Hindawi
Complexity
Volume 2021, Article ID 6641298, 12 pages
https://doi.org/10.1155/2021/6641298

Research Article

Advantages of Combining Factorization Machine with Elman Neural Network for Volatility Forecasting of Stock Market

Fang Wang,[1,2] **Sai Tang**,[3] **and Menggang Li**[2,4,5]

[1]*School of Economics and Management, Beijing Jiaotong University, Beijing 100044, China*
[2]*Beijing Laboratory of National Economic Security Early-Warning Engineering, Beijing Jiaotong University, Beijing 100044, China*
[3]*School of Humanities, Social Sciences & Law, Harbin Institute of Technology, Harbin, China*
[4]*National Academy of Economic Security, Beijing Jiaotong University, Beijing 100044, China*
[5]*Beijing Center for Industrial Security and Development Research, Beijing Jiaotong University, Beijing 100044, China*

Correspondence should be addressed to Menggang Li; mgli1@bjtu.edu.cn

Received 29 December 2020; Accepted 11 May 2021; Published 24 May 2021

Academic Editor: Thiago Christiano Silva

With a focus in the financial market, stock market dynamics forecasting has received much attention. Predicting stock market fluctuations is usually challenging due to the nonlinear and nonstationary time series of stock prices. The Elman recurrent network is renowned for its capability of dealing with dynamic information, which has made it a successful application to predicting. We developed a hybrid approach which combined Elman recurrent network with factorization machine (FM) technique, i.e., the FM-Elman neural network, to predict stock market volatility. In this paper, the Standard & Poor's 500 Composite Stock Price (S&P 500) index, the Dow Jones industrial average (DJIA) index, the Shanghai Stock Exchange Composite (SSEC) index, and the Shenzhen Securities Component Index (SZI) were used to demonstrate the validity of our proposed FM-Elman model in time-series prediction. The results were compared with predictions obtained from the other two models which are basic BP neural network and the Elman neural network. Some experiments showed that the FM-Elman model outperforms others through different accuracy measures. Furthermore, the effects of volatility degree on prediction performance from different stock indexes were investigated. An interesting phenomenon had been found through some numerical experiments on the effects of different user-specified dimensions on the proposed FM-Elman neural network.

1. Introduction

In recent years, the fluctuation analysis of financial time series has received a lot of concerns. Stock market volatility prediction has become a significant topic in economic research. The study of stock market volatility forecasting can be helpful for policy makers to take appropriate decisions on asset allocation and risk management. Therefore, predicting the volatility of financial time series with a reasonable accuracy deserves much attention. However, stock market exhibits nonlinear and chaotic properties in nature [1, 2]. Statistical models then have some difficulties in dealing with nonlinear and nonstationary time series or deriving satisfactory forecasting performance under the statistical assumptions of normally distributed observations. The predicting becomes more challenging.

Artificial neural network has the advantages on learning from sample data and capturing the nonlinear relations among interconnected neurons through training mode [3]. It is capable of dealing with nonlinear high-dimensional data and approximating any nonlinear functions with arbitrary precision [4–7]. Particularly, the simple recurrent network, i.e., Elman neural network (Elman NN) [8] has shown its stronger ability as it has the characteristic of time-varying. And the Elman NN is a kind of feedback network where the added layer connecting to the hidden layer can be regarded as a time delay operator capable of memorizing recent events. It is a time-varying predictive control system that has faster convergence and more accurate mapping ability.

Elman NN has been utilized to financial prediction and applied to many other different types of time series. Most studies on Elman NN obtained higher accuracy. Zheng [9] used an Elman NN to forecast opening prices of the Shanghai Stock Exchange. Wu and Duan applied the Elman NN in predicting stock [10] and gold future markets [11], respectively. In the area of electricity prediction, Rani and Victoire [12] integrated the decomposition method and group search optimization algorithm into the Elman NN. It showed that the Elman NN outperformed other approaches.

There are also other artificial neural networks like wavelet neural network and radial basis function neural network [13–16]. Some developed artificial intelligence techniques like expert systems [17, 18], support vector machines (SVMs) [19, 20], and hybrid methods [21, 22] are also applied in forecasting stock prices. Recently, some novel models have utilized random jump or random time effective function with different neural networks [23, 24] which have been proposed in forecasting financial market.

Although the models which are based on artificial intelligent have achieved remarkable results, there are still limitations. There is a few technique in most models which pay attention to the nonlinear interactions among the inputs. For example, the nonlinearities in neural network models were handled by the activation functions. These models without consideration of interactions between features with different scales have been widely used in some applications such as image processing, mechanical translation, and speech recognition [25–27].

FM is originally used for collaborative recommendations which were first introduced by Rendle [28]. FM is a supervised learning method that can model feature interactions with second-order even when the data have very high sparsity and high dimension. FMs show state-of-the-art performance as they have two main benefits. First, FMs are on a par with polynomial regression but can achieve empirical accuracy with smaller and faster evaluation results. Second, unlike the linear regression, FMs can infer the weights of feature interactions that were not observed in the training dataset. The weights of second-order feature interactions have the low-rank property which makes FMs become increasingly popular in the recommender system. Although FM is a general framework of matrix factorization, FM shows more flexibility as the matrix factorization method only models the relation between two entities [29]. FMs are general predictors like SVMs and have a lot of applications in industry. FMs are applicable to any variables with real feature and are not restricted to recommender systems. FM gives a promising direction for the prediction purpose in regression, classification, and ranking [30–33].

