysis process starts. It is essentially a transformation T that transforms the raw real world data vectors X_{ik}, to a set of new data vectors Y_{ij}. In our experiments, we used Principal Component Analysis (PCA) [3], which involves a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called principal components. In other words, PCA performs feature extraction. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. The preprocessed data is fed into the Artificial Neural Network (ANN) for forecasting the stock outputs. ANN "learns" by adjusting the interconnections (called weights) between layers. When the network is adequately trained, it is able to generalize relevant output for a set of input data. Learning typically occurs by example through training, where the training algorithm iteratively adjusts the connection weights. We trained the ANN using a Scaled Conjugate Gradient Algorithm (SCGA) [6]. In the Conjugate Gradient Algorithm (CGA) a search is performed along conjugate directions, which produces generally faster convergence than steepest descent. A search is made along the conjugate gradient direction to determine the step size, which will minimize the performance function along that line. A line search is performed to determine the optimal distance to move along the current search direction. Then the next search direction is determined so that it is conjugate to previous search direction. The general procedure for determining the new search direction is to combine the new steepest descent direction with the previous search direction. An important feature of the CGA is that the minimization performed in one step is not partially undone by the next, as it is the case with gradient descent methods. However, a drawback of CGA is the requirement of a line search, which is computationally expensive. The SCGA is basically designed to avoid the time-consuming line search at each iteration. SCGA combine the model-trust region approach, which is used in the Levenberg-Marquardt algorithm with the CGA. Detailed step-by-step descriptions of the algorithm can be found in Moller[6].

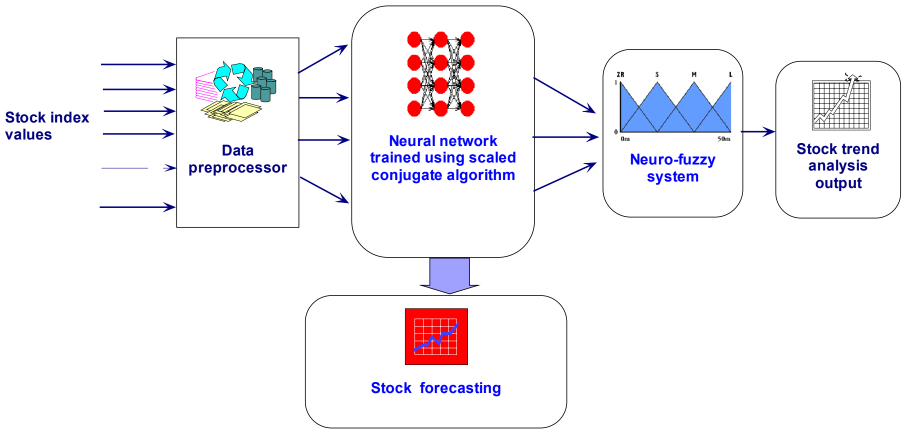

Figure 1. Block diagram showing hybrid intelligent system for stock market analysis

The forecasted outputs by the neural network are further analyzed using a neuro-fuzzy system. This time our aim is to analyze the upward and downward trends of the different forecasted stocks. Since the forecasted values will deviate from the desired value (depending upon the prediction efficiency of ANN), we propose to make use of the uncertainty modeling capability of Fuzzy Inference System (FIS)[7]. We define a neuro-fuzzy system as a combination of ANN and FIS in such a way that neural network learning algorithms are used to determine the parameters of FIS [5]. We used an Evolving Fuzzy Neural Network (EFuNN) implementing a Mamdani type FIS and all nodes are created during learning. EFuNN has a five-layer structure as shown in Figure 2. The input layer followed by a second layer of nodes representing fuzzy quantification of each input variable space. Each input variable is represented here by a group of spatially arranged neurons to represent a fuzzy quantization of this variable. Different Membership Functions (MFs) can be attached to these neurons (triangular, Gaussian, etc.). The nodes representing MFs can be modified during learning. New neurons can evolve in this layer if, for a given input vector, the corresponding variable value does not belong to any of the existing MF to a degree greater than a membership threshold. The third layer contains rule nodes that evolve through hybrid supervised/unsupervised learning. The rule nodes represent prototypes of input-output data associations, graphically represented as an association of hyper-spheres from the fuzzy input and fuzzy output spaces. Each rule node *r* is defined by two vectors of connection weights – $W_1\ (r)$ and $W_2\ (r)$, the latter being adjusted through supervised learning based on the output error, and the former being adjusted through unsupervised learning based on similarity measure within a local area of the input problem space. The fourth layer of neurons represents fuzzy quantification for the output variables. The fifth layer represents the real values for the output variables. EFuNN evolving algorithm used in our experimentation was adapted from [2].

