2012 International Symposium on Computer, Consumer and Control

Computational Intelligence Approaches for Stock Price Forecasting

Jui-Yu Wu*
Dept. of Business Administration
Lunghwa University of Science and Technology
Taoyuan, Taiwan
jywu@mail.lhu.edu.tw

Chi-Jie Lu
Dept. of Industrial Engineering and Management
Ching Yun University
Jung-Li, Taiwan
jerrylu@cyu.edu.tw

***Abstract*—Computational intelligence (CI) approaches such as neural networks (NNs) and neuro-fuzzy approaches have been used for stock price forecasting. Robust and efficient stock market models can achieve more accurate predictions and decision making for individual investors or stock fund managers. This work thus surveys individual and hybrid CI methods, including a self-organizing polynomial neural network (SOPNN) based on statistical learning algorithm, cerebellar model articulation controller NN, standard back propagation NN (BPNN) with the steepest descent method (BPNN-GD), BPNN with scaled conjugate gradient (SCG) method, artificial immune algorithm-based BPNN (AIA-BPNN), advanced simulated annealing-based BPNN (ASA-BPNN) and adaptive network based fuzzy inference system (ANFIS) method. The performances of these methods are evaluated by using the Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) dataset collected from the Taipei Stock Exchange, and root mean square error (*RMSE*), mean absolute difference (*MAD*) and mean absolute percent error (*MAPE*) are used as the performance indices. Experimental results show that the best SOPNN, CMAC NN, BPNN-SCG, AIA-BPNN, ASA-BPNN and ANFIS obtain identical training and test accuracies. Particularly, hybrid CI approaches such as AIA-BPNN and ASA-BPNN are recommended for stock price forecasting, since these methods have the lowest test *RMSE*, *MAD* and *MAPE*.**

Keywords-computational intelligence; neural networks; stock price forecasting; back-propagation neural network

I. INTRODUCTION

Stock price prediction can be considered a challenging in financial time series forecasting. The financial time series are essentially dynamic, non-linear, nonparametric, and chaotic in nature [1]. Successful stock market prediction achieves optimal results based on minimal input data and the simplest stock market model [2]. Statistical and spectral analytic approaches are traditional techniques for forecasting financial time series and have some limitations. For instance, linear time series models such as ARIMA have difficulty for solving nonlinear time series problems. Non-linear time series models (such as the generalized autoregressive conditional heteroscedastic model) and spectral analytical methods are difficult for general practitioners to apply [3].

Computational intelligence (CI) is an emerging paradigm of information processing that focuses on intelligent systems design [4]. The CI family can be divided into four branches, i.e. granular computing, neuro-computing, evolutionary computing and artificial life [5]. The individual CI method has been used for solving financial time series problems, such as cerebellar model articulation controller NN (CMAC NN) [6], back-propagation NN (BPNN), probabilistic NNs and radial basis function network (RBFN) [2].

Zadeh [7] coined the term "soft computing", which denotes the synergistic power of two or more fused CI schemes. Therefore, many hybrid CI methods have been developed for forecasting financial time series. For instance, Hsieh et al. [8] presented a hybrid wavelet and artificial bee colony-recurrent NN based forecasting scheme and applied it to forecast stock markets. Furthermore, Chang et al. [9] developed a hybrid adaptive network based fuzzy inference system (ANFIS) model based on autoregressive model and volatility for forecasting the Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX). Finally, Shen et al. [10] developed a hybrid RBFN and artificial fish swarm algorithm for stock index forecasting.

Robust and efficient stock market models can achieve more accurate predictions and decision making for individual investors or stock fund managers. Therefore, this work surveys many individual and hybrid CI methods for time series forecasting, such as the self-organizing polynomial neural NN (SOPNN) [3], CMAC NN [6], artificial immune algorithm-based BPNN (AIA-BPNN) [11], advanced simulated annealing-based BPNN (ASA-BPNN) [12], ANFIS [13], BPNN with scaled conjugate gradient learning algorithm (BPNN-SCG) [14] and BPNN with the steepest descent method (BPNN-GD). The performances of these methods are compared using the TAIEX dataset collected from the Taipei Stock Exchange.