As far as we know, real-world time series is rarely pure nontime-varying. And the linear regression is not always capable of deriving the interactions between features which however are more common in various applications. Hence, the problem of dealing with time-varying and nonlinear interactions can be solved by combining FM with Elman NN. Moreover, it is almost universally agreed in the forecasting literature that no single model is the best in every situation because a real-world problem is often complex. Using any single model may not be able to capture different patterns equally well [34]. Therefore, we propose a forecasting model combining FM technique with Elman NN for stock market volatility prediction in the present paper.

In this paper, we apply the FM-Elman neural network to forecast the volatility degree's behavior of the Standard & Poor's 500 Composite Stock Price (S&P 500) index, the Dow Jones industrial average (DJIA) index, the Shanghai Stock Exchange Composite (SSEC) index, and the Shenzhen Securities Component index (SZI) from January 2nd, 2000, to December 31st, 2011. Different threshold values were introduced into our model, and the corresponding volatility prediction results were presented. To show the advantages of the proposed FM-Elman model, we compare the predicting results with two other neural network models including BP network and Elman recurrent network through three performance evaluation measures such as the mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE).

The remainder of this paper is presented as following sections. In Section 2, the Elman NN and FM are reviewed where they are prepared for our proposed model. Then, we give the prediction model FM-Elman neural network in Section 3. In this section, we first give the model description and in the same time introduce some needed ingredients of it. And the algorithm of the FM-Elman model is also given. Section 4 presents the main forecasting results of the FM-Elman model. This section gives predicting comparisons among our proposed model, BP neural network, and Elman neural network. It not only presents the effects of different parameters like volatility degree and user-specified dimension on the FM-Elman model's performance but also considers other evaluation measures. And Section 5 highlights some necessary conclusions finally.

2. Elman Neural Network and Factorization Machine

2.1. Elman Neural Network (Elman NN). Elman neural network was founded by Elman [8] in 1990 which is famous for its recurrent topology structure. Unlike the BP network, an Elman NN has a set of recurrent nodes. The so-called recurrent nodes in the buffer received message from the peered output nodes in the hidden layer and then transmitted message to the hidden layer. Every hidden node is connected to only one recurrent neuron, and the message will remain the same after transmitting. Hence, the number of recurrent layer nodes is the same as the number of hidden nodes, and the recurrent layer contains the state of input data from the hidden layer.

Figure 1 gives the structure of multi-input Elman NN. The Elman NN is composed of the input layer, the hidden layer, the output layer, and the recurrent layer. There are n nodes in the input layer, and both the hidden layer and the recurrent layer have m nodes. In the output layer, there exists only one unit neuron. The mathematical computation for the nonlinear state of the Elman NN is

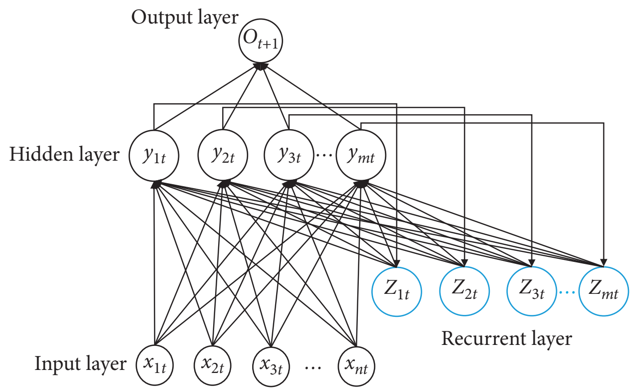


Figure 1: Topology of the Elman neural network.

$$\begin{cases} \vec{u}(t) = f\left(W^A \vec{x}(t-1) + W^B \vec{z}(t)\right), \\ \vec{z}(t) = \vec{u}(t-1), \\ y(t) = g\left(W^C \vec{u}(t)\right), \end{cases} \tag{1}$$

where $\vec{u}(t)$ is the vector of output values in the hidden layer, $y(t)$ is the final output of the network, and $\vec{x}(t-1) = (x_{1,t-1}, x_{2,t-1}, \ldots, x_{n,t-1})$ denotes the input of the network at time $t-1$. The weight matrix W^A connects the input layer node to the node in the hidden layer, W^B connects the node in the recurrent layer to the hidden layer neuron, and W^C is the matrix which connects the node in the hidden layer to the output node. Functions $f(\cdot)$ and $g(\cdot)$ are the activation functions where $f(x)$ is the sigmoid function and $g(x)$ is an identity function in this paper.

From equation (1) and through deduction, we can obtain that

$$\vec{z}(t) = f\left(W^A_{t-1} \vec{x}(t-2) + W^B_{t-1} \vec{z}(t-1)\right), \tag{2}$$

where $\vec{z}(t)$ depends on the matrix W^A_{t-1} and W^B_{t-1} which comes from different time. Elman NN has the ability to adapt to time series varying.

2.2. Factorization Machine. FM has the same prediction ability as SVMs but also has capability of estimating reliable parameters under very sparse data. The feature of modelling all variable interactions is comparable to a polynomial kernel in SVM. The equation for a FM with second-order feature is defined as follows:

$$\hat{y}(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j, \tag{3}$$

where the parameters $w_0 \in \mathbb{R}$, $\mathbf{w} \in \mathbb{R}^n$, and $\mathbf{V} \in \mathbb{R}^{n\times k}$ have to be determined. And $\langle \cdot, \cdot \rangle$ is the inner product of two vectors with size k. Then,

$$\langle v_i, v_j \rangle = \sum_{f=1}^{k} v_{i,f} \cdot v_{j,f}, \tag{4}$$

which models the interaction between the ith variable and the jth variable, where v_i is the ith variable with $k (\in \mathbb{N}_0^+)$ dimension factors.