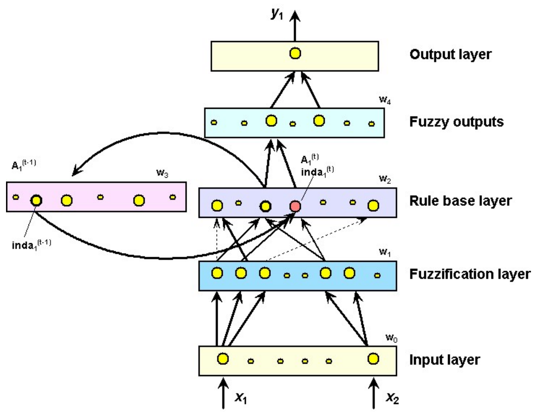

Figure 2. Architecture of EFuNN

The neuro-fuzzy network is trained using the trend patterns of the different stock values. The difference between the day's stock value and the previous day was calculated and used for training the NF system. If all the stock values were increasing we classified it as positive trend "1" and "0" otherwise. The proposed NF system is capable of providing detailed trend analysis of individual stocks and also interdependencies of various stocks and how they affect the overall index.

3. Experimentation setup and test results

We considered 24 months stock data [1] for training and analyzing the efficiency of the proposed hybrid intelligent system. In our experiments, we used Nasdaq-100 main index values and six other companies listed in the Nasdaq-100 index. Apart from the Nasdaq-100 index (IXNDX); the other companies considered were Microsoft Corporation (MSFT), Yahoo! Inc. (YHOO), Cisco Systems Inc. (CSCO), Sun Microsystems Inc. (SUNW), Oracle Corporation (ORCL) and Intel Corporation (INTC). Figures 3 and 4 depict the variation of stock values for a 24 months period from 22 March 1999 to 20 March 2001.

For each *t*, the stock values *x (t)* were first standardized and *x (t-1), x (t-2), x (t-3)* were computed. The data was then passed through the data pre-processor to ensure that the input vectors were uncorrelated. The PCA analysis also revealed that by providing only *t* and *x (t)* the neural networks could be trained within the required accuracy. 80% of the data was used for training and remaining was used for testing and validation. The same set of data was used for training and testing the neuro-fuzzy system. While the proposed neuro-fuzzy system is capable of evolving the architecture by itself, we had to perform some initial experiments to decide the architecture of the ANN. More details are reported in the following sections. Experiments were carried out on a Pentium II 450MHz machine and the codes were executed using MATLAB. Test data was presented to the network and the output from the network was compared with the actual stock values in the time series.

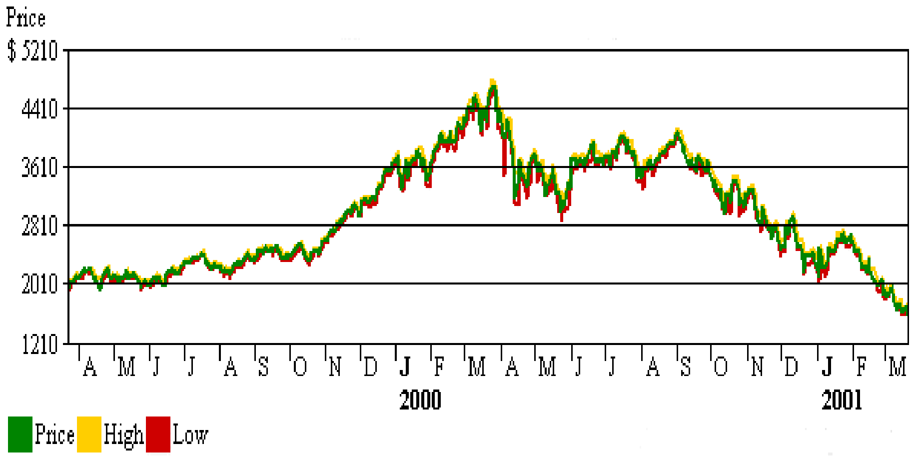

Figure 3. 24 months data of Nasdaq-100 index adapted from [1]

