Available online at www.sciencedirect.com

ScienceDirect

Expert Systems with Applications 34 (2008) 3043–3054

Expert Systems with Applications

www.elsevier.com/locate/eswa

Predicting stock index increments by neural networks: The role of trading volume under different horizons

Xiaotian Zhu [a], Hong Wang [b,*], Li Xu [c], Huaizu Li [d]

[a] National University of Singapore, School of Computing, Singapore, Old Dominion University, College of Business & Public Administration, Department of Finance, 2004 Constant Hall, Norfolk, VA 23529, USA

[b] School of Business and Economics, Department of Business Administration, North Carolina A&T State University, Greensboro, NC 27411, USA

[c] Department of Information Technology and Decision Science, Old Dominion University, Norfolk, VA 23529, USA

[d] School of Management, Xian Jiaotong University, Xian 710049, China

Abstract

Recent studies show that there is a significant bidirectional nonlinear causality between stock return and trading volume. In this research, we reinforce this statement and the results presented in some earlier literatures and further investigate whether trading volume can significantly improve the prediction performance of neural networks under short-, medium-and long-term forecasting horizons. An application of component-based neural networks is used in forecasting one-step ahead stock index increments. The models are also augmented by the addition of different combinations of indices' and component stocks' trading volumes as inputs to form more general ex-ante forecasting models. Neural networks are trained with the data of stock returns and volumes from NASDAQ, DJIA and STI indices. Results indicate that augmented neural network models with trading volumes lead to improvements, at different extents, in forecasting performance under different terms of forecasting horizon. Empirical results indicate that trading volumes lead to modest improvements on the performance of stock index increments prediction under medium-and long-term horizons.

Keywords: Component-based neural networks; Financial forecasting; Stock trading volume

1. Introduction

It has been indicated that conventional statistical techniques for forecasting have reached their limitation in applications with nonlinearities in the dataset (Refenes, Zapranis, & Francies, 1994). Artificial neural network, a computing system containing many simple nonlinear computing units or nodes interconnected by links, is a well-tested method for financial analysis on the stock market (Refenes, Burgess, & Bentz, 1997), pattern recognition and optimization (Wang, Jacob, & Rolland, 2003; Wang, 2005). Neural networks have been shown to be able to decode nonlinear time series data which adequately describe the characteristics of the stock markets (Lapedes & Farber, 1987). In past decades, neural networks have been explored by many researchers for financial forecasting. Among these researches, some are conducted particularly on forecasting the value of a stock index (Abhyankar, Copeland, & Wang, 1997; Castiglione, 2001; Freisleben, 1992; Kim & Chun, 1998; Liu & Yao, 2001; Phua, Zhu, & Koh, 2003; Refenes et al., 1994; Resta, 2000; Sitte & Sitte, 2000; Tino, Schittenkopf, & Dorffner, 2001; Yao & Poh, 1996; Yao & Tan, 2000). Using neural networks to model and predict stock market returns has been the subject of recent empirical and theoretical investigations by academics and practitioners alike. However, several design factors have significant impact on the accuracy of neural network forecasting. Input selection is one of the demanding and intricate tasks of training neural networks.

The existing neural network models may be classified as univariate or multivariate models. A univariate model uses the past value of the time series for building a forecaster.

* Corresponding author.
E-mail addresses: xxzhu@odu.edu (X. Zhu), hwang@ncat.edu (H. Wang).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.06.023

The disadvantage of the univariate model is that it doesn't consider the environmental effects and interaction among factors other than outputs. A multivariate model uses additional information such as market index, technical indicators, or fundamental factors of companies as inputs. The disadvantage of the multivariate model is that the selection of inputs has always been a difficult task (Yao, Tan, & Poh, 1999). Usually, trading volume is considered one of the fundamental factors, which is beneficial for long-term forecasting (Kanas & Yannopoulos, 2001). Recently the relationships between stock price and volume or between volatility and volume have become a hot topic in theory as well as in empirical research (Brooks, 1998; Campbell, Grossman, & Wang, 1993; Hiemstra & Jones, 1994; Wang, Phua, & Lin, 2003). This study aims to find whether trading volume is beneficial to stock market forecasting by using neural networks under different terms of forecasting horizon, especially in stock index increment prediction.

It is more helpful to understand the market through studying the joint dynamics of stock prices and trading volume than by focusing only on the univariate dynamics of stock prices (Gallant, Rossi, & Tauchen, 1992). This argument indicates that trading volume is related to stock price and may be an important determinant of stock prices. Campbell et al. (1993) investigate the relationship between stock market trading volume and the serial correlation of daily stock returns. Their results suggest that the decrease or increase of stock price has a strong relation with the decrease or increase in trading volume. In other words, trading volume has an impact on the dynamic of stock price. Hiemstra and Jones (1994) find evidence of significant bidirectional nonlinear causality between returns and volume. In addition, the authors examine whether the nonlinear causality from volume to return can be explained by volume serving as a proxy for information flow in the stochastic process, and they find that there exists nonlinear causality from volume to returns.

Both theoretical and empirical studies have proven a nonlinear relation between stock return and trading volume. However, whether this nonlinear relationship can be of help to improve the forecasting performance is still an open question. Brooks (1998) explores a number of statistical models (both linear and nonlinear) for predicting the daily stock return volatility of an aggregate of all stocks traded on the New York Stock Exchange market (NYSE). The author finds that lagged volume leads to very modest improvement in forecasting performance. It is inferred that such results are attributed to the transformation method applied in the data, which may lose the important information in trading volume. Brooks (1998) simply focuses on the relationship of trading volume and stock volatility. It is still unclear whether trading volume can improve the forecasting performance of stock return. Kanas and Yannopoulos (2001) introduce stationary transformation of dividends and trading volume as fundamental explanatory variables to neural network models. Their results indicate that inclusion of nonlinear term in the relation between stock returns and fundamentals can improve the out-of-sample forecasting accuracy. However this study just focuses on long-term (monthly return) forecasting and not short-term, limiting the validation of the conclusion.

Our study puts emphasis on determining whether trading volume can improve the forecasting performance of neural networks or whether neural networks can take advantage of such nonlinearity to get more accurate results. The terminal goal of this research is to investigate to what degree trading volume can improve the performance of stock return forecasting and to give some guidance on input selection to achieve such performance. Four kinds of neural networks with different inputs are trained to simulate financial time series, and the results are obtained from daily, weekly and monthly financial time series of three well known stock markets indices: National Association of Securities Dealers Automated Quotation (NASDAQ) index, Dow Jones Industry Average (DJIA) index and Straits Times Index (STI), traded in the Singapore Exchange.

The paper is organized as following: Section 2 gives a brief introduction on component-based neural network financial forecasting model; Section 3 presents the details of our experiment design; Section 4 discusses the results and analyzes the possible reasons; Section 5 draws the conclusions.