Our intuition for the complexity of equation (3) is in $O(kn^2)$ because all pairwise interactions have to be computed. As there is no parameter in a model depending on two variables directly, the pairwise interactions in equation (3) are reformulated as follows:

$$\sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j = \frac{1}{2} \sum_{f=1}^{k} \left(\left(\sum_{i=1}^{n} v_{if} x_i \right)^2 - \sum_{i=1}^{n} v_{if}^2 x_i^2 \right). \tag{5}$$

And the equation only needs linear runtime $O(kn)$ to be computed after the reformulation. So, FMs are applicable from a computational point of view.

3. Our Proposed Method

3.1. Modelling. We construct the Elman recurrent neural network with factorization machine, i.e., FM-Elman neural network, to predict the volatility of different stock indexes. The detailed topology of FM-Elman neural network is presented in Figure 2.

The layers of the FM-Elman neural network are analyzed as follows:

(1) *Hidden Layer*. The nodes in the hidden layer are partitioned into two parts. One part of them has normal nodes which show the linear relations among the input data. And the remaining nodes in the other part incorporate all interactions between each pair of features from the input data. The results are computed by

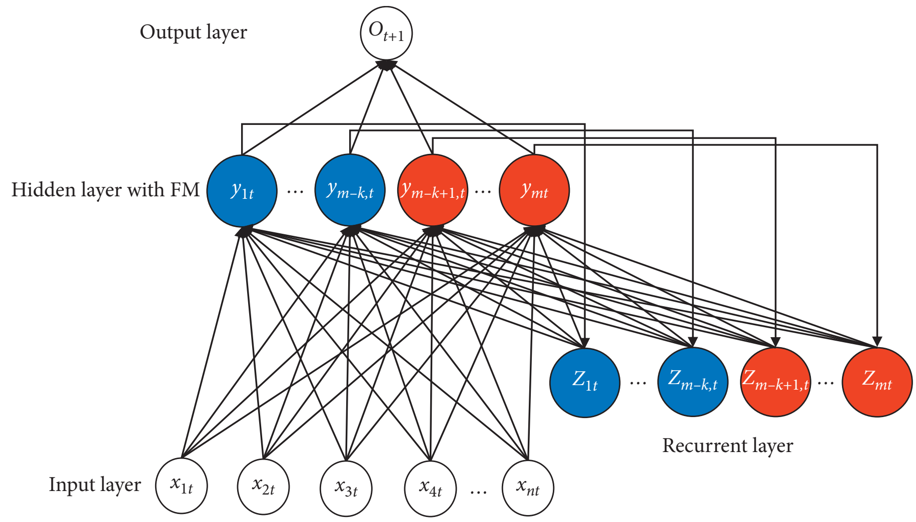


FIGURE 2: Structure of the Elman neural network with factorization machine.

$$y_j(t) = f\left(\sum_{i=1}^{n} v_{ij}x_i(t) + \sum_{z=1}^{m} u_{zj}x_z(t)\right), \quad i = 1,2,\ldots,n,\ j = 1,2,\ldots,m-k, z = 1,2,\ldots,m,$$

$$y_{m-k+j}(t) = f\left(\begin{array}{c}\frac{1}{2}\left(\left(\sum_{i=1}^{n}\hat{v}_{i,m-k+j}x_i(t)\right)^2 - \sum_{i=1}^{n}\left(\hat{v}_{i,m-k+j}\right)^2 x_i^2(t)\right) + \\ \frac{1}{2}\left(\left(\sum_{i=1}^{n}\hat{u}_{z,m-k+j}x_z(t)\right)^2 - \sum_{i=1}^{n}\left(\hat{u}_{z,m-k+j}\right)^2 x_z^2(t)\right)\end{array}\right), \quad i = 1,2,\ldots,n,\ j = 1,2,\ldots,k, z = 1,2,\ldots,m, \tag{6}$$

where x_i is the input value from input node i, y_j denotes the value of the jth node in the hidden layer, v_{ij} is the undetermined weight which relates the ith input node to the jth normal node in the hidden layer, $\hat{v}_{ij}$ is the weight connecting the ith input node to the remaining node in the hidden layer which is also undetermined, u_{zj} and $\hat{u}_{zj}$ have the same meaning with v_{ij} and $\hat{v}_{ij}$ except for the first two parameters which link the nodes in the recurrent layer to the hidden layer, t is the iteration number in the formulas, k is the user-specified dimension, and f is the activation function.

(2) *Recurrent Layer*. The number of nodes in the recurrent layer is the same as the number of hidden nodes. Each hidden node is connected to only one node in the recurrent layer, and the connected weight is a constant value one. So, the recurrent layer is also partitioned into two parts which are presented as follows:

$$\begin{aligned} x_z(t) &= y_j(t-1), \quad j = 1,2,\ldots,m-k, \\ &\quad z = 1,2,\ldots,m-k, \\ x_{m-k+z}(t) &= y_{m-k+j}(t-1), \quad j = 1,2,\ldots,k, \\ &\quad z = 1,2,\ldots,k. \end{aligned} \tag{7}$$

(3) *Output Layer*. The outputs are

$$O_j(t) = g\left(\sum_{i=1}^{m} w_{ij}y_i(t)\right), \quad j = 1,2,\ldots,p, \tag{8}$$

where w_{ij} is the undermined weight, O_j is the output value of jth node in the output layer, and g is the activation function.

There are different loss functions to calculate the error between the actual and the estimated values from the output layer. We consider the squared loss function which is given as follows:

$$E(t) = \frac{1}{2}(T(t) - O(t))^2, \tag{9}$$

where $T(t)$ is the actual value in the tth iteration.