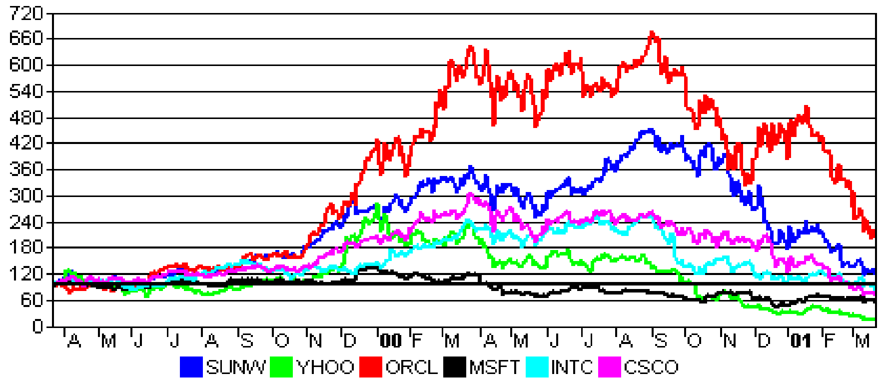

Figure 4. 24 months data of 6 companies adapted from [1]

ANN – SCG Algorithm

We used a feedforward neural network with 8 input nodes and two hidden layers consisting of 20 neurons each. We used tanh-sigmoidal activation function for the hidden neurons. The training was terminated after 2000 epochs. The test data was passed through the network after the training was completed.

EFuNN Training

Each of the input variables consists of the difference in the stock value (example. today's value – yesterday's value). For training the EFuNN, we had 8 input variables, the standardized stock value differences and the time factor. We used 4 membership functions for each of the 8 input variable and the following EFuNN parameters: sensitivity threshold *Sthr*=0.99, error threshold *Errthr*=0.001. Online learning of EFuNN created 503 rule nodes and the training error achieved was 6.5E-05.

However we report only the collective trend of all the seven stock values. If all the trends were increasing we classified as "1" and "0" if the trends were going down.

Performance and Results Achieved

Table 1 summarizes the training and test results achieved for the different stock values. Figure 5 and 6 depicts the test results for the prediction of Nasdaq-100 index and other company stock values. Table 2 summarizes the trend prediction results using EFuNN.

Table 1. Training and testing results using neural network

	Nasdaq	Microsoft	Sun	Cisco	Yahoo	Oracle	Intel
Learning epochs	2000						
Training error (RMSE)	0.0256						
Testing error (RMSE)	0.028	0.034	0.023	0.030	0.021	0.026	0.034
Computational load	7219.2 Giga Flops						

Table 2. Test results of trend classification using EFuNN

	Actual quantity	EFuNN classification	% Success
Positive trends	22	22	100
Negative trends	78	78	100
Computational load	18.25 Giga Flops		

