Knowledge-Based Systems 24 (2011) 378–385

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm

Wei Shen [a], Xiaopen Guo [a], Chao Wu [b], Desheng Wu [c,d,*]

[a] School of Business and Administration, North China Electric Power University, Beijing 102206, China
[b] University of Waterloo, Computing and Financial Management, Ontario, Canada N2L 3G1
[c] RiskLab, University of Toronto, Toronto, ON M5S 3G3 Canada
[d] Reykjavik University Menntavegur 1, 101 Reykjavik, Iceland

ARTICLE INFO

Article history:
Received 9 March 2010
Received in revised form 17 October 2010
Accepted 5 November 2010
Available online 26 November 2010

Keywords:
Artificial fish swarm algorithm
Radial basis function neural network
K-means clustering algorithm
Data mining
Shanghai Stock Exchange Index

ABSTRACT

Stock index forecasting is a hot issue in the financial arena. As the movements of stock indices are non-linear and subject to many internal and external factors, they pose a great challenge to researchers who try to predict them. In this paper, we select a radial basis function neural network (RBFNN) to train data and forecast the stock indices of the Shanghai Stock Exchange. We introduce the artificial fish swarm algorithm (AFSA) to optimize RBF. To increase forecasting efficiency, a *K*-means clustering algorithm is optimized by AFSA in the learning process of RBF. To verify the usefulness of our algorithm, we compared the forecasting results of RBF optimized by AFSA, genetic algorithms (GA) and particle swarm optimization (PSO), as well as forecasting results of ARIMA, BP and support vector machine (SVM). Our experiment indicates that RBF optimized by AFSA is an easy-to-use algorithm with considerable accuracy. Of all the combinations we tried in this paper, BIAS6 + MA5 + ASY4 was the optimum group with the least errors.
© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Stock index forecasting is an important tool for participants in financial markets. Investors rely on it to guard against risks; government organizations use it to monitor market fluctuations. It also serves as a reference for researchers in their studies of financial issues, such as portfolio selection and pricing of financial derivatives. To carry out accurate forecasting, researchers have tried various models and algorithms, and have achieved considerable results. According to theories of model building, stock index forecasting models fall into two categories. In the first category are models based on statistical theories, e.g. General Autoregressive Conditional Heteroskedasticity (GARCH) and Stochastic Volatility model (SV) [1]. In the second category are models based on artificial intelligence, such as the artificial neural network (ANN) [2], the support vector machine (SVM) [3], and the particle swarm optimization (PSO) [4].

Existing research indicates that intelligent forecasting models outperform traditional models, especially in short-term forecasting [5]. However, there is room for improvement on the part of intelligent forecasting models, because stock indices move dramatically in response to many complex factors. To increase forecasting speed and accuracy, researchers have tried to combine and optimize different algorithms, and build hybrid models. For example, Armano et al. [6] optimized ANN with GA to forecast stock indices; Shen and Zhang [7] combined SVM and PSO to carry out stock index forecasting.

In this paper, we chose RBF to forecast the stock index of the Shanghai Security Exchange. An RBF neural network is a three-layered feed-forward network. It has been widely used in short-term prediction for its self-adapting and self-learning features. However, being a typical artificial intelligence network, it has limitations with regard to convergence speed and forecasting accuracy. To tackle this problem, researchers have introduced intelligent algorithms to optimize RBF. Feng and Zhao [8] used RBF optimized by SVM to forecast electricity loads; Zhang and He [9] optimized RBF with GA to forecast nonlinear time series; used AFSA to optimize RBF before using it for facial expression recognition. However, to the best of our knowledge, no optimized RBF algorithms have yet been applied to stock index forecasting. So, we decided to use AFSA to optimize RBF and forecast the Shanghai Security Exchange index.

AFSA, a novel intelligent algorithm, was first proposed in 2002. It was inspired by the natural social behavior of fish in searching, swarming and following. Each individual fish searches for food based in its own way. Information on searching is passed to others, and the swarm achieves a global optimum. The *K*-means clustering algorithm is an effective learning algorithm of RBF, but it is likely to

* Corresponding author at: RiskLab, University of Toronto, 1 Spadina Crescent Room 205, Toronto, ON Canada M5S 3G3.
E-mail addresses: dash@risklab.ca, dwu@rotman.utoronto.ca (D. Wu).

0950-7051/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2010.11.001

converge to a local minimum. AFSA is parallel and independent of initial values, and can also avoid convergence to a local minimum. So in this paper, we used AFSA to adjust width and weight of the center of the *K*-means clustering algorithm for optimizing RBF.

In Section 2 of this paper, we give a brief overview of AFSA and RBF. In Section 3 we optimize the *K*-means clustering algorithm and determine the linking weight of RBF with AFSA. In Section 4, with data obtained from the above procedures, we establish models of the RBF neural network and forecast stock indices with different combination of variables. By using data mining technique, we select important indicators with strong influence on short-term changes in stock indices and form them into different groups from, which we select combinations with the smallest errors. Then we compare the forecasting results of various algorithms. Finally we reach our conclusion.

2. Fundamentals of AFSA and RBF

2.1. Artificial fish swarm algorithm

Inspired by swarm intelligence, AFSA is an artificial intelligent algorithm based on the simulation of the collective behavior of schools of fish. It simulates the behavior of a single artificial fish (AF), and then constructs a swarm of AF. Each AF will search its own local optimum and pass on information in its self-organized system and finally achieve the global optimum.

Suppose the searching space is D-dimensional and there are N fishes in the colony. The current state of a AF is a vector $X = (x_1, x_2, \ldots, x_n)$, where x_i $(i = 1, \ldots, n)$ is the variable to be optimized. The food consistence of AF in the current position is represented by $Y = F(x)$, where Y is the objective function. The distance between the ith and jth individual AF can be expressed as $D_{ij} = ||X_j - X_i||$. In the initial state of the algorithm, the variable of trial number should be defined as trial times of AF searching for food.