II. RELATED WORKS

A. SOPNN

The SOPNN scheme based on the group method of data handling algorithm is developed [3]. The SOPNN creates an optimal network topology of the model through successive generations of neurons regarded as quadratic regression polynomials with two input variables. Only strong neurons can survive during an evolutionary process. By using a deterministic learning algorithm, the proposed SOPNN scheme can obtain unique training and test accuracies by using a single execution. The implementation of the

978-0-7695-4655-1/12 $26.00 © 2012 IEEE
DOI 10.1109/IS3C.2012.23

SOPNN scheme has to set the following parameter: predefined threshold value in layer k (σ_k).

B. CMAC NN

The CMAC NN [6] scheme for forecasting stock index closing price has the advantages of very fast learning and reasonable generalization ability. Moreover, the performance of the CMAC NN is superior to support vector regression and BPNN-GD. The CMAC NN has three parameters, including quantization size ($N_i, i = 1, 2, ..., n_{\max}$), generalization size ($g$) and learning rate ($\beta$).

C. BPNN-SCG

The standard BPNN-GD uses a local optimization algorithm to update weights. Therefore, the conventional BPNN-GD often reaches a local optimum when solving complex problems. Moreover, standard BPNN-GD has a poor convergence rate and depends on user specified parameters. The SCG algorithm [14] was developed to overcome the above limitations. The SCG algorithm does not contain any user-dependent parameters whose values are crucial for the SCG success, and avoids a time consuming line search per learning iteration. The parameter of BPNN-SCG includes the number of neurons in a hidden layer ($neuro\_num$).

D. ANFIS

Jang [13] developed ANFIS, which integrates NN and the Takagi-Suguno fuzzy inference system. Moreover, ANFIS uses a hybrid learning algorithm, which integrates BP and the least square estimation algorithm. Therefore, ANFIS learns rapidly. The parameter of ANFIS includes the number of membership function (MF_num) for each input variable.

E. AIA-BPNN

The weight optimization of a BPNN can be considered a high dimensional and unconstrained optimization problem. The AIA-BPNN [11] uses an AIA to optimize weights in the network topology of a BPNN. The AIA is a stochastic global optimization method, and has been used to solve constrained global optimization [15]. The parameters of the AIA-BPNN have repertoire (population) size rs and the threshold degree of antibodies-antibodies **(Ab-Ab)** recognition p_{rt}, $neuro\_num$ and the lower and upper boundaries of weights and biases.

F. ASA-BPNN

Wu [12] developed an advanced simulated annealing-based BPNN (ASA-BPNN) for forecasting chaotic time series. The ASA-BPNN employs an ASA algorithm to optimize the weights in network topology of a BPNN. The ASA algorithm is a stochastic global optimization method. The parameters of the ASA-BPNN have initial generating temperature of decision variable x_n ($T_{g,n}(0)$), initial acceptance temperature $T_a(0)$, user-defined parameters (constants) $z_{g,n}$, user-defined parameter (constant) z_a, $neuro\_num$ and the lower and upper boundaries of weights and biases.

III. REAL-WORLD DATASET

This work collected daily TAIEX data between September 17, 2007 and September 29, 2011 from the Taipei stock exchange as the dataset. Figure 1 plots daily TAIEX closing prices. The figure shows that the dataset contains a total 1000 data point, and the time series is non-linear.

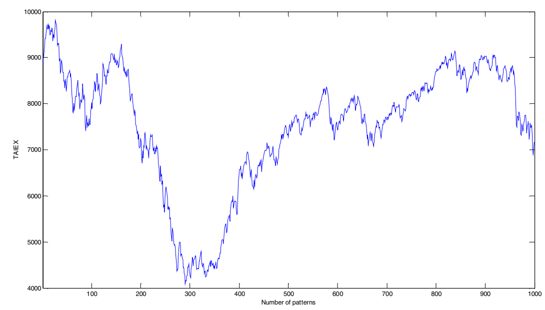

Figure 1. The daily TAIEX closing prices from September 17, 2007 to September 29, 2011

Four measurements, namely the open price at time $t-1$ $o(t-1)$, the highest price at time $t-1$ $h(t-1)$, the lowest price at time $t-1$ $l(t-1)$ and close price at time $t-1$ $c(t-1)$ are used to predict close price at time t $c(t)$. The dataset was then divided into two subsets, i.e. a training dataset with 900 input patterns and a test dataset with 100 input patterns.