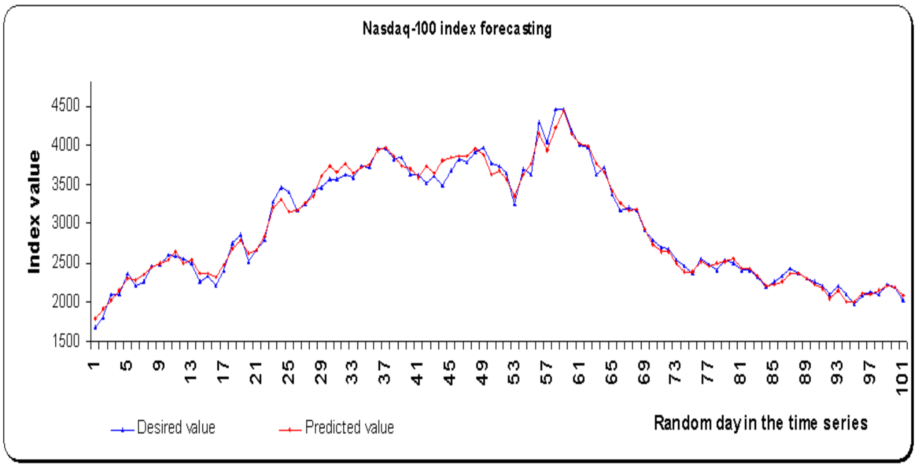

Figure 5. Forecast test results for Nasdaq-100 index

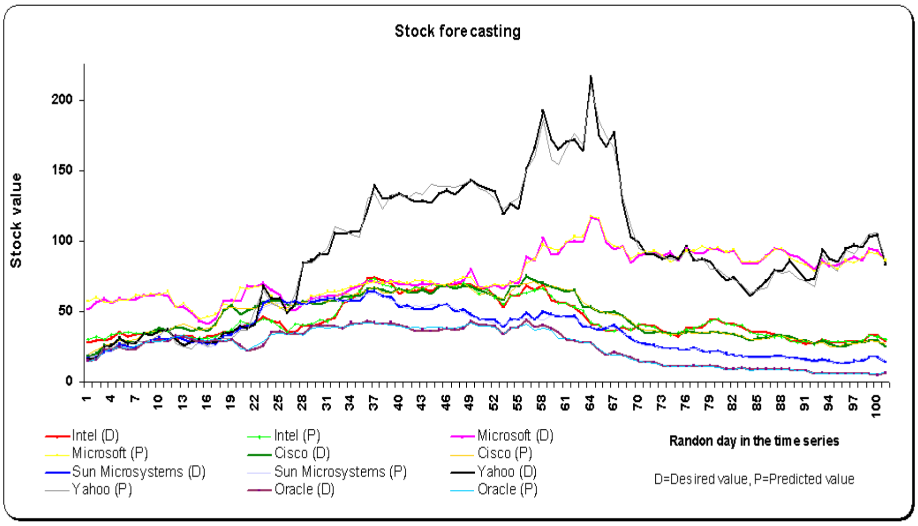

Figure 6. Forecast test results for the six companies listed under Nasdaq-100 index

Figure 7 shows the test results for the collective trend prediction of Nasdaq-100 index and the six 6 company stock values using EFuNN.

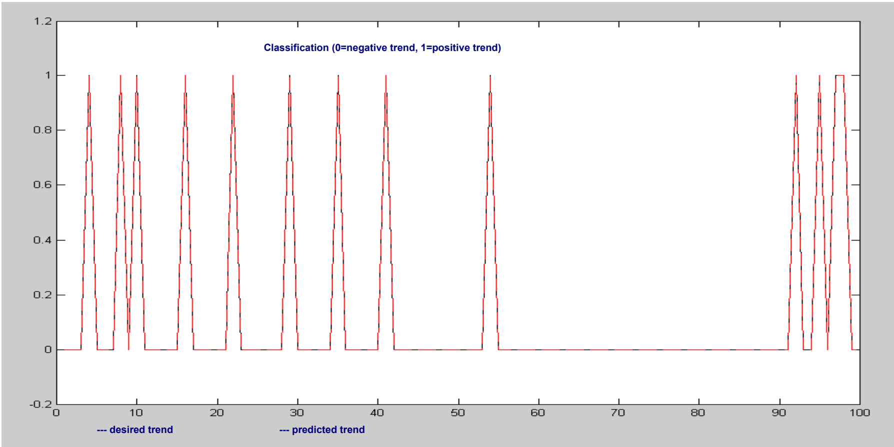

Figure 7. Test results for trend prediction using EFuNN

4. Conclusions

In this paper, we proposed a hybrid intelligent system based on an artificial neural network trained using scaled conjugate algorithm and a neuro-fuzzy system for stock market analysis. For forecasting stocks, the RMSE on test data are comparatively small showing the reliability of the developed prediction model. The proposed neuro-fuzzy model also gave 100% trend prediction showing the efficiency of the technique. The proposed hybrid system can be easily implemented and the empirical results are very promising. From the viewpoint of the stock exchange owner, participating companies, traders and investors the technique might help for better understanding of the day-to-day stock market performance.

The stock forecast error could have been improved if individual neural networks were used rather than a single network. Also various trend analyses could have been done using the proposed neuro-fuzzy system. Some of the possible analyses are individual stock trend predictions, interdependency of different stocks with respect to the main index as well as individual companies. Our future works will be organized in that direction.

References

[1] Nasdaq Stock Market^{SM}: http://www.nasdaq.com

[2] Kasabov N and Qun Song, *Dynamic Evolving Fuzzy Neural Networks with 'm-out-of-n' Activation Nodes for On-line Adaptive Systems*, Technical Report TR99/04, Department of information science, University of Otago, 1999.

[3] Duszak Z and Loczkodaj W W, *Using Principal Component Transformation in Machine Learning*, Proceedings of International Conference on Systems Research, Informatics and Cybernetics, Baden-Baden Germany, p.p 125-129, 1994.

[4] Zadeh LA, *Roles of Soft Computing and Fuzzy Logic in the Conception, Design and Deployment of Information/Intelligent Systems*, Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration with Applications, O Kaynak, LA Zadeh, B Turksen, IJ Rudas (Eds.), pp1-9, 1998.

[5] Abraham A & Nath B, *Designing Optimal Neuro-Fuzzy Systems for Intelligent Control*, In proceedings of the Sixth International Conference on Control Automation Robotics Computer Vision, (ICARCV 2000), Singapore, December 2000.

[6] Moller A F, *A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning*, Neural Networks, Volume (6), pp. 525-533, 1993.

[7] Cherkassky V, *Fuzzy Inference Systems: A Critical Review*, Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration with Applications, Kayak O, Zadeh LA et al (Eds.), Springer, pp.177-197, 1998.