We now describe fish swarm behavior in the following five steps.

(1) Searching behavior

Suppose the current state of an AF is X_i, and we randomly select a new state X_j in its visual field. If, in the maximum problem $Y_i < Y_j$ (as the maximum problem and minimum problem can convert with each other, we will discuss maximum problem as example in the following analysis), moves a step in that direction; otherwise, select a state X_j randomly again and judge whether it satisfies the forward condition. If it cannot be satisfied after a pre-set try-number times, it moves a step randomly. The step moving follows the following rule:

$$\begin{cases} X_{i+1} = X_i + Step\frac{X_j - X_i}{\|X_j - X_i\|} & (Y_j > Y_i) \\ X_{i+1} = X_i + Step & (Y_j \leqslant Y_i) \end{cases} \tag{1}$$

(2) Swarming behavior

An AF at current state X_i seeks the companion's number NF and their central position X in its current neighborhood (d_{ij} < Visual); if Y_c/NF > δY_i, it means that at the center of the fish colony, there is enough food and it is not too crowded. Mathematical expression of the swarming behavior:

$$\begin{cases} X_{i+1} = X_i + Step\frac{X_c - X_i}{\|X_c - X_i\|} & (Y_c/\mathrm{NF} > \delta Y_i \text{ and } \mathrm{NF} \geqslant 1) \\ X_{i+1} = Formula\ (1) & (Y_c/\mathrm{NF} \leqslant \delta Y_i \text{ or } \mathrm{NF} = 0) \end{cases} \tag{2}$$

(3) Following behavior

Suppose X_i is the current state of AF searching companion $X_{\max}$ in the neighborhood with $Y_{\max}$, if $Y_{\max}$/NF > δY_i, it means the current position of companion $X_{\max}$ has higher food consistence and it is not too crowded. The AF will move a step towards companion $X_{\max}$; otherwise, continue searching behavior.

Mathematic description of following behavior:

$$\begin{cases} X_{i+1} = X_i + Step\frac{X_{\max} - X_i}{\|X_{\max} - X_i\|} & (Y_{\max}/\mathrm{NF} > \delta Y_i \text{ and } \mathrm{NF} \geqslant 1) \\ X_{i+1} = Formula\ (1) & (Y_{\max}/\mathrm{NF} \leqslant \delta Y_i \text{ or } \mathrm{NF} = 0) \end{cases} \tag{3}$$

(4) Behavior selection

We evaluate the current environment of the FA according to the problem we are to address, and choose a behavior to simulate. Trial method has been frequently taken to simulate fish behaviors, and the best results are implemented after evaluation. In this paper, we observe and analyze three biological behaviors of fish swarm, namely, searching behavior, following behavior and swarming behavior.

(5) Bulletin

Bulletin is used to record the AF's optimal state and the optimal value of the problem. Each AF updates and compares its own state with the bulletin after making movements. If its current state of AF is better, then the value on the bulletin will be replaced.

2.2. Radial basis function neural network (RBF)

Radial basis function neural network is a three-layered feed-forward network. It consists of input layer, hidden layer and output layer. The input layer contains units of signal source, and the second layer is hidden layer. The number of units on the hidden layer is determined by necessity. The third layer is an output layer which reacts to input model. Movement from input layer to hidden layer is nonlinear and that from hidden layer to output layer is linear. Activation function of the units in hidden layer is RBF, which can be demonstrated by the following graph (see Fig. 1).

In Fig. 1, $X = (x_1, x_2, \ldots, x_m)$ is an m-dimensional vector; and $W = (w_1, w_2, \ldots, w_n)$ is the weight of output layer. Activation function is Gaussian and denoted as $g_i(X)$, $i = 1, 2, \ldots,$ n, where n represents the number of neurons in hidden layer. Where $g_i(X)$, $i = 1, 2, \ldots, n$; $g_i(X) = g_i(||X - C_i||)$, C_i is the center of ith activation function, and $||{*}||$ is Euclid norm.

The output of the ith neuron in hidden layer of RBFA can be assumed as:

$$q_i = g_i(\|X - C_i\|) = \exp\left(-\frac{\|X - C_i\|^2}{2\sigma_i^2}\right) \tag{4}$$

where σ_i is the width of the receptive field.

The activation of the output layer is linear combination of units on the hidden layer, which can be expressed as:

$$y = \sum_{i=1}^{n} w_i q_i \tag{5}$$

where w_i is the connecting weights from hidden layer to output layer.

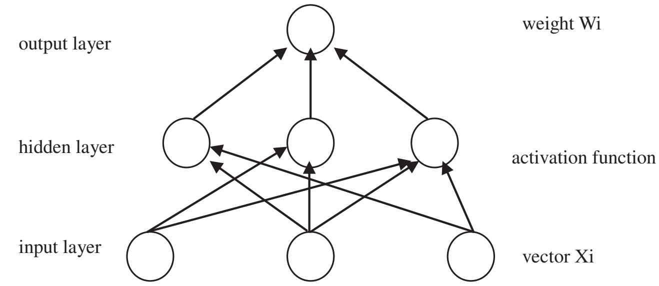

Fig. 1. Structure of RBF.

3. RBF optimized by AFSA in *K*-means clustering algorithm

Before using RBFN, we need to train the model using original data. Learning algorithms are used to determine activation function center C_i of each neuron in the hidden layer, width σ_I, weight W_i from hidden layer to output layer.

Learning algorithm consists of two parts. The first part is clustering of all input samples and working out activation function center C_i of units in hidden layer, and the width σ_i. The second part is to train the linking weights from hidden layer to output layer after C_i is determined.