2. Component based neural network forecasting model

Feed-forward neural networks are the most commonly used networks for a variety of applications in finance and accounting (Coakley & Brown, 2000). In this research, we also use the framework of feed-forward neural networks for financial forecasting. We notice that the stock indices are usually computed from the values of their component instruments. Let I_t be the stock market index at time t, and m be the number of component stocks of I_t. Then I_t can be computed by

$$I_t = \sum_{j=1}^{m} w_j p_{t,j} \quad (1)$$

where $P_{t,j}$ is the price of the component stock j (C_j) at time t, and w_j is the weighting coefficient for C_j. When internal/external information related to a particular component stock is perceived, the price of that stock will change, and this will cause the corresponding index to change as well. Hence it is natural and logical to predict a market index by considering the prices of its component stocks. For predicting a general financial market index, we propose the following component-based forecasting model:

$$I_{t+1} = f(C_{1t}, C_{2t}, \ldots, C_{mt}, I_t) \quad (2)$$

where $C_{1t}, C_{2t}, \ldots, C_{mt}$ are the closing prices of the component stocks $C_1, C_2, \ldots, C_m$ at time t, while I_t and I_{t+1} are values of the market indices computed at time t and $t+1$, respectively.

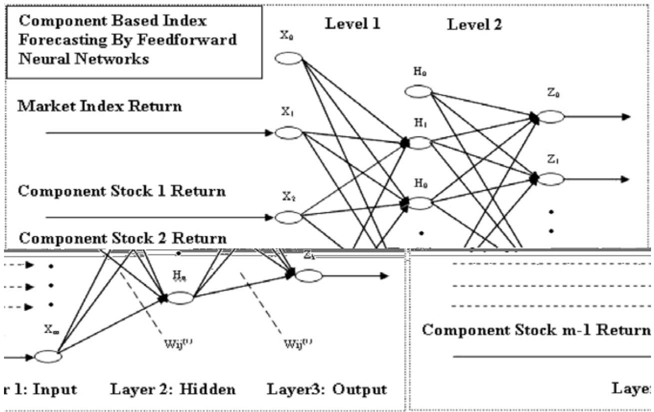

Fig. 1. The structure of the feed forward neural networks for component based financial index forecasting.

For the purpose of conducting experiments, we choose a three-layered feed-forward neural network architecture for component based financial forecasting. Fig. 1 shows the structure of a component-based neural network forecasting model. The input, hidden and output layers are noted as $\{X, H, Z\}$, respectively. Here the input layer X has $(m+1)$ neurons with $X_0=1$, the hidden layer H have $(n+1)$ neurons with $H_0=1$, and the output layer Z has K output values. Let $w^{(L)}$ denote the weights at Level L. Let (X_p, Y_p), $p=1,2,\ldots,P$ be the set of given input/output vectors for training the neural network. For each input X_{pi} $(i=0,1,\ldots,m)$, the neurons H_{pj} and Z_{pk} are calculated according to the following equations:

$$H_{pj}(w^{(1)})=f^{(1)}\left(\sum_{i=0}^{m}X_{pi}w_{ij}^{(1)}\right),\quad j=1,2,\ldots,n \tag{3}$$

$$Z_{pk}(w^{(1)},w^{(2)})=f^{(2)}\left(\sum_{j=0}^{n}H_{pj}(w^{(1)})\cdot w_{jk}^{(2)}\right)\quad k=1,2,\ldots,K \tag{4}$$

Here, the transfer function $f^{(1)}$ from the input to the hidden layer is the sigmoid function $y=1/(1+e^{-x})$, and the transfer function $f^{(2)}$ from the hidden layer to the output layer is the linear function $y=x$. The training of the neural network is done by feeding the set of input-output vectors (X_p, Y_p) to the neural networks and by minimizing the following objective function:

$$g(w)=\frac{1}{p\times k}\sum_{p=1}^{p}\sum_{k=1}^{k}[y_{pk}-z_{pk}(w)]^2 \tag{5}$$

where $w=w^{(1)}\cup w^{(2)}$, represents the weights of the neural network. The error function g defined by (5) is the mean-squared error (MSE).

3. Experiment design

The data from all stock indices, NASDAQ, DJIA and STI, are employed to train neural networks. Component-based three-layer feed-forward neural networks with various numbers of hidden neurons are employed to model nonlinearity among time series. The details are shown in the following:

3.1. Data

As we are going to investigate whether trading volume can improve the neural network prediction performance under different forecasting horizons, the daily, weekly and monthly financial time series data from three stock markets of NASDAQ, DJIA and STI are employed in this research. Observations for daily, weekly and monthly time series are ranged from November 5th, 1997, June 12th, 1990 and March 1st, 1989 to October 18th 2005, respectively with a total of 2000 daily, 800 weekly and 200 monthly records for each horizon. Both the indices' and its corresponding component stocks' price returns and trading volumes are considered as inputs for the component-based neural network forecasting models. To reflect the gains and losses of investors, the prices of stock indices and its corresponding component stocks are converted to their respective returns. In fact, we compute the returns R_{it} of each component stock C_i or index returns IR_{it} of stock index I_i as follows:

$$R_{it}=\frac{C_{i,t}-C_{i,t-1}}{C_{i,t-1}}\times 100 \tag{6}$$

$$\mathrm{IR}_{it}=\frac{I_{i,t}-I_{i,t-1}}{I_{i,t-1}}\times 100 \tag{7}$$

where C_{it} and C_{it-1} are closing prices of the component stocks C_i for day t and day $t-1$, while I_{it} and I_{it-1} are closing prices of the stock index I_i for day t and day $t-1$, respectively.

The stationary transformation method used by Kanas and Yannopoulos (2001) is also applied to obtain information about the volume. That is, the natural logarithm of the change in trading volume that are computed as 100 times are also introduced as inputs into neural networks. One advantage of such a transformation is the stationary property, which will not lose the important information from the trading volume.

Table 1 shows some statistical results of the daily, weekly and monthly time series for three stock markets. In all the cases, say short-, medium-and long terms, NASDAQ appears to be the most volatile index in return, followed by STI, and with DJIA ranking the least volatile in all terms. In contrast, for the trading volume, STI appears to be more volatile than both of the other two in all three different terms. Another finding is that the ranking of the linear correlation coefficient between index return and trading volume among three markets in different terms is totally different. Particularly, the most linear correlated indices between its returns and volumes are STI, NASDAQ and DJIA for daily, weekly and monthly time series data, respectively. We therefore re-affirm the previous research finding regarding the nonlinear relationship between the stock return and trading volumes (Brooks, 1998). In the