The close price $c_j(t)$ ($j = 1, 2, \ldots, n_{\text{total}}$) are desired output values $y_{d,j}$. They undergo a normalization process, as follows:

$$y'_{d,j} = \frac{y_{d,j} - y_d^{\min}}{y_d^{\max} - y_d^{\min}}(E_{\max} - E_{\min}) + E_{\min}, \quad j = 1, 2, \ldots, n_{\text{total}} \quad (6)$$

where

$y'_{d,j}$ = normalized value j

$\mathbf{Y}_d = [y_{d,1}, y_{d,1}, \ldots y_{d,n_{\text{total}}}]^T$, desired output vector

$y_d^{\min}$ = minimum value of $\mathbf{Y}_d$

$y_d^{\max}$ = maximum value of $\mathbf{Y}_d$

$E_{\min}$ = minimum value of expected output

$E_{\max}$ = maximum value of expected output

n_{total} = total number of data points

According to the literature [16], the values [$E_{\min}$, $E_{\max}$] are generally set to [0.2, 0.8].

IV. RESULTS

The SOPNN, CMAC NN, BPNN-GD, BPNN-SCG, ANFIS, AIA-BPNN ASA-BPNN were executed on MATLAB software and run on a Pentium D 3.0 (GHz) personal computer. The TAIEX dataset is used to evaluate the performance of the above approaches. Three indices, root mean square error (*RMSE*), mean absolute difference (*MAD*) and mean absolute percent error (*MAPE*), are used to measure forecasting accuracy as follows:

$$RMSE = \sqrt{\frac{\sum_{j=1}^{n_{\text{total}}} (y_{d,j} - y_{o,j})^2}{n_{\text{total}}}}, \; j = 1, 2, ..., n_{\text{total}} \quad (7)$$

$$MAD = \frac{\sum_{j=1}^{n_{\text{total}}} \left| y_{d,j} - y_{o,j} \right|}{n_{\text{total}}}, \; j = 1, 2, ..., n_{\text{total}} \quad (8)$$

$$MAPE = \frac{\sum_{j=1}^{n_{\text{total}}} \left| \frac{y_{d,j} - y_{o,j}}{y_{d,j}} \right|}{n_{\text{total}}} \times 100\%, \; j = 1, 2, ..., n_{\text{total}} \quad (9)$$

where

$y_{o,j}$ = the forecasting value j

The CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN and ASA-BPNN conducted 30 independent runs under each parameter setting and summarized the mean training *RMSE*s, mean test *RMSE*s, mean *MAD*s and mean *MAPE*s and mean computational CPU times (MCCTs). The SOPNN and ANFIS can obtain unique training *RMSE*, test *RMSE*, *MAD* and *MAPE* through a single execution. Table I lists the parameter settings for the SOPNN, CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN, ASA-BPNN and ANFIS. Table II summarizes the best mean training *RMSE*s, the best mean test *RMSE*s, the best mean *MAD*s, the best mean *MAPE*s and MCCTs obtained from CMAC NN with (N_i=30, g = 90 and β = 0.005), BPNN-GD with ($neuro\_num$ = 5 and β=0.1), BPNN-SCG with ($neuro\_num$ = 5), AIA-BPNN with ($neuro\_num$ = 3 and rs = 24) and ASA-BPNN with ($neuro\_num$ = 3). Analysis of variance (ANOVA) is also performed. According to Table II, the mean test *RMSE*s, mean *MAD*s and mean *MAPE*s obtained using CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN and ASA-BPNN are statistically significant difference at the $P < 0.05$. Therefore, Fisher's least significant difference (LSD) is performed for pairwise comparison of the means. Moreover, the P value = 0.07 is very close to the significant level 0.05 for the significant test of mean training *RMSE*s of CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN and ASA-BPNN. Fisher's LSD is also performed.