Being parallel, simple, global and fast, AFSA can be used to upgrade the learning algorithms in determining parameters for neural network.

Suppose the number of training samples is N. First, initiate I (randomly chosen training samples) as center C_0 and width σ_0 ($1 < N$, the number of hidden units). Then we adjust the center following the principle that training samples should be closest in distance to the center, and execute iteration until we find the right primary function C_i. Next, we work out width σ_I with specific RBF. Solution procedures by AFAS include:

(1) Initializing AFAS

Input the number of hidden units I as the population of AF colony; number of training samples N is the maximum iterative times; visual area of AF is 1, moving step-length is η ($0 < \eta < 1$). With stochastic algorithm, we select different samples of number I as initial center $C_i(0)$, where $i = 1, 2, \ldots I$. 0 in the bracket means the iterative number n is 0.

Initial vector of weights $W(0)$, each component vector $w_i(0)$ are random number between $[-1, 1]$. Suppose the number of units in hidden layer is M, and W is the M-dimensional vector, the number 0 in the bracket means the first time of calculation. E_0 is the acceptable network output error. Use output error E to define the food concentration (FC) in the current position of each AF, $\text{FC} = 1/E$, the maximum value of FC will be included in the bulletin and the value assignment of the individual F at the position of the highest FC value is recorded in the bulletin.

(2) Denoting the adjustment of center with the swarming behavior of AF

In the following equation, $i = 1, 2, \ldots, I$; $k = 1, 2, \ldots, N$. In the process of iteration, local optimum of training sample $i(X_k)$ is:

$$i(X_k) = \arg\min \|X_k - C_i(n)\| \tag{6}$$

To adjust a center, we use the following formula:

$$C_i(n+1) = \begin{cases} C_i(n) + \eta[X_k(n) - C_i(n)], & i = i(X_k) \\ C_i(n), & \text{others} \end{cases} \tag{7}$$

That is, in iteration n, we use sample X_k, which is closest to current center $C_i(n)$, to replace $C_i(n)$.

(3) Weight with searching behavior of AF

$$\begin{cases} w_i(t+1) = w_i(t) + \eta \frac{w_i(t+1) - w_i(t)}{\|W_i(t+1) - W_i(t)\|} & (\text{FC}_{t+1} > \text{FC}_t) \\ w_i(t+1) = w_i(t) + \eta & (\text{FC}_{t+1} \leqslant \text{FC}_t) \end{cases} \tag{8}$$

where $w_i(t)$ and $w_i(t+1)$ represents the corresponding values of the ith component weights in the process of iteration from time t to time $t+1$.

(4) Swarming behavior in the adjustment of the center: compare the results of center adjustment with previous iterations to determine error value $0 < \xi < 1$. Thus we suppose the center distribution is approximately fixed and then update the optimum result to the bulletin.

Searching behavior in determining the weight value: compare current FC with that on the bulletin, replace the state on the bulletin if the result is better.

(5) Swarming behavior in the adjustment of the center: now we need to answer the question whether we have tried all the training samples and whether the center distribution is fixed. If yes, stop the process; if no, then $n = n + 1$, enter the next round of iteration.

Searching behavior in determining the weight value: after each iteration, compare current FC with that on the bulletin, replace the state on the bulletin if the result is better. To continue or discontinue the AF action is determined by pre-set condition ($\text{FC} > 1/E_0$) or if or not the maximum iterative N has been achieved. If conditions are satisfied, results are released as an output. Otherwise, iterate again.

(6) Finally we work out the C_i which is the center of primary function of RBF, then variance σ_{I} is determined on the basis of C_i. In terms of Gauss function:

$$\sigma_i = d_{\min}/\sqrt{2I} \tag{9}$$

where $d_{\min}$ is the closest distance between the chosen centers.

Above procedures can be demonstrated by the following structure graph (see Fig. 2).

4. Experiments

4.1. Data

In this paper, we apply RBF algorithm optimized by AFSA to forecast the trend of Shanghai Composite Indices. Forecasting date is from 06.03.2006 to 17.03.2006, with 10 groups of data. Our forecast is a short-term one, and data far from forecasting date provide less and less information useful to forecasting value, therefore we select 30 groups of data from 12.01.2006 to 03.03.2006 as input to construct and train neural networks. Then we carry out forecast and compare the result with actual data.

As stock indices are subject to many factors of influence, we cannot expect to achieve good result by applying a single-factor indicator. Therefore, through using data mining technique, we select 12 indicators with great influence to short-term stock indices (see Table 1) and classify them into three groups. Then we process data as follows.

First, we select and train a group of technical indicators (BIAS6, MA5, OBV, PSY12) and compare the error between the forecast value and the real value. Secondly, we choose and train the average yields of each of the five days prior to a given date, and compare the forecast value to the real value. Thirdly, we choose and train the indices of each of the three days prior to a given date, and compare the forecast value to the real value. Finally, we select from each group the indictors with relatively smaller errors and form optimized group of indicators. We train the optimized groups and compare the errors between the forecast value and the real value.