Table 1
The preliminary statistics for different training patterns

Preliminary statistics	DJIA		NASDAQ		STI	
	Return	Trading volume	Return	Trading volume	Return	Trading volume
The preliminary statistics of 2000 (daily) training pattern						
Mean	0.03230731	0.000618268	0.062951811	0.000618742	0.00832874	0.002136347
Standard deviation	1.210333649	0.212822186	2.461516225	0.179630696	1.52301723	0.531524798
Skewness	−0.107535029	0.194311024	0.331161149	−0.139349277	0.43462894	−0.300522157
Kurtosis	3.175038675	21.23194229	3.165117248	9.450326561	7.87406859	9.402351168
Correlation Coefficient	−0.003738351		0.011897184		−0.032429049	
The preliminary statistics of 800 (weekly) training pattern						
Mean	0.194449529	0.002427256	0.257312712	0.002910007	0.09873309	0.007370755
Standard deviation	2.181913519	0.14520414	4.191262449	0.13194158	3.08518762	0.519513065
Skewness	−0.482419251	−0.227516161	−2.203397712	−0.27261457	0.0537132	0.035401821
Kurtosis	3.182310283	3.87653066	26.7866022	3.288402894	8.45374492	4.601672459
Correlation coefficient	−0.040437832		−0.064805497		0.00346815	
The preliminary statistics of 200 (monthly) training pattern						
Mean	0.882965631	0.010321025	0.785579115	0.013202461	0.10090482	0.029903179
Standard deviation	4.201305571	0.113609104	8.243736221	0.116603652	6.25211223	0.438514115
Skewness	−0.463403466	0.377519994	−1.094444664	0.046012053	0.61767064	0.164135894
Kurtosis	1.004937238	2.082166814	6.673435145	1.05876258	5.93891276	2.064197669
Correlation coefficient	−0.166573145		0.030352831		0.00389178	

rest of this paper, we further advance this area of research by answering the open question related to predictability of stock index increments using trading volumes.

The data sets with 2000, 800, and 200 training patterns are used in short-term (daily), medium-term (weekly) and long-term (monthly) one-step ahead forecasting, respectively. Each data set is divided into two parts: 90% are used to train neural networks while the latest 10% are employed to test the generalization ability.

3.2. Basic and augmented neural network models

We employ the component-based feed-forward neural networks, first proposed in (Phua et al., 2003), to explore the usefulness of volume information in the explanation of the predictability of stock index returns. Although a variety of different neural network models have been devised in the literature, the feed-forward network is adopted in the present study as it has the most widespread use. Unlike other financial forecasting models, our model directly uses the component stocks of the index as inputs for the prediction. In particular, in our former research, we have shown that this model is able to forecast the sign of the index increments with an average success rate that is statistically significant (Phua et al., 2003). In this study, to select the inputs for the neural network, we choose the component stocks whose correlation coefficient with their corresponding index ranks within the highest 5th. The correlation coefficient, $r_i(I)$ of the component stock C_i, is computed as follows:

$$r_i(I) = \frac{\sum_{t=1}^{N}[(R_{it}-\overline{R_i})(I_t-\overline{I})]}{\sqrt{\sum_{t=1}^{N}(R_{it}-\overline{R_i})^2}\sqrt{\sum_{t=1}^{N}(I_t-\overline{I})^2}} \tag{8}$$

where $\overline{R_i} = \frac{1}{N}\sum_{t=1}^{N} R_{it}$, and $\overline{I} = \frac{1}{N}\sum_{t=1}^{N} I_t$.

Consequently, the basic one-step ahead prediction model for forecasting stock returns in this study can be constructed as follows.

$$R_{\mathrm{Index}}(t) = \Phi(R_{\mathrm{Index}}(t-1), R_1(t-1), R_2(t-1), \ldots, R_{\underline{m}}(t-1)) \tag{9}$$

where, $R_i(t-1)$, $i = 1, 2, \ldots \underline{m}$ is the daily return of the component stock C_{i} computed at day $(t-1)$.

Though former researches have shown that this basic component-based model can achieve significant performance in index increments forecasting (Phua et al., 2003), we believe that the model performance could be further improved by enhancing the input selection scheme. In practice, to build a forecaster of time series, the inputs of the networks are carefully selected to not only reflect the internal movement dynamics of the time series by using its time delay values, but also reveal the environmental impacts and interactions among different effective factors by adding explanatory variables. Particularly, such augmented models for stock index forecasting by adding more explanatory variables can be expressed as:

$$R_{\mathrm{Index}}(t) = \Phi(R_{\mathrm{Index}}(t-1), R_1(t-1), R_2(t-1), \ldots, R_{\underline{m}}(t-1), \mu_{1t}, \mu_{2t}, \ldots, \mu_{mt}) \tag{10}$$

where $\mu_{1t}, \mu_{2t}, \ldots, \mu_{mt}$ are explanatory variables at time t. Additional explanatory variables such as interest rate differentials, book-to-market values, lagged trading activity measures, etc. could also easily be included. On the other hand, this paper only focuses on the consideration of one of the daily trading activity measures: volume. Following the component-based input selection scheme, we not only consider the total market trading volume but also the individual trading volume for each corresponding component stock in the stock market. Total stock market trading volume is simply just the sum of all the individual trading volumes of all corresponding component stocks in the markets. Our former research, which simply adds total trading volume to the basic model, can only improve the forecasting performance weakly or irregularly in short-term forecasting (Wang et al., 2003), and this result may be caused by over-fitting. In other words, by including the information of total market trading volume, some redundant or intercorrelated information may also be included; it thus does not lead to the desired improvements for network forecasting. In considering that some key component stocks' individual trading volumes can represent most of the important information contained in the total market trading volume, and at the same time neglecting some unnecessary redundant information, we augmented the basic model by adding three component stocks' individual trading volumes whose correlation coefficient with the total market trading volume ranks the highest. To facilitate comparison and to choose the best volume input selection scheme, we choose the following four typical models for financial forecasting:

- M1, the basic model without adding any trading volumes as inputs:

$$R_{\text{Index}}(t) = \Phi(R_{\text{Index}}(t-1), R_1(t-1), R_2(t-1), \ldots, R_5(t-1))$$

- M2: augmented model added with total market trading volume V_{Total} as inputs:

$$R_{\text{Index}}(t) = \Phi(R_{\text{Index}}(t-1), R_1(t-1), R_2(t-1), \ldots, R_5(t-1), V_{\text{Total}}(t-1))$$

- M3: augmented model added with three selected individual trading volumes as inputs:

$$R_{\text{Index}}(t) = \Phi(R_{\text{Index}}(t-1), R_1(t-1), R_2(t-1), \ldots, R_5(t-1), V_1(t-1), V_2(t-1), V_3(t-1))$$

- M4: augmented model added with both the selected individual trading volumes and the total market trading volume V_{Total} as inputs:

$$R_{\text{Index}}(t) = \Phi(R_{\text{Index}}(t-1), R_1(t-1), R_2(t-1), \ldots, R_5(t-1), V_{\text{Total}}(t-1), V_1(t-1), V_2(t-1), V_3(t-1))$$

The value of all weights can be numerically estimated in the training procedure which is equivalent to typically solving nonlinear optimization problems. There are scores of numerical computation algorithms designed for training neural networks such as back propagation algorithms, nonlinear least square techniques based algorithms, nonlinear optimization techniques based algorithms, and so on. Due to the complexity and high-degree nonlinearity of the problem, there is no guaranteed way that the global minimum can be reached. To improve the robustness and convergent speed of training algorithms, we employ the Self-Scaling Parallel Quasi-Newton (SSPQN) algorithms to train all the neural networks forecasting models (Phua & Ming, 2003). In comparison to standard serial QN methods, the proposed SSPQN algorithms show significant improvement in the total number of iterations and function/gradient evaluations required in solving a wide range of optimization problems. In fact, the average speedup factors obtained by the new SSPQN algorithms over the conventional quasi-Newton methods are more than 300, both in terms of total number of iterations and total number of function/gradient evaluations required. For some test problems, the speedup factor gained by the new algorithms can be as high as 25 times.