Based on Fisher's LSD, the mean training *RMSE*, mean test *RMSE*, mean *MAD* and mean *MAPE* obtained using the BPNN-GD are worse than those obtained using CMAC NN, BPNN-SCG, AIA-BPNN and ASA-BPNN. Moreover, the mean training *RMSE*s, mean test *RMSE*s, mean *MAD*s and mean *MAPE*s obtained using CMAC NN, BPNN-SCG, AIA-BPNN and ASA-BPNN do not differ statistically. Table III compares the best results obtained using the SOPNN, CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN, ASA-BPNN and ANFIS. The table shows that these best test *RMSE*s, *MAD*s and *MAPE*s obtained SOPNN, CMAC NN, BPNN-SCG, AIA-BPNN, ASA-BPNN and ANFIS are identical. Totally, SOPNN, CMAC NN, BPNN-SCG, AIA-BPNN, ASA-BPNN and ANFIS can be used for stock price forecasting, and hybrid CI algorithms such as AIA-BPNN and ASA-BPNN can obtain the smallest test *RMSE*s, *MAD*s and *MAPE*s.

V. CONCLUSIONS

This work has compared the prediction accuracies of the SOPNN, CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN ASA-BPNN and ANFIS using the TAIEX dataset. Experimental results indicate that SOPNN, CMAC NN, BPNN-SCG, AIA-BPNN, ASA-BPNN and ANFIS have identical results, and that hybrid algorithms such as AIA-BPNN and ASA-BPNN are recommended for stock price forecasting. Further investigations will develop another hybrid algorithm (real-coded GA-based BPNN, RGA-BPNN) and compare the performance of the SOPNN, CMAC NN, BPNN-SCG, AIA-BPNN, ASA-BPNN, ANFIS with RGA-BPNN using Asian stock market datasets such as Nikkei255 from the Tokyo Stock Exchange, Straits times index (STI) from the Singapore Stock Exchange and Korean Composite stock price indexes (KOSPI) from Korea Stock Exchange.

TABLE I. THE PARAMETER SETTINGS FOR THE SOPNN, CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN, ASA-BPNN AND ANFIS

	Individual algorithms				Hybrid algorithms		
	SOPNN	CMAC NN	BPNN-GD	BPNN-SCG	AIA-BPNN	ASA-BPNN	ANFIS
Parameters settings	σ_k = 0.5	N_i ={30, 40, 50} g = {50, 70, 90, 110} β = 0.005	$neuro\_num$ = {3, 4, 5} β = 0.1	$neuro\_num$ = {3, 4, 5}	$neuro\_num$ = {3, 4, 5} rs ={8, 16, 24} p_{rt} = 0.9	$neuro\_num$ = {3, 4, 5} $T_{g,n}(0)$ =2000 $T_a(0)$ = 1000 $z_{g,n} = z_a$ = 20	MF_num = 2
Termination condition	$k_{\max}$ = 5	$epoch_{\max}$ =200	$epoch_{\max}$ = 10000	$epoch_{\max}$ = 10000 or minimum gradient reached	$g_{\max}$ = 10000	$t_{\max}$ =10000	$epoch_{\max}$ = 30

k_{max} = maximum number of evolutionary layers; $epoch_{max}$ = maximum epoch number; g_{max} = maximum generation number; t_{max} = maximum iteration number

TABLE II. THE BEST MEAN RESULTS OBTAINED USING THE CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN AND ASA-BPNN

	Individual algorithms			Hybrid algorithms		*P* value
	CMAC NN	BPNN-GD	BPNN-SCG	AIA-BPNN	ASA-BPNN	
Mean training *RMSE*	116.21	173.08	108.06	108.40	109.71	0.07
Mean test *RMSE*	130.29	172.66	120.41	120.60	121.51	0.03
Mean test *MAD*	95.08	131.86	89.17	89.20	89.39	0.04
Mean test *MAPE*	1.194	1.617	1.121	1.121	1.124	0.04
MCCT	64.37	41.18	4.06	219.12	193.43	