Input data need to be pre-processed. In this paper, we preprocess input data of BIAS6, OBV and PSY12 with formula (10); input data of MA5 is preprocessed with formula (11). BIS6, OBV and PSY12 are discrete variables correlating to closing index, therefore we use quotient of deviation and standard deviation as input to reduce departure. MA5 is a single factor data processed with mean value method, so we simply use its deviation as input

$$X' = (X - E(X))/(m * \sigma) \tag{10}$$

$$X' = (X - E(X))/m \tag{11}$$

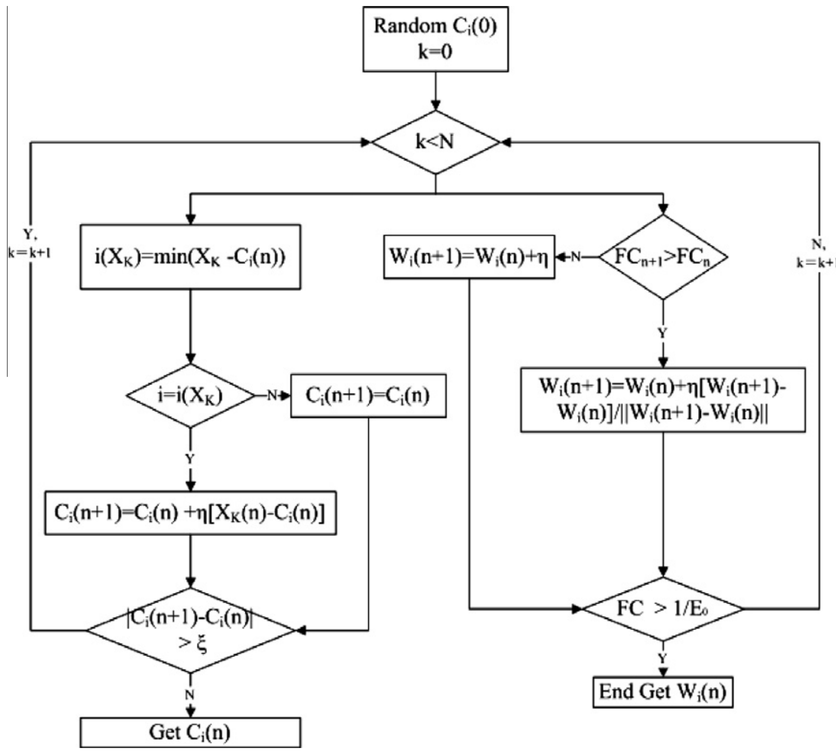

Fig. 2. Procedures of center and weight determination by *K*-means clustering algorithm.

Table 1
Indicators in optimized RBF.

Group	Indicator	Formulas and explanations
1	OBV	OBV (on balance volume) = $v_i + v_{i-1}$, v_i represents trade volume of the current day
1	MA5	MA5 (moving average for 5 days) = $(P_c + P_{c-1} + P_{c-2} + P_{c-3} + P_{c-4})/5$, P_c is the closing index of the current day
1	BIAS6	BIAS6 = $p_c[(p_c - MA6)/MA6] \times 100$
1	PSY12	PSY12 (psychological line for 12 days) = $(D_{UP12}/12) \times 100$, D_{up12} means the number of days when price going up within 12 days
2	ASY5	ASY5 (average stock yield of 5 days before the forecasting date) = $(SY_{c-1} + SY_{c-2} + SY_{c-3} + SY_{c-4} + SY_{c-5})/5$, SY (stock yield) = $(\ln p_c - \ln p_{c-1}) \times 100$
2	ASY4	ASY4 (average stock yield of 4 days before the forecasting date) = $(SY_{c-1} + SY_{c-2} + SY_{c-3} + SY_{c-4})/4$
2	ASY3	ASY3 (average stock yield of 3 days before the forecasting date) = $(SY_{c-1} + SY_{c-2} + SY_{c-3})/3$
2	ASY2	ASY2 (average stock yield of 2 days before the forecasting date) = $(SY_{c-1} + SY_{c-2})/2$
2	ASY1	ASY1 (average stock yield of 1 day before the forecasting date) = SY_{c-1}
3	CI3	Closing indices of 3 days before the forecasting date
3	CI2	Closing indices of 2 days before the forecasting date
3	CI1	Closing indices of 1 day before the forecasting date

where X is the original input vector and $E(X)$ is the expected value of input vector X, σ is the standard deviation of X, m is the product of a constant and the sum of n_1 and n_2, n_1 represents the sequence number of input vector of training sample X_1, n_2 represents the sequence number of input vector of forecast value X_2. We fix the constant at 4.5, which means $m = (n_1 + n_2) \times 4.5$. X' is the transposed input vector.

4.2. Forecasting analysis with single indicator

We carry out forecasts by using each of the above 12 indicators in 3 groups, respectively (see Fig. 3 and Table 2). From forecasting result we reach the following conclusions:

(1) In technology indicator group, OBV forecast has the biggest error ratio and BIAS6 yields the best result.
(2) In average stock yield indictor group, ASY5 and ASY4 have better results than others. This result indicates that weekly average-stock yield has greater influence on closing index.
(3) In closing index group, we find that error increases with the number of days of closing indices. Forecasting error of closing indices of previous 3 days stands high at 2.0292, so we conclude the most reliable indicator in this group is the closing index of 1 day ahead.

Generally speaking, forecasting results with a single indicator are not satisfactory, as there are more factors having influence on stock index movement. However, through our experiment, we have found some indicators with relatively good performance in forecasting accuracy.

4.3. Forecast and analysis with optimized group

In this section, we extract indicators with better performance from the above three groups and put them into optimized groups for forecasting. There are n kinds of indicators and their output are formed into linear combinations: $Y = w_1 * Y_1 + \cdots + w_n * Y_n$, where Y_n is the output vector of the corresponding indicator and w_n represents its weight.