So, the final output error of the FM-Elman model is computed by

$$l = \frac{1}{N}\sum_{t=1}^{N}E(t) = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}(T(t) - O(t))^2. \tag{10}$$

To optimize the FM-Elman model, we often use the stochastic gradient decent method to update the weights until it achieves convergence.

3.2. Algorithm of FM-Elman Model. The training process of the FM-Elman model is detailed as follows:

(1) The gradients of the weights and the updated rule in the output layer are computed as follows:

$$\Delta w_j = -\eta \frac{\partial l}{\partial w_j} = \eta (T(t) - O(t)) y_j(t),$$
$$w_j \leftarrow w_j - \eta \frac{\partial l}{\partial w_j}, \tag{11}$$

where $j = 1, 2, \ldots, m$, η is the learning rate.

(2) The gradients of the weights and the updated rule in the hidden layer connected by the recurrent layer are calculated as the following two cases:

$$\Delta u_{zj} = -\eta \frac{\partial l}{\partial u_{zj}} = \eta (T(t) - O(t)) g' w_j f' x_z(t), \quad j = 1, 2, \ldots, m-k,$$
$$\Delta \widehat{u}_{z,m-k+j} = -\eta \frac{\partial l}{\partial \widehat{u}_{z,m-k+j}} = \eta (T(t) - O(t)) g' w_{m-k+j} f' \left(\left(\sum_{z=1}^{m} \widehat{u}_{z,m-k+j} x_z(t) \right) x_z(t) - \sum_{z=1}^{m} \widehat{u}_{z,m-k+j} x_z^2(t) \right), \quad j = 1, 2, \ldots, k,$$
$$u_{zj} \leftarrow u_{zj} - \eta \frac{\partial l}{\partial u_{zj}}, \quad j = 1, 2, \ldots, m-k,$$
$$\widehat{u}_{z,m-k+j} \leftarrow \widehat{u}_{z,m-k+j} - \eta \frac{\partial l}{\partial \widehat{u}_{z,m-k+j}}, \quad j = 1, 2, \ldots, k, \tag{12}$$

where $z = 1, 2, \ldots, m$, η is the learning rate, k is the user-specified dimension, and g' and f' are corresponded derivative functions of g and f.

(3) The gradients of the weights and the updated rule in the hidden layer connected by the input layer are computed as the following two cases:

$$\Delta v_{ij} = -\eta \frac{\partial l}{\partial v_{ij}} = \eta (T(t) - O(t)) g' w_j f' x_i(t), \quad j = 1, 2, \ldots, m-k,$$
$$\Delta \widehat{v}_{i,m-k+j} = -\eta \frac{\partial l}{\partial \widehat{v}_{i,m-k+j}} = \eta (T(t) - O(t)) g' w_{m-k+j} f' \left(\left(\sum_{i=1}^{m} \widehat{v}_{i,m-k+j} x_i(t) \right) x_z(t) - \sum_{i=1}^{m} \widehat{v}_{i,m-k+j} x_i^2(t) \right), \quad j = 1, 2, \ldots, k,$$
$$v_{ij} \leftarrow v_{ij} - \eta \frac{\partial l}{\partial v_{ij}}, \quad j = 1, 2, \ldots, m-k,$$
$$\widehat{v}_{i,m-k+j} \leftarrow \widehat{v}_{i,m-k+j} - \eta \frac{\partial l}{\partial \widehat{v}_{i,m-k+j}}, \quad j = 1, 2, \ldots, k, \tag{13}$$

where $i = 1, 2, \ldots, n$, η is the learning rate, k is the user-specified dimension, and g' and f' are corresponded derivative functions of g and f.

4. Forecasting Results

4.1. Data Selecting and Processing. Stock prices' different changing behaviors and volatility predicting study have long been a focus in economic research. We use the logarithmic return to describe the statistical characteristic of a stock return volatility. The stock logarithmic return is defined as

$$r(t) = \ln S(t) - \ln S(t-1), \tag{14}$$

where $S(t)$ denotes the stock daily closing price at time t.

The data (http://www.finance.yahoo.com) chosen for our experiment are the Standard & Poor's 500 Composite Stock Price (S&P 500) index, the Dow Jones industrial average (DJIA) index, the Shanghai Stock Exchange Composite (SSEC) index, and the Shenzhen Securities Component index (SZI). All the data are collected from trading days ranging from January 2nd, 2000, to December 31st, 2011. So, the size of different time series is 3019, 3027, 2901, and 2902, respectively. We partition them into training sets (from January 2nd, 2000, to December 31st, 2007) and testing sets (from January 2nd, 2008, to December 31st, 2011). Table 1 describes the data's statistical feature, where N_{all} and N_{test} denote the sizes of the whole data and the testing samples, respectively.

In this paper, a threshold value $\theta(\geq 0)$ is introduced as the volatility degree. Let $\mathfrak{R}(\theta)$ denote the set in which the stock returns' absolute values are greater than the valueθ, and the definition is given by $\mathfrak{R}(\theta) = \{r(t) \| r(t)| \geq \theta,\ t = 1, 2, \ldots, T\}$. Once the value θ is set, we can obtain the dataset including the satisfied stock daily closing price. Figure 3 gives an example of stock returns with different thresholds. For a fixed threshold value θ, the corresponding stock trading dates are determined. The trading dates are in the set where time t values satisfy $|r(t)| \geq \theta$, $t = 1, 2, \ldots, T$. Newly formed series are arranged in a chronological order. Table 2 gives the numbers of data for stock indexes distributed in training data and testing data under different threshold values θ.