TABLE III. THE BEST RESULTS OBTAINED USING THE SOPNN, CMAC NN, BPNN-GD, BPNN-SCG, AIA-BPNN, ASA-BPNN AND ANFIS

	Individual algorithms				Hybrid algorithms		
	SOPNN	CMAC NN	BPNN-GD	BPNN-SCG	AIA-BPNN	ASA-BPNN	ANFIS
Training *RMSE*	107.68	115.87	120.67	108.06	108.71	109.14	106.14
Test *RMSE*	124.96	127.84	129.87	120.41	**119.57**	**119.26**	122.35
Test *MAD*	90.61	93.86	97.16	89.17	**88.50**	**87.99**	89.69
Test *MAPE*	1.140	1.179	1.216	1.121	**1.112**	**1.106**	1.129
Computational CPU time	0.55	64.22	40.77	4.06	217.30	179.66	9.20

ACKNOWLEDGMENT

The author would like to thank the National Science Council of the Republic of China, Taiwan for financially supporting this research under Contract Nos. NSC 100-2622-E-262-006-CC3, NSC 98-2221-E-262-014- and NSC 97-2218-E-262-002-.

REFERENCES

[1] J. Z. Wang, J. J. Wang, Z. G. Zhang and S. P. Guo, “Forecasting Stock Indices with Back Propagation Neural Network,” *Expert Systems with Applications*, vol. 38, no. 11, 2011, pp. 14346-14355.

[2] G. S. Atsalakis and K. P. Valavanis, “Surveying Stock Market Forecasting Techniques - Part II: Soft Computing Methods,” *Expert Systems with Applications*, vol. 36, no. 3, Part 2, 2009, pp. 5932-5941.

[3] J. Y. Wu, “Self-Organizing Polynomial Neural Network for Forecasting Chaotic Time Series: Comparison Results,” *Communications in Computer and Information Science,* vol. 201, 2011, pp. 500-510.

[4] W. Pedrycz, “Computational Intelligence as an Emerging Paradigm of Software Engineering,” *Proc. The 14th International Conference on Software Engineering and Knowledge Engineering*, 2002, pp. 7-14.

[5] A. Konar, *Computational Intelligence-Principles, Techniques and Applications*, Springer, 2005.

[6] C. J. Lu and J. Y. Wu, “An Efficient CMAC Neural Network for Stock Index Forecasting,” *Expert Systems With Applications*, vol. 38, no. 12, 2011, pp. 15194-15201.

[7] L. A. Zadeh, “Fuzzy Logic, Neural Networks, and Soft Computing,” *Communications of the ACM* vol. 37, no. 3, 1994, pp. 77-84

[8] T. J. Hsieh, H. F. Hsiao and W. C. Yeh, “Forecasting Stock Markets Using Wavelet Transforms and Recurrent Neural Networks: An Integrated System Based on Artificial Bee Colony Algorithm,” *Applied Soft Computing*, vol. 11, no. 2, 2011, pp. 2510-2525.

[9] J. R. Chang, L. Y. Wei and C. H. Cheng, “A Hybrid Anfis Model Based on Ar and Volatility for Taiex Forecasting,” *Applied Soft Computing*, vol. 11, no. 1, 2011, pp. 1388-1395.

[10] W. Shen, X. Guo, C. Wu and D. Wu, “Forecasting Stock Indices Using Radial Basis Function Neural Networks Optimized by Artificial Fish Swarm Algorithm,” *Knowledge-Based Systems*, vol. 24, no. 3, 2011, pp. 378-385.

[11] J. Y. Wu, “Forecasting Chaotic Time Series Using an Artificial Immune System Algorithm-Based BPNN,” *Proc. The 2010 Conference on Technologies and Applications of Artificial Intelligence*, 2010, pp. 524-531.

[12] J. Y. Wu, “Advanced Simulated Annealing-based BPNN for Forecasting Chaotic Time Series,” *Proc. The 2010 International Conference on Electronics and Information Engineering*, 2010, pp. 38-43.

[13] J. S. Jang, “ANFIS: Adaptive-Network-based Fuzzy Inference Systems,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 23, no. 3, 1993, pp. 665-685.

[14] A. F. Moller, “A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning,” *Neural Networks*, vol. 6, no. 4, 1993, pp. 525-533.

[15] J. Y. Wu, “Solving Constrained Global Optimization Via Artificial Immune System,” *International Journal on Artificial Intelligence Tools*, vol. 20, no. 1, 2011, pp. 1-27.

[16] I. C. Yeh, “Structural Engineering Applications with Augmented Neural Networks,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 13, no. 2, 1998, pp. 83-90.