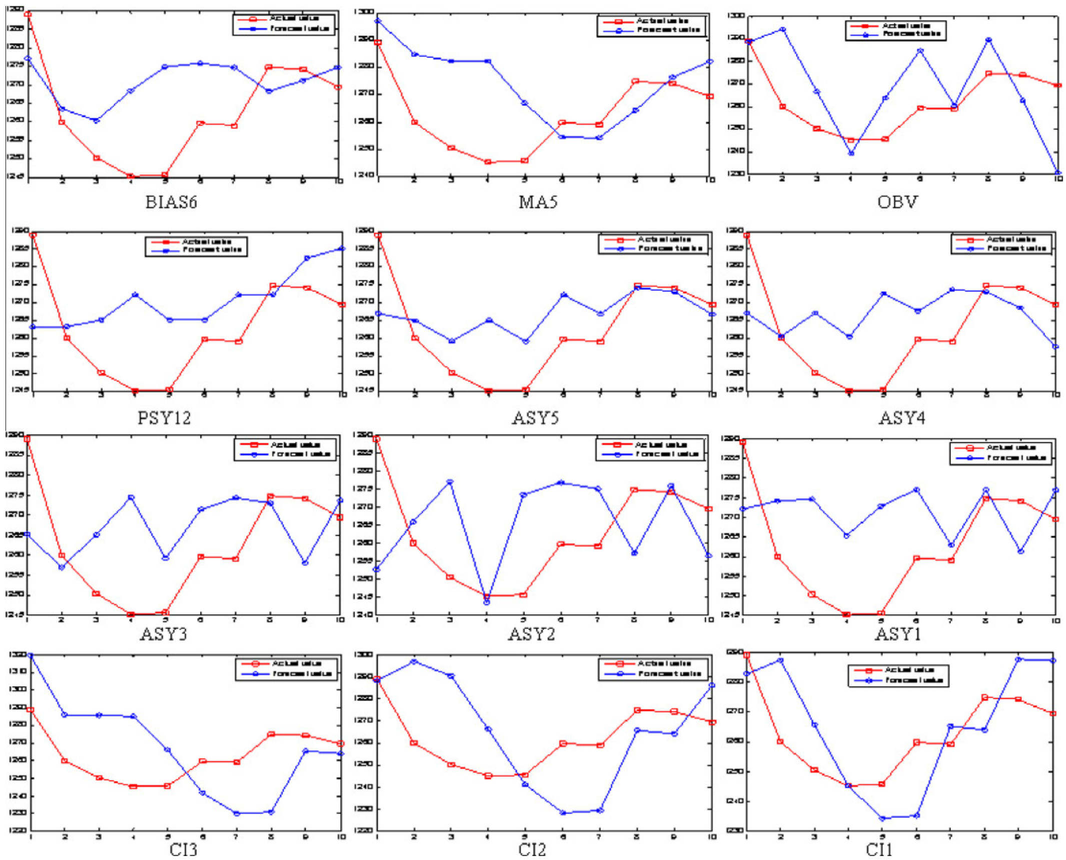

Fig. 3. Comparison of forecasting value with single-factor indicators to the actual value.

Table 2
Daily forecasting error and average error based on single-factor indicators.

Indicator	Error											Average error (%)
	Date	06.03	07.03	08.03	09.03	10.03	13.03	14.03	15.03	16.03	17.03	
	Actual value	1289.0	1259.9	1250.4	1245.2	1245.7	1259.7	1259.0	1274.8	1274.2	1269.5	
BIAS6	Forecast ratio	1277.1	1263.4	1260.3	1268.4	1274.9	1275.9	1274.7	1268.3	1271.2	1274.7	0.9773
	Error ratio (%)	0.9274	0.2775	0.7890	1.8307	2.2904	1.2692	1.2252	0.5144	0.2381	0.4110	
MA5	Forecast ratio	1297.2	1284.9	1282.4	1282.3	1266.9	1254.5	1254.2	1264.3	1276.4	1282.2	1.2445
	Error ratio (%)	0.6333	1.9418	2.4987	2.8949	1.6780	0.4150	0.3893	0.8292	0.1707	0.9941	
OBV	Forecast ratio	1288.7	1294.4	1266.8	1239.3	1263.9	1284.9	1260.5	1289.8	1262.8	1230.9	1.3177
	Error ratio (%)	0.0177	2.6634	1.2990	0.4762	1.4440	1.9626	0.1192	1.1595	0.9003	3.1346	
PSY12	Forecast ratio	1263.3	1263.3	1265.1	1272.1	1265.1	1265.1	1272.1	1272.1	1282.4	1285.2	1.0652
	Error ratio (%)	2.0319	0.2668	1.1628	2.1198	1.5368	0.4285	1.0289	0.2106	0.6395	1.2266	
ASY5	Forecast ratio	1267.0	1264.9	1259.1	1265.1	1259.0	1272.1	1266.7	1274.1	1272.9	1266.7	0.7421
	Error ratio (%)	1.7368	0.3979	0.6916	1.5724	1.0618	0.9789	0.6059	0.0575	0.1004	0.2175	
ASY4	Forecast ratio	1267.0	1260.5	1267.1	1260.4	1272.5	1267.5	1273.6	1273.0	1268.5	1257.6	0.9713
	Error ratio (%)	1.7323	0.0468	1.3191	1.2047	2.1083	0.6181	1.1440	0.1420	0.4514	0.9462	
ASY3	Forecast ratio	1265.2	1256.9	1265.0	1274.5	1259.2	1271.4	1274.3	1272.9	1257.9	1273.6	1.0535
	Error ratio (%)	1.8753	0.2394	1.1575	2.2996	1.0759	0.9233	1.1951	0.1491	1.2918	0.3278	
ASY2	Forecast ratio	1252.7	1265.8	1277.0	1243.5	1273.4	1276.8	1275.1	1257.1	1276.0	1256.5	1.2938
	Error ratio (%)	2.8982	0.4611	2.0869	0.1370	2.1788	1.3429	1.2578	1.4050	0.1385	1.0318	
ASY1	Forecast ratio	1272.1	1274.2	1274.6	1265.3	1272.8	1277.1	1262.8	1276.9	1261.2	1276.9	1.1491
	Error ratio (%)	1.3258	1.1175	1.8988	1.5904	2.1355	1.3609	0.2951	0.1601	1.0273	0.5795	
CI3	Forecast ratio	1319.4	1285.9	1285.7	1285.2	1265.9	1241.9	1230.0	1230.9	1265.4	1263.9	2.0292
	Error ratio (%)	2.3100	2.0212	2.7471	3.1143	1.6004	1.4301	2.3648	3.5652	0.6980	0.4404	
CI2	Forecast ratio	1288.2	1297.0	1290.3	1266.2	1241.2	1228.4	1229.5	1265.6	1264.2	1286.1	1.5792
	Error ratio (%)	0.0561	2.8557	3.0958	1.6614	0.3615	2.5442	2.4070	0.7247	0.7941	1.2918	
CI1	Forecast ratio	1282.7	1287.3	1265.7	1245.2	1234.2	1235.2	1265.3	1264.0	1287.6	1287.1	1.0487
	Error ratio (%)	0.4841	2.1254	1.2133	0.0028	0.9242	1.9802	0.4906	0.8583	1.0396	1.3683	