When the threshold value θ equals 0, the averaged values of daily absolute returns for S&P 500, DJIA, SSEC, and SZI are 0.0094, 0.0089, 0.0117, and 0.0130, respectively. We set different volatility degrees 0.003, 0.006, 0.009, and 0.012 to see the data numbers distributed in the training and testing data set, respectively. In Table 2, as the threshold value θ increases, the quantity of data in both training dataset and testing dataset that exceeds the given threshold value gradually reduces. And it can be predicted that when θ is larger than 0.012, the corresponding numbers will be fewer.

Four input variables including the daily opening prices, the daily highest prices, the daily lowest prices, and the daily closing prices are selected according to the newly formed dates. And we choose the next time daily closing price in the chronological ordered datasets as the output variable. In order to reduce the noise's impact on the stock markets, all the input data X are normalized as follows:

Table 1: The data and their descriptive statistical analysis.

Name	N_{all}	N_{test}	Mean	Std.
S&P 500	3019	1009	1190.02	184.26
DJIA	3027	1009	10612.79	1397.02
SSEC	2901	976	2226.36	973.46
SZI	2902	976	7028.88	4389.84

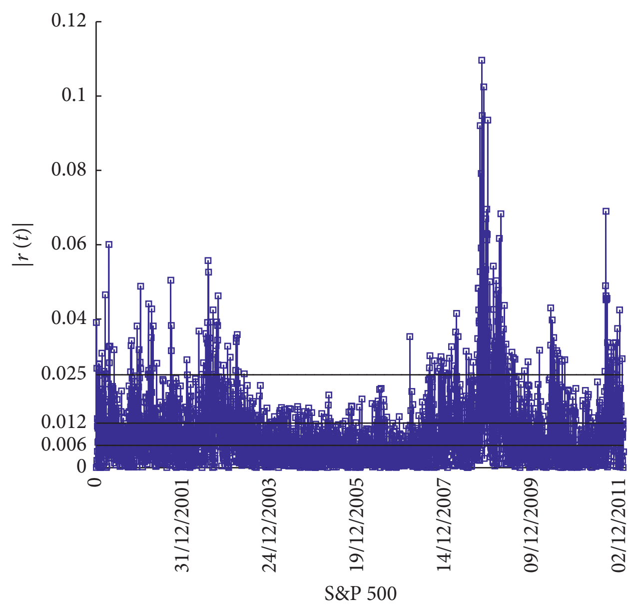


Figure 3: Absolute returns of S&P 500 with different thresholds.

$$X' = \frac{X - \min X}{\max X - \min X}. \tag{15}$$

Then, it is easily to obtain the actual prediction value through $X = X'(\max X - \min X) + \min X$.

4.2. Performances of FM-Elman Model. In the proposed FM-Elman neural network, we choose the structure with $4 \times 10 \times 1$ where the number of input nodes is 4, the number of the hidden nodes is 10, and the number of the output nodes is 1. We set the maximum iterations number as 5000, $\eta = 0.02$, and the predefined minimum training threshold is $\varepsilon = 5 \times 10^{-5}$.

To analyze and evaluate the predicting performance of the FM-Elman neural network model, we use the accuracy measures with the corresponding definitions as follows:

$$\text{MAE} = \frac{1}{N}\sum_{t=1}^{N} |T(t) - O(t)|,$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N} (T(t) - O(t))^2}, \tag{16}$$

$$\text{MAPE} = 100 \times \frac{1}{N}\sum_{t=1}^{N} \left|\frac{T(t) - O(t)}{T(t)}\right|,$$

Table 2: Numbers of data distributed in training data and testing data under different threshold values θ.

Threshold θ	S&P 500		DJIA		SSEC		SZI	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing
0	2010	1009	2018	1009	1925	976	1926	976
0.003	1415	777	1426	756	1485	785	1520	824
0.006	987	585	945	544	1105	643	1190	688
0.009	663	455	622	431	817	520	923	562
0.012	447	350	418	327	615	411	688	462

where $T(t)$ and $O(t)$ are the actual value and the predictive value in the tth iteration and N is the sample size. When the values of these evaluation measures are smaller, the prediction performance is better.

In this section, we derive the stock prices' different fluctuation behaviors through the proposed FM-Elman neural network. First, the new proposed model is proved to be better compared with BPNN and Elman neural model through different evaluation measures. Then, the prediction performance of FM-Elman neural network is measured when threshold value θ varies. And finally, we can see how the user-specified dimension k impacts the performance of the proposed model.

When the threshold value equals 0, the training datasets and testing datasets are the original datasets. And the prediction results of S&P500 and SSEC by the FM-Elman neural model with $k = 3$ are presented in Figures 4 and 5.

We then give the performance comparisons among different prediction models for $\theta = 0$ in Table 3 where the MAPE (100) means the latest 100 days of MAPE in the testing data. Three different prediction models include BPNN, Elman neural network, and FM-Elman neural network with the user-specified dimension $k = 3$. Table 3 shows that FM-Elman model' evaluation errors are all smaller than those in the other two models. In addition, the MAPE (100) value is smaller than the corresponding stock index' MAPE value. It shows that the short-term prediction outperforms the long-term prediction.

4.3. The Impact of θ. When θ varies from 0.003 to 0.012, different prediction analysis of indexes S&P 500, DJIA, SSEC, and SZI can be performed by the FM-Elman neural network. Figures 6 and 7 are the prediction analysis of S&P 500 and SSEC by the FM-Elman neural model. The two figures also show the effectiveness of forecasting with different volatility degree values of θ. When θ is small, the performance of volatility prediction is revealed better through the empirical results. Like $\theta = 0.03$ and $\theta = 0.06$ in Figures 6(a) and 6(b), the predictive values are closer to the actual values than those in Figures 6(c) and 6(d). Figure 7 also indicates the similar results.