3.3. One-step ahead stock index increments forecasting

In this research, the final estimation of the performance in forecasting is made by means of a one-step sign prediction rate ξ defined on T as follows:

$$\xi = \frac{1}{|T|} \sum_{t \in T} [HS(\Delta C_t \cdot \Delta G_t) + 1 - HS(|\Delta C_t| + |\Delta G_t|)] \tag{11}$$

where $\Delta C_t = C_t - C_{t-1} = C_{t-1} \times R_t$ is the price change at time $t \in T$ and $\Delta G_t = G_t - C_{t-1} = C_{t-1} \times GR_t$ is the guessed price change at the same time step, where GR_t is the guessed return at time t. Note that we assume the availability of the value of C_{t-1} to evaluate $\Delta\ G_t$. HS is a modified Heaviside function $HS(x) = 1$ for $x > 0$ and 0 otherwise. The argument of the summation in (9) gives one if ΔC_t and ΔG_t are non-zero and with same sign, or if ΔC_t and ΔG_t are both zero. Since our model use both the index and component stock returns as network inputs, the sign prediction rate ξ can also be expressed without change in value as follows:

$$\xi = \frac{1}{|T|} \sum_{t \in T} [HS(R_t \cdot GR_t) + 1 - HS(|R_t| + |GR_t|)] \tag{12}$$

In other words, ξ is the probability of a correct guess on the sign of the price increment estimated on T. In fact, the probability of making a correct guess on the sign of the increment seems independent from the magnitude of the increment ΔC itself.

To check and compare the performance of basic network models and our proposed augmented network models, the optimal network topology is applied to perform one-step ahead prediction of three different indices of DJIA, NASDAQ and STI price increments under short-term, medium-term and long-term forecasting horizons.

4. Results and discussions

To enhance the analysis of results, two evaluation criteria of one-step sign prediction rate ξ and mean squared error (MSE) are applied to test the generalization ability and accuracy. Since the forecasts generated by the various models for the same time series are not statistically independent, the comparisons are made using the paired t-test, which ensures robustness under such problems (Nelson, Hill, Temus, & O'Connor, 1999).

The stopping criterion is set by the mean square error (MSE). In order to avoid the effect of initial weights, each training trial is preceded ten times with different initial weights and average results which are presented in the following comparison tables. In the short term experiments, we apply 1800 sample data records as training patterns to build neural networks and the latest 200 sample records as testing patterns to verify the generalization ability. In medium-term experiments, 720 sample data records are used for training and the latest 80 sample data records are for evaluation. In long-term experiments, the two numbers are 180 and 20 in training and testing processes, respectively.

4.1. Convergence rate

In this study, convergence rate is simply measured by the number of iterations for training networks. Tables 2–4 show the convergence rates of the four network models under short-term, medium-term and long-term forecasting horizons, respectively.

First, on the whole, the iteration numbers for each model keep increasing with the number of hidden neurons increasing. Such results show that the more complex the neural networks are, the more time they need to deal with the time series.

Second, in short-term forecasting horizon, it is interesting to find that those augmented models of M2, M3 and M4 converge obviously faster than the basic model of M1 in most cases of the stock forecasting in all three markets. For NASDAQ, an average of 170.6 iterations are needed for training basic network model M1 with 15 hidden neurons, while the augmented network models M2, M3 and M4 just need 139, 121 and 116 epochs, respectively for the same network structure. Though most of the augmented models with volume information outperform the basic model without adding volumes, the distribution of the fastest convergent model is irregular in short-term horizon. In most cases, M4 and M3 converge fastest among all the models. In medium-term forecasting horizon, the advantages of augmented models in convergence speed become less obvious. In this category, in almost half of the cases, basic model converges faster than all the augmented models. And the situations in different markets are very similar. In long-term forecasting horizon, the comparison results are strongly dependent on market divergence. The situations in NASDAQ and STI are very similar. That is, almost all augmented models outperform in convergence rate than the basic model in all cases, and at the same time, model M3 always performs the best among all models. In addition to the basic inputs from the component-based model, M3 includes three selected component stocks' individual trading volumes as inputs for the network forecasting. However, in DJIA forecasting, only when the hidden neuron number is 10 or 15, some of the augmented models outperform the basic model. From the results, we can see that the information of trading volume can affect the convergence rate of neural network forecasting models and this kind of effect is both forecasting horizon and market dependent.

4.2. Generalization ability

In this paper, we use one-step sign prediction rate ξ to test the correct direction forecasting rate of the trained neural networks. One-step sign prediction rate ξ is also used to test the generalization ability of these network models. Tables 5–7 present these results for neural networks that underwent training in short-term, medium-term and long-term horizons, respectively.

First of all, regardless of the forecasting horizon or the forecasting market chosen, the forecasting performance of each model will improved significantly with the increase of the hidden neuron number. A reasonable comment on this result is that with the increasing complexity of the network model, the model enhances its ability to simulate the internal nonlinear relationships between input and output information with enough training. It thus will increase its forecasting ability consistently. But too complex network

Table 2
Average convergence rate in short-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	73.9 (10.3)	72.3 (16.6)	75.7 (17.5)	57.0 (8.5)[a]	84.6 (10.2)	116.2 (13.7)	81.1 (30.6)	57.7 (13.1)[a]	90.3 (10.8)	138.6 (27.0)[b]	114.8 (13.5)	124.4 (15.4)
10	89.2 (11.1)	152.0 (19.7)[b]	118.1 (24.6)	99.4 (8.6)	137.2 (21.2)	148.6 (9.1)	83.5 (15.2)	102.6 (20.3)	229.3 (12.4)	222.8 (13.8)	172.1 (18.6)	188.6 (21.4)
15	195.8 (23.3)	169.4 (10.1)	167.7 (21.3)	130.9 (14.0)[a]	170.6 (9.4)	139.1 (18.3)	121.1 (25.4)	116.6 (19.3)[a]	232.2 (12.8)	199.8 (8.5)	228.9 (12.2)	245.6 (4.2)
20	195.7 (18.6)	170.7 (18.2)	140.8 (14.6)[a]	214.4 (20.3)	157.3 (16.3)	228.9 (19.1)	153.1 (22.0)	155.7 (35.5)	239.9 (8.3)	250.3 (8.6)	244.0 (5.6)	229.0 (6.8)

Average epochs of 10 trails with different initial weights are shown with their standard deviations in parentheses.
[a] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.
[b] M1 outperforms M2 or M3 or M4 in terms of paired t-test for 0.05 significant differences.