(1) Combination of BIAS6 and MA5

On the basis of forecasting with MA5 and BIAS6, weights are given to each output values in different combinations. The results we yielded are combinative values based on MA5 and BIAS6.

Forecast result based on combination of BIAS6 and MA5 has relatively stable error ratio, and its curve does not have drastic up-and-downs. The curve fitting degree is between the results achieved by using BIAS6 and MA5 separately.

(2) Combination of ASY3 and CI1

When we forecast with combined indicators of CI1 and ASY3, we find the forecasting average error smaller than the errors from

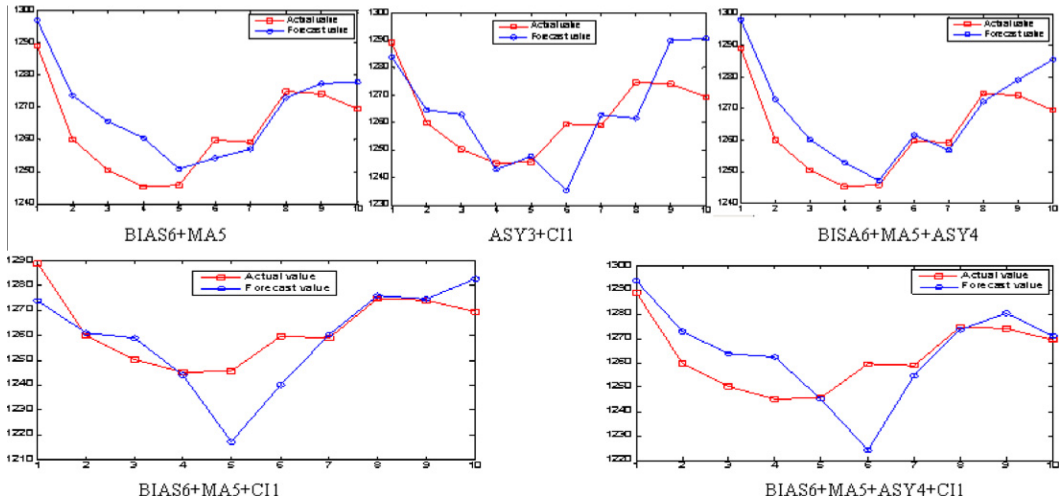

Fig. 4. Comparison of forecasting value with multiple-factor indicators to the actual value.

Table 3
Daily forecasting error and average error based on multiple-factor indicators.

Indicator	Error											Average error (%)
	Date	06.03	07.03	08.03	09.03	10.03	13.03	14.03	15.03	16.03	17.03	
	Actual value	1289.0	1259.9	1250.4	1245.2	1245.7	1259.7	1259.0	1274.8	1274.2	1269.5	
BIAS6 + MA5	Forecast ratio	1296.8	1273.7	1265.5	1260.4	1250.7	1254.1	1256.9	1272.9	1277.3	1277.8	0.6168
	Error ratio (%)	0.6086	1.0804	1.1916	1.2111	0.4066	0.4483	0.1696	0.1529	0.2430	0.6560	
ASY3 + CI1	Forecast ratio	1284.0	1264.7	1263.0	1243.0	1247.7	1235.1	1262.8	1261.7	1289.8	1290.7	0.8287
	Error ratio (%)	0.3877	0.3817	1.0017	0.1728	0.1620	1.9880	0.3006	1.0392	1.2089	1.6442	
BIAS6 + MA5 + ASY4	Forecast ratio	1298.2	1272.9	1260.0	1252.9	1247.0	1261.4	1256.8	1272.1	1279.1	1285.6	0. 5395
	Error ratio (%)	0.7146	1.0198	0.7661	0.6196	0.1094	0.1395	0.1791	0.2108	0.3820	1.2539	
BIAS6 + MA5 + CI1	Forecast ratio	1273.9	1261.1	1259.0	1244.1	1217.0	1240.0	1260.5	1276.1	1274.7	1282.9	0. 7285
	Error ratio (%)	1.1804	0.0915	0.6820	0.0838	2.3509	1.5895	0.1167	0.1045	0.0413	1.0444	
BIAS6 + MA5 + ASY4 + CI1	Forecast ratio	1293.7	1273.0	1263.8	1262.6	1245.1	1224.2	1254.8	1273.7	1280.5	1271.2	0. 7829
	Error ratio (%)	0.3688	1.0273	1.0620	1.3777	0.0433	2.8975	0.3363	0.0900	0.4898	0.1364	