We choose the often recommended criterion MAPE to measure the prediction performance for stock indexes S&P 500, DJIA, SSEC, and SZI under the FM-Elman neural model which is presented in Table 4. When the value of θ increases, the value of MAPE increases gradually. And the numerical experiment results show that using the FM-Elman neural network model, the volatility degree forecasting is feasible.

4.4. The Impact of k. In this subsection, we want to analyze how the user-specified dimension k affects the prediction performance by the FM-Elman neural model through numerical experiment when $\theta = 0$. From the descriptions in Section 3, when k increases, the amount of red nodes in both hidden layer and recurrent layer becomes larger. That means more hidden and recurrent nodes in the proposed FM-Elman neural network will contain interaction information from the connected inputs. Then, the computation becomes complicated as k increases. It is interesting to see in Table 5 that the evaluation values of MAE, MAPE, and RMSE for indexes S&P 500, DJIA, SSEC, and SZI increase first and then decrease with the increasing of k. So, the low user-specified dimension or high user-specified dimension is a better choice. No matter which outperforms the predicting results of BPNN and Elman NN from the previous Table 3.

4.5. Further Predicting Performance Evaluation. We adopt three trend-type statistical methods, i.e., directional symmetry (DS), correct up-trend (CP), and correct down-trend (CD) [35], to check the practical stock movement. When the values of these three performance evaluation results become larger, the forecasting of change direction will be more precise. The definitions of these three performance evaluation methods are given as

$$\mathrm{DS} = \frac{100}{N_1} \sum_{t=1}^{N_1} a_t, \quad a_t = \begin{cases} 1, & \text{if } (T_t - T_{t-1})(O_t - O_{t-1}) \geq 0, \\ 0, & \text{otherwise,} \end{cases} \tag{17}$$

where N_1 is the number of testing samples.

$$\mathrm{CP} = \frac{100}{N_2} \sum_{t=1}^{N_2} a_t, \quad a_t = \begin{cases} 1, & \text{if } T_t - T_{t-1} > 0 \text{ and } O_t - O_{t-1} \geq 0, \\ 0, & \text{otherwise,} \end{cases} \tag{18}$$

where N_2 is the number of testing samples which satisfy $T_t - T_{t-1} > 0$.

$$\mathrm{CD} = \frac{100}{N_3} \sum_{t=1}^{N_3} a_t, \quad a_t = \begin{cases} 1, & \text{if } T_t - T_{t-1} < 0 \text{ and } O_t - O_{t-1} \leq 0, \\ 0, & \text{otherwise,} \end{cases} \tag{19}$$

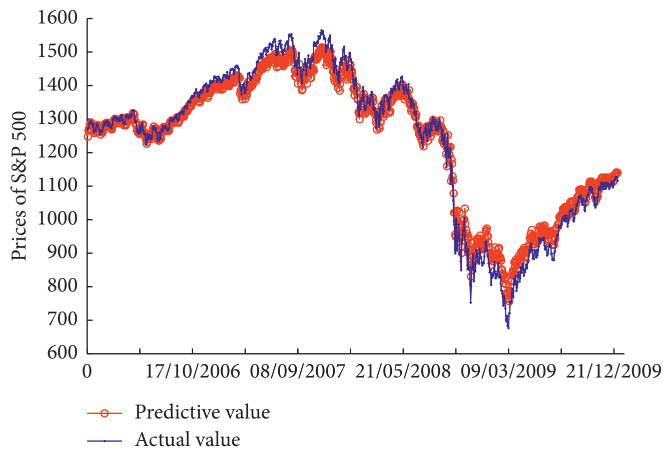


Figure 4: Comparisons of prediction results and the daily closing prices of S&P 500 for testing datasets.

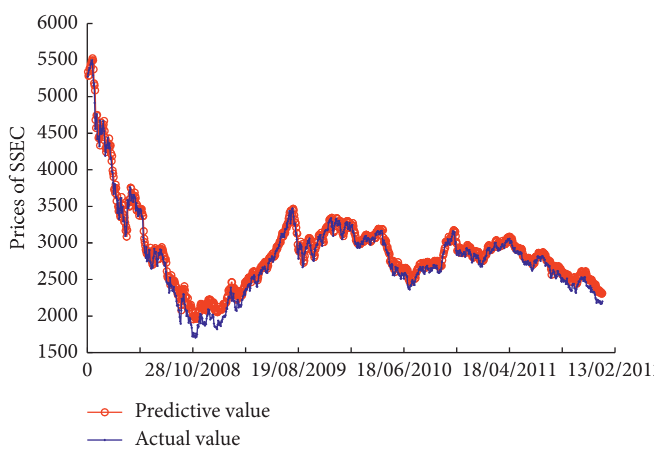


Figure 5: Comparisons of prediction results and the daily closing prices of SSEC for testing datasets.

Table 3: Comparisons of prediction performances among different models for $\theta = 0$.