Table 3
Average convergence rate in medium-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	40.5 (5.0)	31.5 (2.6)	47.4 (7.0)	41.4 (5.1)	50.1 (6.5)	75.7 (18.5)	62.1 (8.7)	82.6 (22.2)	34.7 (6.2)	38.9 (1.0)	42.3 (6.6)	36.5 (5.1)
10	39.8 (1.8)	57.2 (5.6)	52.4 (6.6)	63.9 (8.5)[a]	84.7 (19.1)	70.3 (18.2)	112.7 (19.6)	108.8 (28.7)[a]	66.5 (9.6)	54.2 (5.1)	56.9 (5.1)	50.0 (4.9)
15	64.5 (6.0)	66.6 (8.4)	55.5 (4.4)	82.5 (9.4)	128.6 (28.3)	144.5 (27.9)	119.5 (14.2)	137.5 (11.9)	77.4 (3.7)	84.6 (14.1)	77.1 (16.0)	76.1 (6.2)
20	98.3 (5.5)	84.9 (9.4)	102.4 (14.3)	82.7 (11.0)	100.3 (6.6)	179.5 (20.1)[a]	161.2 (21.6)[a]	186.2 (15.4)[a]	106.5 (17.7)	90.0 (9.4)	81.0 (8.7)[b]	111.8 (4.4)

Average epochs of 10 trails with different initial weights are shown with their standard deviations in parentheses.
[a] M1 outperforms M2 or M3 or M4 in terms of paired t-test for 0.05 significant differences.
[b] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

Table 4
Average convergence rate in long-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	45.9 (3.9)	76.2 (18.8)	47.0 (6.0)	52.4 (5.7)	81.7 (10.5)	51.0 (2.3)	55.9 (5.1)	51.5 (2.9)	72.1 (6.9)	73.0 (19.9)	59.3 (16.3)	69.9 (15.7)
10	98.9 (26.1)	97.6 (13.9)	125.2 (13.6)	176.6 (15.2)[a]	135.2 (8.7)	97.0 (6.7)	85.5 (7.4)[b]	100.8 (6.1)	136.8 (23.4)	116.3 (17.6)	71.0 (3.3)[b]	85.3 (7.8)
15	187.1 (19.3)	196.8 (19.0)	172.8 (25.2)	148.9 (25.7)	242.2 (7.3)	200.8 (18.0)	149.5 (14.3)	158.0 (10.6)	235.5 (8.2)	127.3 (11.9)	108.4 (7.7)	144.2 (27.0)
20	218.7 (24.8)	247.0 (2.9)	250.3 (2.9)	246.4 (2.9)	248.7 (1.4)	237.2 (6.0)	187.5 (18.6)	198.5 (19.3)	232.0 (16.3)	204.4 (14.2)	151.6 (4.4)	180.8 (18.9)

Average epochs of 10 trails with different initial weights are shown with their standard deviations in parentheses.
[a] M1 outperforms M2 or M3 or M4 in terms of paired t-test for 0.05 significant differences.
[b] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

Table 5
Average training one-step sign prediction rate ξ in short-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	53.75 (0.02)	53.27 (0.02)	54.46 (0.03)	54.78 (0.01)	54.40 (0.03)	55.45 (0.03)	56.27 (0.02)	56.78 (0.02)	55.53 (0.01)	56.32 (0.02)	55.61 (0.02)	57.15 (0.03)
10	54.66 (0.02)	56.95 (0.02)	56.82 (0.02)	57.53 (0.02)	57.94 (0.02)	58.33 (0.01)	58.63 (0.03)	58.94 (0.02)	58.19 (0.03)	57.91 (0.02)	58.91 (0.02)	59.40 (0.03)
15	55.50 (0.02)	58.95 (0.02)	59.77 (0.03)	60.59 (0.02)[a]	59.59 (0.04)	59.50 (0.03)	61.06 (0.02)	62.06 (0.02)[a]	60.30 (0.02)	59.48 (0.02)	61.51 (0.02)	60.10 (0.02)
20	58.89 (0.02)	60.58 (0.01)	61.39 (0.02)[a]	62.31 (0.02)[a]	60.79 (0.02)	61.18 (0.01)	63.01 (0.02)	63.46 (0.02)[a]	61.17 (0.01)	57.19 (0.02)	64.68 (0.02)	64.73 (0.03)[a]

Average one-step sign prediction rate ξ (% percentage) of 10 different initial weights are shown in the tables. Standard deviations are shown in the parenthesis.
[a] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

models or over-training can cause the phenomenon of over-fitting, which will produce excellent results in training process but lose its generalization ability in out-of-sample testing process.

Second, in the training process, almost all the augmented network models M2, M3, and M4 outperform the basic model M1 under the evaluation of one-step sign prediction rate. At the same time, model M4, augmented with both total market and selected component stock's trading volumes as inputs, performs the best with significance among all the models in almost all cases. It indicates that in the training process, the more complex the neural

Table 6
Average Training one-step sign prediction rate ξ in medium-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	58.78	59.85	59.70	60.00	64.15	63.53	63.57	64.85	61.45	62.88	63.59	63.61
	(0.03)	(0.01)	(0.02)	(0.01)	(0.02)	(0.01)	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.01)
10	61.54	61.65	63.39	63.87	65.90	66.03	67.80	68.09	63.87	64.75	66.37	67.20
	(0.02)	(0.02)	(0.01)[a]	(0.02)[a]	(0.05)	(0.01)	(0.03)	(0.02)	(0.02)	(0.03)	(0.02)	(0.01)
15	64.49	65.54	65.52	66.02	68.37	67.52	71.35	71.55	67.01	65.97	67.93	69.68
	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.03)	(0.02)	(0.02)	(0.02)	(0.01)	(0.02)	(0.02)
20	67.65	67.56	69.39	68.74	69.98	70.62	75.19	75.40	68.41	69.28	71.47	72.82
	(0.02)	(0.01)	(0.03)[a]	(0.02)	(0.01)	(0.01)	(0.02)[a]	(0.02)[a]	(0.01)	(0.02)	(0.02)	(0.03)[a]

Average one-step sign prediction rate ξ (% percentage) of 10 different initial weights are shown in the tables. Standard deviations are shown in the parenthesis.
[a] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

Table 7
Average training one-step sign prediction Rate ξ in long-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	70.33	68.93	68.02	71.30	75.88	75.63	76.29	78.41	76.76	78.24	80.65	81.48
	(0.01)	(0.01)	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.04)	(0.01)	(0.02)	(0.02)	(0.01)[a]
10	74.97	75.17	75.21	77.25	80.23	76.95	80.76	81.92	78.95	82.09	84.69	86.63
	(0.02)	(0.01)	(0.02)	(0.01)	(0.02)	(0.02)[b]	(0.02)	(0.02)	(0.03)	(0.02)	(0.02)	(0.01)[a]
15	77.57	78.71	78.40	78.37	83.47	79.75	83.14	84.41	81.39	85.29	86.89	90.03
	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)	(0.01)[b]	(0.02)	(0.02)	(0.02)	(0.02)	(0.02)[a]	(0.01)[a]
20	76.57	77.81	72.45	79.26	81.98	81.89	88.37	86.78	79.29	85.73	89.44	93.00
	(0.02)	(0.03)	(0.02)	(0.02)[a]	(0.01)	(0.01)	(0.02)	(0.02)[a]	(0.01)	(0.02)[a]	(0.02)[a]	(0.02)[a]

Average one-step sign prediction rate ξ (% percentage) of 10 different initial weights are shown in the tables. Standard deviations are shown in the parenthesis.
[a] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.
[b] M1 outperforms M2 or M3 or M4 in terms of paired t-test for 0.05 significant differences.

network structure is or the more trading volume information is used as the inputs to train the network, the better the training or fitting results. We observe that M4 can greatly improve the forecasting performance for training and these improvements are statistically significant. For STI, when the number of hidden neurons is 20, the one-step sign prediction rate for M4 is 72.824%, which is significantly greater than 68.412% of M1. (P-value is less than 0.001). At the same time, M2 and M3 also outperform M1 in almost all cases. M2 only augments the total market trading volume as inputs, while M3 only augments the selected component stock's individual trading volumes as inputs. It can be concluded that trading volume can significantly improve the forecasting performance, and the more variety of trading volume information used, the better the network training performance in forecasting.