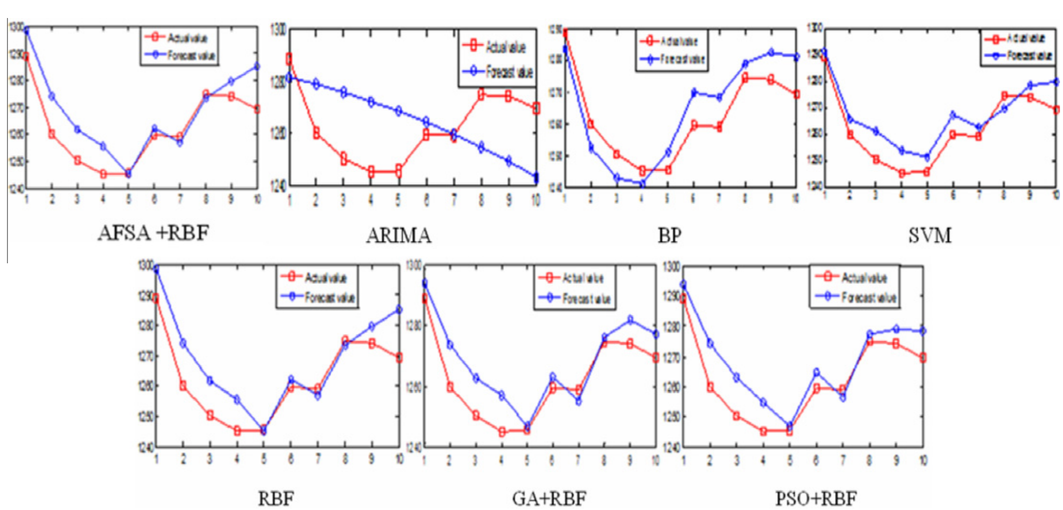

Fig. 5. Comparison of the forecasting results of seven discussed models.

forecasting based on the above two indicators separately. Curve fitting degree is also better.

(3) Combination of BIAS6, MA5 and ASY4

We combine BIAS6, MA5 with ASY5, ASY4, ASY3 and ASY2, and get average error at 0.8839, 0.5395, 0.9819 and 0.8499 respectively. In average stock yield indicator group, BIAS6 + MA5 + ASY4 has the smallest error ratio and the goodness-of-fit is also better than the result from BIAS6 + MA5.

(4) Combination of BIAS6, MA5 and CI1

When CI1 is added to the combination of BIAS6 + MA5, the curve fitting degree is low at the 5th phase and the 6th phase with large deviation, and the average error is larger than that of BIAS6 + MA5.

(5) Combination of BIAS6, MA5 ASY4 and CI1

When we try to add CI1 to BIAS6 + MA5 + ASY4, we find a big deviation appears at the 6th phase on the forecasting curve and that leads to higher average error than that of BIAS6 + MA5 + ASY4.

Through comparison and analysis of the above five groups of forecasting results (see Fig. 4 and Table 3), we can see that the combination of BIAS6 + MA5 + ASY4 has better result, with least error ratio and higher fitting degree to the actual value. This result also indicates that average fluctuation of stock indices and average yield within one week time have more influence on future movement of stock indices.

4.4. Comparison and analysis

To verify the effectiveness of the combination of BIAS6 + MA5 + ASY4 and the forecasting accuracy of AFSA + RBF model, we select 6 forecasting models and compare their forecasting results with that of AFSA + RBF (see Fig. 5 and Table 4).

First, let's see the ARIMA model. Through observation of the historical data series of the closing prices, we found the data series a non-stationary sequence. By ADF inspection, we fixed the exponent number of difference at 1. Then we established ARMA(0, 1), or MA(1) model after analyzing the autocorrelation and partial autocorrelation of the said sequence. As the MA(1) model took only singular quantitative indicators, we input closing price indicators, and got the forecasting result of closing prices of 10 days. From Table 4 we can see that the average error ratio is 1.4134%.

Secondly, we tried BP model. We carried out forecasting with BIAS6, MA5 and ASY4, and the average error ratio was 0.5816%, a reasonably good result. When using BP model, iterations were done day by day, so the forecasting result of the first day had significant influence on the subsequent results.

Thirdly, we introduced BIAS6, MA5 and ASY4 into SVM model and the forecasting error ratio was 0.5091%, which is a very good result.

Fourthly, to further validate our approach in a more comprehensive way, we also carry out experiment with original RBF and GA-based RBF and PSO-based RBF. We introduce BIAS6 + MA5 + ASY4 to RBF algorithm and forecast the stock indices of Shanghai Stock Exchange. The forecasting results are satisfactory with average error ratio at 0.5720. Then we use GA and PSO to optimized RBF and use the optimized algorithms to forecast stock indices. For GA + RBF, the average error ratio is 0.5382, for PSO + RBF, the average error ratio is 0.5177, both yielding very good results as reflected in Fig. 5 and Table 4.

Through above comparative analysis, we found BIAS6 + MA5 + ASY4 had stable performance in the above groups of

Table 4
Daily forecasting error and average error of seven discussed models.