Evaluation errors	S&P 500			DJIA		
	BPNN	Elman NN	FM-Elman	BPNN	Elman NN	FM-Elman
MAE	54.1822	46.8432	43.5088	321.3915	294.4472	280.9011
MAPE	5.1566	4.5231	4.1692	3.3196	3.0473	2.9379
RMSE	60.2121	53.8414	49.7034	382.7756	354.6576	348.8277
MAPE (100)	1.7813	1.5526	1.5273	1.1938	1.1733	1.1103
Evaluation errors	SSEC			SZI		
	BPNN	Elman NN	FM-Elman	BPNN	Elman NN	FM-Elman
MAE	68.5672	53.6271	50.4433	287.2003	255.4868	228.2216
MAPE	2.4598	1.9369	1.8008	2.6363	2.2933	2.0552
RMSE	85.4527	72.682	69.7871	355.0629	324.7933	298.6005
MAPE (100)	2.8151	2.5933	2.5618	3.0594	2.8707	2.7645

where N_3 is the number of testing samples which satisfy $T_t - T_{t-1} < 0$. T_t and O_t are the actual value and the predictive value in the tth iteration, respectively.

In Table 6, the trend-type measures DS, CP, and CD for stock indexes S&P 500, DJIA, SSEC, and SZI under varying volatility degrees are presented through some numerical experiments. When the value of θ changes, all the stock indexes change a little. And we can see that the direction forecasting results of SSEC and SZI show better performance than S&P 500 and DJIA since the DS, CP, and CD values in two indexes before all exceed 50. And the performance results of stock indexes S&P 500 and DJIA are more sensitive

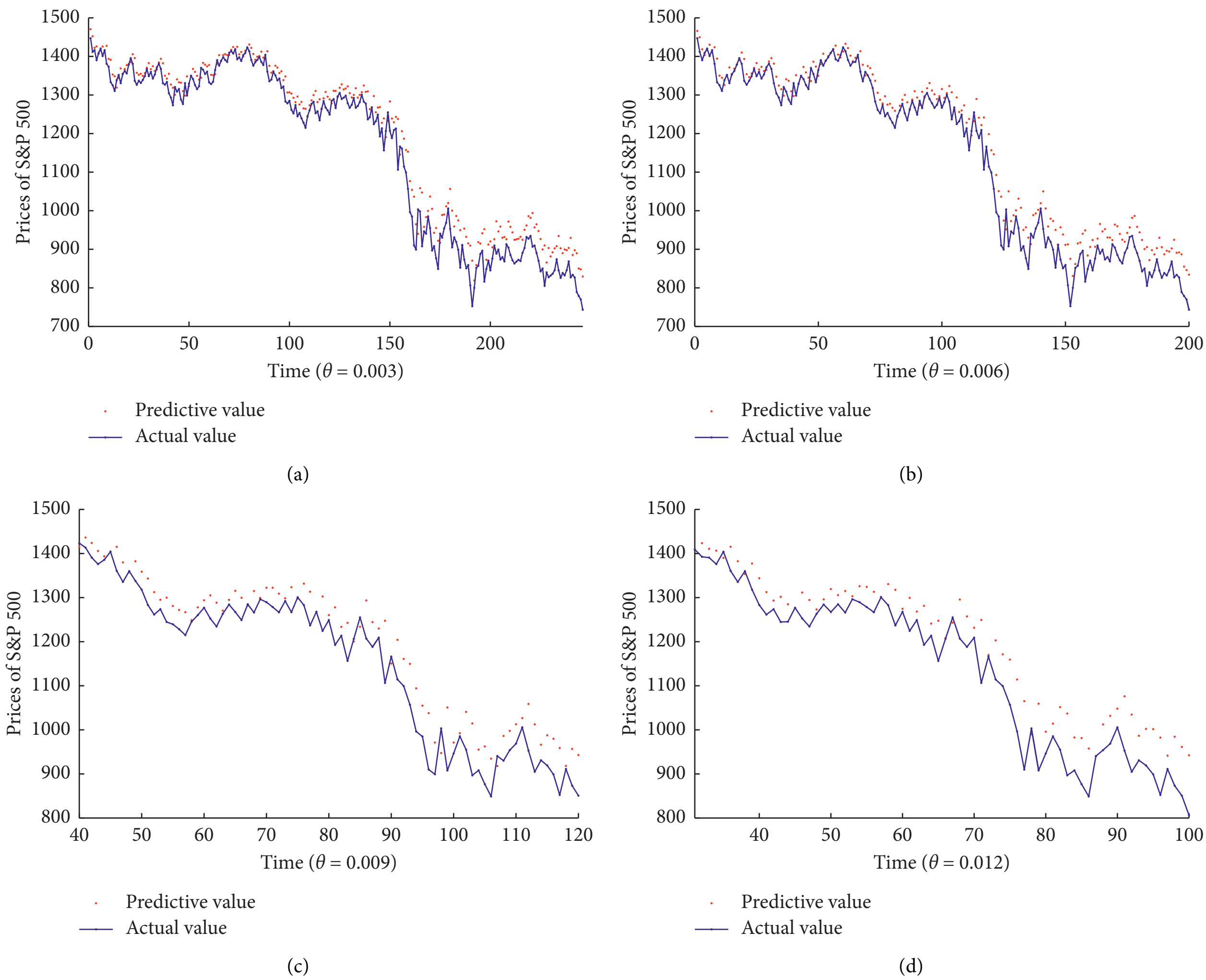


FIGURE 6: Volatility degree prediction of S&P 500 with different thresholds θ.