However, for testing, the results are not so consistent. Tables 8–10 present these results for testing process in short-term, medium-term and long-term horizons, respectively. Under short-term horizon, the trading volume does not improve the forecasting performance in testing as it does in training process. Though all augmented models still outperform the basic models for DJIA index prediction, such improvement becomes quite modest as p-values are all larger than 0.2. Furthermore, extremely different results are obtained from NASDAQ and STI indices testing process. In some cases, the testing results taken from basic model M1, significantly outperform all those augmented models (P-value is less than 0.001). In comparing M1 with augmented models, M1 performs at least as good as, if not better than, those models. On the other hand, the distribution of the best model in the aspect of testing performance is totally irregular under short-term horizon. Empirical results in this study indicate that trading volume leads to irregular improvements on the performance of stock index increments forecasting under short-term horizon.

Under medium-term forecasting horizon, basic models are always outperformed by some particular augmented models in almost all cases. The best performance models under this horizon are mainly M2 or M3. That is, in order to improve the direction forecasting performance under weekly or medium-term forecasting horizon, one should add in either total market trading volume (like M2) or selected component stock's individual trading volumes (like M3), instead of adding them together (like M4) to make a forecaster.

Table 8
Average testing one-step sign prediction rate ξ in short-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	48.65 (0.05)	49.66 (0.07)	48.99 (0.03)	52.12 (0.06)	51.82 (0.09)	51.61 (0.04)	51.42 (0.07)	52.19 (0.05)	49.44 (0.04)	50.32 (0.05)	50.26 (0.06)	47.66 (0.03)
10	48.15 (0.06)	51.26 (0.04)	49.32 (0.05)	50.99 (0.04)	51.02 (0.03)	53.19 (0.02)	51.88 (0.06)	52.85 (0.04)	52.71 (0.03)	50.62 (0.04)	49.78 (0.07)[a]	50.46 (0.03)
15	48.79 (0.03)	50.80 (0.04)	50.93 (0.03)	52.21 (0.02)	52.63 (0.03)	49.39 (0.05)[a]	51.18 (0.08)	50.61 (0.03)	48.24 (0.06)	51.33 (0.06)	51.93 (0.02)[b]	46.85 (0.02)
20	49.39 (0.02)	50.11 (0.08)	51.41 (0.04)	50.72 (0.03)	50.65 (0.05)	48.99 (0.03)	53.55 (0.04)	53.69 (0.01)[b]	52.07 (0.03)	45.12 (0.05)[a]	50.85 (0.07)	48.38 (0.04)

Average one-step sign prediction rate ξ (% percentage) of 10 different initial weights are shown in the tables. Standard deviations are shown in parentheses.
[a] M1 outperforms M2 or M3 or M4 in terms of paired t-test for 0.05 significant differences.
[b] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

Table 9
Average testing one-step sign prediction rate ξ in medium-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	51.55 (0.08)	55.12 (0.04)	54.89 (0.02)	57.22 (0.06)[a]	50.14 (0.03)	49.65 (0.05)	50.41 (0.07)	50.75 (0.08)	54.37 (0.06)	53.16 (0.06)	55.77 (0.06)	52.76 (0.02)
10	51.77 (0.03)	48.52 (0.06)	51.83 (0.04)	51.57 (0.07)	50.89 (0.05)	50.44 (0.06)	52.94 (0.03)	50.72 (0.06)	53.98 (0.03)	52.88 (0.05)	54.90 (0.02)	53.48 (0.07)
15	49.88 (0.07)	50.25 (0.06)	50.11 (0.06)	47.92 (0.04)	49.73 (0.02)	49.97 (0.02)	48.03 (0.08)	49.19 (0.06)	53.18 (0.02)	56.18 (0.08)	51.39 (0.06)	54.51 (0.04)
20	50.28 (0.03)	50.30 (0.05)	51.06 (0.05)	47.17 (0.05)	53.11 (0.02)	52.35 (0.07)	54.57 (0.02)	48.67 (0.08)	54.21 (0.07)	55.60 (0.07)	50.23 (0.03)	50.26 (0.03)

Average one-step sign prediction rate ξ (% percentage) of 10 different initial weights are shown in the tables. Standard deviations are shown in parentheses.
[a] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

Table 10
Average testing one-step sign prediction rate ξ in long-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	57.67 (0.06)	59.53 (0.07)	58.81 (0.04)	51.99 (0.03)[a]	64.73 (0.03)	59.72 (0.04)	66.27 (0.04)	63.78 (0.06)	56.14 (0.08)	56.81 (0.07)	69.19 (0.03)[b]	56.11 (0.02)
10	54.14 (0.05)	57.24 (0.03)	57.83 (0.05)	48.40 (0.03)	62.66 (0.04)	59.72 (0.07)	63.28 (0.07)	60.82 (0.04)	58.61 (0.03)	58.28 (0.06)	57.36 (0.06)	50.27 (0.08)[a]

A very interesting finding has been discovered by the experiments under long-term forecasting horizon. Aside from when the hidden neuron number is 5 in DJIA and when the hidden neuron number is 10 in STI, M3 always performs best among all models in out-of-sample index direction forecasting. Furthermore, in more than half of these outperforming cases, M3 significantly improves the correct direction prediction rate compared to the basic model in that P-value is less than 0.001. This result confirms the former literature describing that inclusion of nonlinear terms in the relation between stock returns and fundamentals indicators like trading volume can improve the long-term (monthly return) forecasting.

Conclusively, although all augmented network models with trading volumes can significantly improve the one-step sign prediction results in training process, such improvement is inconsistent in out-of-sample testing process. The improvement that trading volume gains apparently depends on the forecasting horizon and, at the same time, is a little affected by the market diversity. Under short-term horizon, we use daily data which is noisier and has high volatility. Thus adding more information in the inputs for the network training may lead to overfitting and may decrease the generalization ability of the network. Experiments under short-term forecasting indicate that improvements, if there are any, made by adding trading volume is irregular and not obvious. Most likely, it may incline to overfitting instead of improving the forecasting performance, especially in NASDAQ and STI markets. Refenes et al. (1997) state that when adding more fundamental factors into neural networks to produce better correlation results with observed values, the network is facing the risk of overfitting. But, by experiments under medium-term and long-term forecasting horizons, trading volume has been shown to be able to at least modestly improve the direction prediction, if the selection scheme of trading volumes is properly considered.