Indicator	Error											Average error (%)
	Date	06.03	07.03	08.03	09.03	10.03	13.03	14.03	15.03	16.03	17.03	
	Actual value	1289.0	1259.9	1250.4	1245.2	1245.7	1259.7	1259.0	1274.8	1274.2	1269.5	
AFSA + RBF (BIAS6 + MA5 + ASY4)	Forecast ratio	1298.2	1272.9	1260.0	1252.9	1247.0	1261.4	1256.8	1272.1	1279.1	1285.6	0. 5395
	Error ratio (%)	0.7146	1.0198	0.7661	0.6196	0.1094	0.1395	0.1791	0.2108	0.3820	1.2539	
ARIMA (closing price)	Forecast ratio	1281.5	1278.8	1275.8	1272.3	1268.5	1264.2	1259.4	1254.4	1249.0	1243.1	1.4134
	Error ratio (%)	0.5800	1.4760	1.9899	2.1353	1.8002	0.3560	0.0278	1.6292	2.0183	2.1212	
BP (BIAS6 + MA5 + ASY4)	Forecast ratio	1283.8	1252.3	1243.1	1241.2	1251.2	1269.9	1268.6	1279.1	1282.5	1281.2	0.5816
	Error ratio (%)	0.4015	0.6082	0.5860	0.3195	0.4433	0.8053	0.7533	0.3352	0.6480	0.9162	
SVM (BIAS6 + MA5 + ASY4)	Forecast ratio	1291.3	1265.5	1261.3	1253.8	1251.5	1267.2	1262.7	1269.5	1278.6	1279.7	0. 5091
	Error ratio (%)	0.1816	0.4412	0.8655	0.6886	0.4672	0.5939	0.2859	0.4184	0.3449	0.8000	
RBF (BIAS6 + MA5 + ASY4)	Forecast ratio	1298.3	1274.0	1261.7	1255.7	1245.3	1262.3	1257.1	1273.3	1279.7	1285.2	0.5720
	Error ratio(%)	0.7198	1.1032	0.8961	0.8357	0.0316	0.2049	0.1546	0.1187	0.4306	1.2246	
GA + RBF (BIAS6 + MA5 + ASY4)	Forecast ratio	1293.8	1273.8	1262.7	1257.2	1246.8	1263.1	1255.3	12763	1281.9	1277.2	0.5382
	Error ratio (%)	0.3745	1.0900	0.9754	0.9572	0.0920	0.2712	0.2983	0.1166	0.6015	0.6059	
PSO + RBF (BIAS6 + MA5 + ASY4)	Forecast ratio	1293.6	1274.1	1263.1	1254.8	1246.9	1264.6	1256.7	1277.4	1278.9	1278.2	0. 5177
	Error ratio (%)	0.3591	1.1133	1.0067	0.7678	0.1000	0.3895	0.1865	0.2026	0.3683	0.6836	

models. This result further highlighted influence of average movement of stock indices and average yield within one week time on short-term index forecasting. The forecasting accuracy of AFSA + RBF was lower than SVM, GA + RBF and POS + RBF, however, as a new intelligent algorithm, it successfully increased the forecasting result of original RBF, and its forecasting error ratio was very close to those of GA + RBF and PSO + RBF, which are mature and well-established. Therefore we have reason to believe AFSA + RBF should be a reliable forecasting model.

5. Conclusions and future work

In this paper, we introduced a new hybrid algorithm, RBF optimized by AFSA, to forecast indices of the Shanghai Stock Exchange. First, we used AFSA to optimize a *K*-means clustering algorithm, and then we used it to determine the linking weight of RBF. Through observation of the relations between stock indices and key technical indicators, average yields, closing indices, we proposed a method for forecasting stock indices. Data mining technique was also introduced to select indicators with better performance and form them into various groups. Then we set them to forecasting by linear transformation. Results of error ratios and curve fitting of various groups were also observed in order to give reference to practical application.

In the technical indicator group, BIAS6 and MA5 outperformed other indicators. Using a single indicator to forecast stock closing index, whether it was a technical indicator, an average stock yield indicator or a closing index indicator, the accuracy was lower than that achieved with group indicators. Of all the hybrid combinations, BIAS6 + MA5 + ASY4 were the optimum group with the smallest forecasting error. When we substituted this combination to other forecasting models, we had similar results, and this indicated that the average movement of stock indices and average stock yield within one week time had more influence on short-term stock index forecasting results.

The average forecasting error of RBF was relatively small. Forecasting accuracy of RBF algorithms optimized by AFSA, GA and PSO improved to some extent and were very close to each other. RBF optimized with AFSA, though not the highest in accuracy, is a useful and easy-to-apply method for parallel computation and being independent from initial value. Moreover, as a new intelligent algorithm, AFSA has room for improvement and development.

This work mainly employs quantitative indicators to forecast the closing indices of Shanghai Stock Exchanges. However, stock movements are affected not only by quantitative factors, but also by non-quantitative factors, such as breaking news, macroeconomic policies and regulations, psychological factors, etc. How to integrate these non-quantitative factors into mathematical algorithms using text-mining techniques to effectively increase forecast accuracy will be left to future research [10].

References

[1] B.D. Garland, SV mixture models with application to S&P 500 index returns, Journal of Financial Economics 85 (3) (2007) 822–856.
[2] C.H. Chen, Neural networks for financial market prediction, IEEE World Congress Computational Intelligence 27 (2) (1994) 1199–1202.
[3] W. Wuang, Y. Nakamori, S. Wang, Forecasting stock market movement direction with support vector machine, Computers and Operations Research 32 (10) (2005) 2513–2522.
[4] R. Majhi, G. Panda, Prediction of S&P500 and DJIA stock indices using particle swarm optimization technique, in: IEEE World Congress on Computational Intelligence, June 1–6, 2008, pp. 1276–1282.
[5] T.O. Hill, M. Connor, W. Remus, Neural network models for time series forecasts, Management Science 42 (7) (1996) 1082–1092.
[6] G. Armano, M. Marchesi, A. Murru, A hybrid genetic-neural architecture for stock indexes forecasting, Information Sciences 170 (2005) 3–33.
[7] W. Shen, Y. Zhang, Stock return forecast with LS-SVM and particle swarm optimization, in: International Conference on Business Intelligence and Financial Engineering, BIFE 2009, 2009, pp. 143–147.
[8] Y. Feng, H. Zhao, Price forecasting algorithm for coal and electricity based on PSO and RBF neural network, in: IEEE International Conference on Control and Automation, 2009, pp. 1365–1369.
[9] Q. Zhang, X. He, RBF network based on genetic algorithm optimization for nonlinear time series prediction, in: Proceedings of International Symposium on Circuits and System, vol. 5, 2003, pp. 693–696.
[10] N. Li, D.D. Wu, Using text mining and sentiment analysis for online forums hotspot detection and forecast, Decision Support Systems 48 (2) (2010) 354–368.