4.3. Accuracy

In order to draw more reasonable conclusions, another performance evaluation criterion, Mean Squared Error (MSE), is also employed, due to its popularity and its evaluation on the forecasting results accuracy. Similar to the condition under one-step sign prediction rate, augmented models with trading volumes all significantly outperform the basic model without trading volumes measured in MSE in training process. The dominant model in aspect of performance in training MSE is also model M4.

Tables 11–13 present the MSE testing results under short-term, medium-term and long-term horizons. One of the most interesting findings is that the results of MSE bear close similarity with those from direction forecasting. Under short-term horizon, the results are dynamic: in most cases M1 is better than all augmented models or M2, M3, and M4, while at other times some particular augmented model outperforms M1. Because M1 outperforms the other models with statistical significance in many cases, on the whole, we can say that adding trading volume cannot help in network performance improvements measured under MSE in short-term forecasting. Under medium-term horizon, augmented models outperform the basic model in most cases but only make statistically significant improvements in STI market. Particularly, in STI market, M3 dominantly outperforms other models significantly. However, such MSE advantages of augmented models are not significant in DJIA and NASDAQ markets (P-values are always larger than 0.25). In another words, the results of these four models are almost the same in DJIA and NASDAQ. A very interesting finding under long term horizon is that the M2 model dominates the best performance in MSE results in most cases. And many of these improvements by M2 are statistically significant. Thus we can say that by adding total stock market trading volume, we can improve the accuracy of network forecasting model.

4.4. Discussion

The above analysis indicates that the extent in which trading volume can improve the direction forecasting accuracy depends on the forecasting horizon and at the same time is slightly affected by the market diversity. Experiments show that trading volume cannot significantly improve the direction forecasting performance in short-term horizon. Under medium-term horizon, trading volume could modestly improve the network performance in direction forecasting if the volume selection scheme is properly considered. Our experiment results show that either total market trading volume or selected component stocks' individual trading volumes could be considered to be augmented into the basic model in order to enhance the network performance in direction prediction. Particularly, for the long-term horizon prediction, experiments show that the augmented model M3 gives out the best prediction in all three markets. Thus adding three component

Table 11
Average testing mean squared error (MSE) in short-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	0.70 (0.26)	0.56 (0.22)	0.59 (0.24)	1.17 (0.25)[a]	1.64 (0.06)	1.76 (0.63)	1.66 (1.23)	2.17 (0.32)[a]	0.74 (0.03)	0.77 (0.01)	0.92 (0.23)	0.86 (0.12)
10	0.59 (0.19)	0.61 (0.23)	1.19 (0.28)[a]	0.67 (0.31)	1.96 (1.66)	1.82 (1.32)	3.84 (0.98)[a]	2.23 (1.42)	0.78 (0.44)	0.84 (0.15)	1.06 (0.23)	1.52 (0.24)[a]
15	0.65 (0.18)	1.60 (0.23)[a]	1.07 (0.20)	0.77 (0.27)	3.41 (3.21)	2.47 (1.11)[b]	2.58 (0.78)	2.74 (1.45)	1.11 (0.23)	1.43 (0.24)	0.97 (0.39)	2.04 (0.13)[a]
20	0.82 (0.24)	1.00 (0.25)	0.86 (0.27)	0.88 (0.19)	2.58 (0.86)	2.64 (1.23)	2.97 (1.76)	2.82 (1.23)	0.92 (0.38)	1.07 (0.43)	2.03 (0.23)	3.03 (0.21)[a]

Average mean squared error (MSE) of 10 different initial weights are shown in the tables. Standard deviations are shown in parentheses.
[a] Shows that M1 outperforms M2 or M3 or M4 in terms of paired t-test for 0.05 significant differences.
[b] M2 or M3 or M4 outperforms M1 in terms of paired t-test for 0.05 significances.

Table 12
Average testing mean squared error (MSE) in medium-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	2.61	2.42	2.39	2.81	9.99	14.83	11.30	9.67	5.26	5.12	5.56	5.53
	(0.34)	(0.31)	(0.29)	(0.23)	(1.24)	(2.3)[a]	(3.24)	(1.34)	(0.15)	(0.24)	(0.18)	(0.32)
10	3.02	3.05	2.85	3.07	10.94	10.83	11.60	12.10	6.18	6.08	6.82	6.61
	(0.27)	(0.33)	(0.32)	(0.41)	(1.9)	(0.66)	(0.89)	(1.28)	(0.52)	(0.32)[b]	(0.19)	(0.27)
15	3.32	10.06	3.49	3.78	15.97	14.67	17.36	16.31	7.61	6.86	7.79	9.06
	(0.12)	(0.32)[a]	(0.34)	(0.35)	(1.02)	(0.99)	(1.35)	(3.49)	(0.74)	(0.64)[b]	(0.11)	(0/32)
20	5.08	4.29	4.02	4.27	15.27	16.71	20.23	21.89	8.30	7.51	10.01	10.42
	(0.53)	(0.35)	(0.39)[b]	(0.42)	(1.34)	(1.39)	(3.42)	(6.81)[a]	(0.66)	(0.71)[b]	(0.36)	(0.41)[a]

Average mean squared error (MSE) of 10 different initial weights are shown in the tables. Standard deviations are shown in parentheses.
[a] Shows that M1 outperforms M2 or M3 or M4 in terms of paired *t*-test for 0.05 significant differences.
[b] M2 or M3 or M4 outperforms M1 in terms of paired *t*-test for 0.05 significances.

Table 13
Average testing mean squared error (MSE) in long-term forecasting

NNs models	DJIA				NASDAQ				STI			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
5	17.64	12.07	13.24	12.65	29.05	22.84	34.20	36.21	19.78	16.53	22.22	16.98
	(1.71)	(2.01)[a]	(1.89)	(1.4)	(5.3)	(6.72)[a]	(4.55)[b]	(9.71)[b]	(3.21)	(1.02)[a]	(4.12)	(2.45)
10	16.72	15.80	17.20	20.69	47.64	48.73	54.29	43.70	25.93	26.54	44.16	64.62
	(2.31)	(2.53)	(1.94)	(2.03)	(4.78)	(2.13)	(3.12)	(8.84)	(7.23)	(4.45)	(3.78)[b]	(1.11)[b]
15	26.03	17.12	19.86	26.57	44.38	43.17	54.33	62.08	75.08	39.48	51.27	45.46
	(2.22)	(2.98)[a]	(3.04)	(1.2)	(4.5)	(5.67)	(7.13)	(1.38)[b]	(10.1)	(9.31)[a]	(7.23)	(3.21)
20	50.60	16.05	15.28	30.02	88.29	60.85	90.48	61.44	27.94	17.80	56.81	59.09
	(0.98)	(3.4)[a]	(3.2)	(6.4)	(2.33)	(4.57)[a]	(7.8)	(9.1)[a]	(11.3)	(4.5)[a]	(7.89)[b]	(9.34)[b]

Average mean squared error (MSE) of 10 different initial weights are shown in the tables. Standard deviations are shown in parentheses.
[a] M2 or M3 or M4 outperforms M1 in terms of paired *t*-test for 0.05 significances.
[b] Shows that M1 outperforms M2 or M3 or M4 in terms of paired *t*-test for 0.05 significant differences.

stocks' individual trading volumes whose correlation coefficient with the total market trading volume ranks within the highest 3 as additional inputs of the basic component-based model could apparently improve the network direction forecasting.

Former researches indicate that causality nonlinearity is found from trading volume with stock return. In order to improve the network forecasting performance in a significant way, proper use of the inherent information or logics of trading volume is important; furthermore, filtration of redundant or noisy information is crucial. An appropriate input selection scheme is the key point in using the information in trading volume when applied to financial forecasting.

5. Conclusions

This study employs a component-based three-layer feedforward neural network as the basic model to make one-step ahead stock index increments forecasting. By adding trading volumes under different input selection schemes to the basic model, we set up three augmented network models to test whether trading volume can significantly improve the forecasting performance in financial markets. Contrary to theories and empirical based-exception, the results show that trading volume can not significantly improve the forecasting performance under short-term horizon. This may be due to the high noise level and high volatility of the daily data which attenuate nonlinearity between the stock and trading volume. This study also shows it is possible to modestly or significantly improve the network performance by adding trading volume under medium-term or long-term forecasting, if an appropriate input selection scheme is carefully considered. Significant improvement could be achieved by including only the selected component trading volumes to the basic model in the long-term forecasting.

These findings have a number of important implications for future research in this area. The most interesting one is how to use the information in trading volume to improve the forecasting results, especially when forecasting horizons are different. In this paper, either total market trading volume, individual component stock's trading volume or their combination could be considered to considerably improve the performance, compared with those without trading volume. One possible way to further improve the network performance is to combine trading volume and other fundamental factors in neural network models, based on some particular designed input selection schemes. An alternative is to apply other kinds of network models as the basic model, such as time delay neural networks (TDNN), recurrent networks or probability neural networks. This way, the augmented model design can be based on other kinds of basic models by adding trading volume schemes to the basic model. A third consideration is to use the time delayed trading volume with different looking back window sizes.

References

Abhyankar, A., Copeland, L. S., & Wang, W. (1997). Uncovering nonlinear structure in real time stock market indexes: The S& P 500, the DAX, the Nikkei 225, and the FTSE-100. *Journal of Business & Economic Statistics, 15*(1), 1–14.

Brooks, C. (1998). Predicting stock index volatility: Can Market Volume help? *Journal of Forecasting, 17*, 59–98.

Campbell, J. Y., Grossman, S. J., & Wang, J. (1993). Trading volume and serial correlation in stock returns. *The Quarterly Journal of Economics, 108*, 905–936.

Castiglione, F. (2001). Forecasting price increments using an artificial neural network. *Advances in Complex Systems, 4*(1), 45–56.

Coakley, J. R., & Brown, C. E. (2000). Artificial neural networks in accounting and finance: Modeling issues. *International Journal of Intelligent Systems in Accounting, Finance & Management, 9*, 119–144.

Freisleben, B. (1992). Stock market prediction with back propagation networks, *Proceedings of the 5th international conference on industrial and engineering application of artificial intelligence and expert system* (pp. 451–460).

Gallant, R., Rossi, P. E., & Tauchen, G. (1992). Stock prices and volume. *Review of Financial Studies, 5*, 199–242.

Hiemstra, C., & Jones, J. D. (1994). Testing for linear and nonlinear Granger causality in the stock price-volume relation. *The Journal of Finance, 49*, 1639–1664.

Kanas, A., & Yannopoulos, A. (2001). Comparing linear and nonlinear forecasts for stock return. *International Review of Economics and Finance, 10*, 383–398.

Kim, S. H., & Chun, S. H. (1998). Graded forecasting using an array of bipolar predictions: application of probabilistic neural networks to a stock market index. *International Journal of Forecasting, 14*, 323–337.

Lapedes, A. & Farber, R. (1987). Nonlinear signal processing using neural networks. *Proceedings of the IEEE conference on neural information processing system-natural and synthetic.*

Liu, Y., & Yao, X. (2001). Evolving neural networks for Hang Seng stock index forecast. *Proceedings of the 2001 Congress on Evolutionary Computation, 1*, 256–260.

Nelson, M., Hill, T., Temus, W., & O'Connor, M. (1999). Time series forecasting using neural networks: Should the data be deseasonalized first? *Journal of Forecasting, 18*, 359–367.

Phua, P. K. H., & Ming, D. (2003). Parallel nonlinear optimization techniques for training neural networks. *IEEE Transactions on Neural Networks, 14*(6), 1460–1468.

Phua, P. K. H., Zhu, X., & Koh, C. (2003). Forecasting stock index increments using neural networks with trust region methods. *Proceedings of IEEE International Joint Conference on Neural Networks, 1*, 260–265.

Refenes, A. P. N., Burgess, A. N., & Bentz, Y. (1997). Neural networks in financial engineering: a study in methodology. *IEEE Transactions on Neural Networks, 8*(6), 1222–1267.

Refenes, A. N., Zapranis, A., & Francies, G. (1994). Stock performance modeling using neural networks: a comparative study with regression models. *Neural Networks, 5*, 961–970.

Resta, M. (2000). Towards an artificial technical analysis of financial markets. *Proceedings of the IEEE–INNS–ENNS international joint conference on neural networks, 5*, 117–122.

Sitte, R., & Sitte, J. (2000). Analysis of the predictive ability of time delay neural networks applied to the S& P 500 time series. *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Review, 30*(4), 568–572.

Tino, P., Schittenkopf, C., & Dorffner, G. (2001). Financial volatility trading using recurrent neural networks. *IEEE Transactions on Neural Networks, 12*(4), 865–874.

Wang, H. (2005). Flexible flow shop scheduling: Optimum, heuristics, and AI solutions. *Expert Systems, 22*(2), 78–85.

Wang, H., Jacob, V., & Rolland, E. (2003). Design of efficient hybrid neural networks for flexible flow shop scheduling. *Expert Systems, 20*(4), 208–231.

Wang, X., Phua, P. H. K., & Lin, W. (2003). Stock market prediction using neural networks: Does trading volume help in short-term prediction? *Proceedings of IEEE International Joint Conference on Neural Networks, 4*, 2438–2442.

Yao, J. T., & Poh, H.-L. (1996). Equity forecasting: a case study on the KLSE index. In A.-P. N. Refenes, Y. Abu-Mostafa, J. Moody, & A. Weigend (Eds.). *Neural networks in financial engineering, proceedings of the 3rd international conference on neural networks in the capital markets* (pp. 341–353). World Scientific.

Yao, J. T., & Tan, C. L. (2000). Time dependent directional profit model for financial time series forecasting. *Proceedings of the IEEE–INNS–ENNS international joint conference on neural networks, 5*, 291–296.

Yao, J. T., Tan, C. L., & Poh, H. L. (1999). Neural networks for technical analysis: A study on KLCI, International. *Journal of Theoretical and Applied Finance, 2*(2), 221–241.