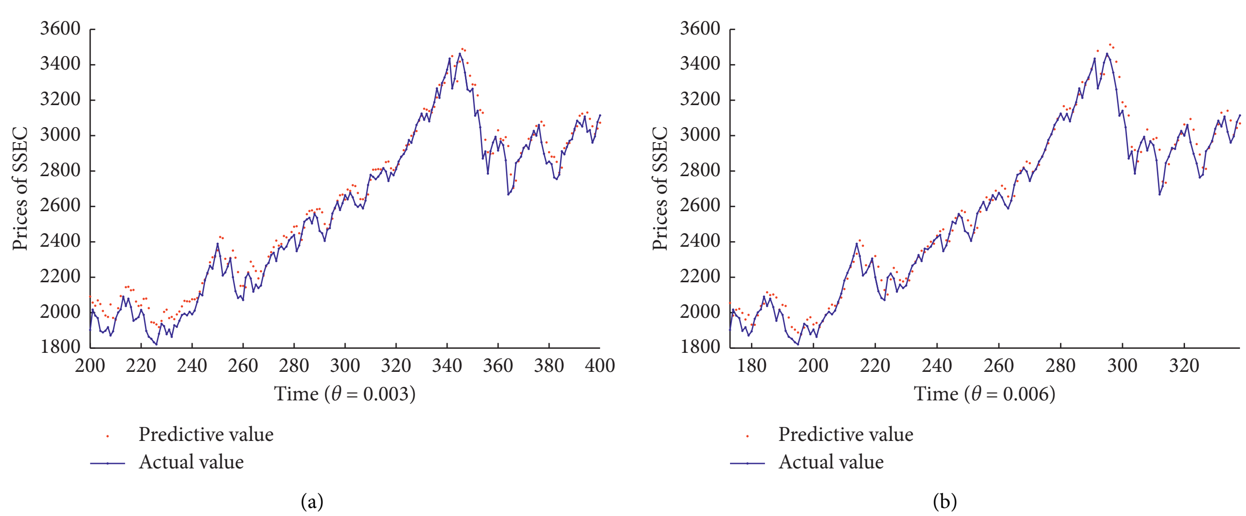
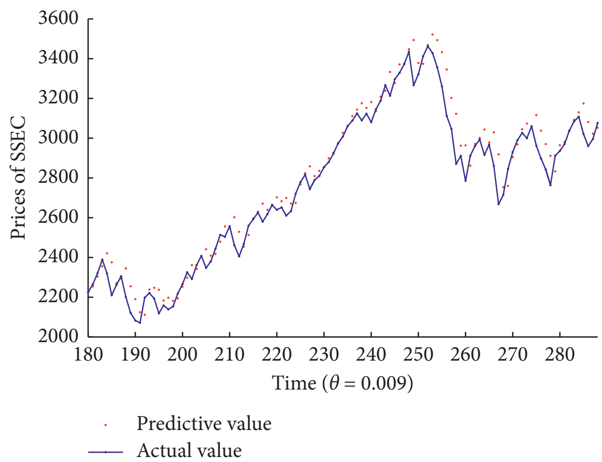
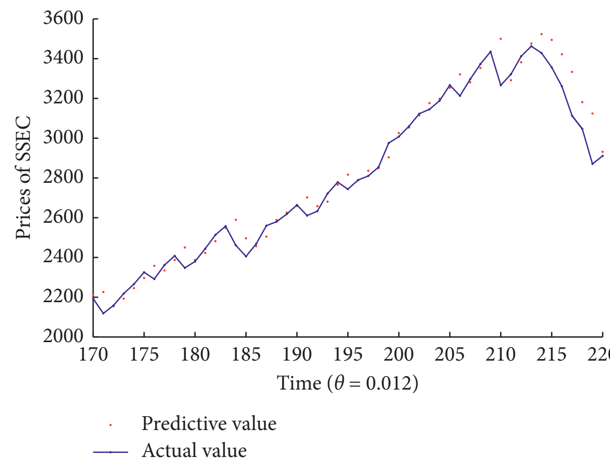


FIGURE 7: Continued.



(c)



(d)

Figure 7: Volatility degree prediction of SSEC with different thresholds θ.

Table 4: Forecasting errors (MAPE) of the FM-Elman ($k = 3$) model as threshold value θ varies.

MAPE	S&P 500	DJIA	SSEC	SZI
$\theta = 0.003$	3.6657	2.9006	2.3593	2.5499
$\theta = 0.006$	4.4312	3.2554	2.3608	2.7811
$\theta = 0.009$	5.3726	3.2998	2.7977	3.1299
$\theta = 0.012$	6.3368	3.7571	2.9498	3.1579

Table 5: Forecasting errors of the FM-Elman model when $\theta = 0$.

Dimension	S&P 500			DJIA		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
$k = 1$	40.3236	3.8834	46.6309	263.665	2.7759	335.7891
$k = 3$	43.5088	4.1692	49.7034	280.9011	2.9379	348.8277
$k = 6$	39.5023	3.8293	46.4366	254.584	2.6649	321.9889
$k = 9$	39.041	3.7597	45.4894	227.0823	2.3703	290.28
Dimension	SSEC			SZI		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE
$k = 1$	47.0132	1.6545	68.2408	222.8488	2.032	293.7208
$k = 3$	50.4433	1.8008	69.7871	228.2216	2.0552	298.6005
$k = 6$	50.8593	1.8233	70.5519	241.7771	2.1825	317.8106
$k = 9$	48.4128	1.7097	68.4333	240.4362	2.1684	316.2495

Table 6: Trend-type forecasting evaluation of the FM-Elman ($k = 3$) model as threshold value θ varies.

Evaluation errors	S&P 500			DJIA		
	DS	CP	CD	DS	CP	CD
$\theta = 0$	48.662	52.7473	43.8445	49.1576	52.3985	45.3961
$\theta = 0.003$	48.5199	52.1327	44.2254	49.2063	52.451	45.4023
$\theta = 0.006$	51.1111	52.9412	49.1039	47.6103	48.9362	46.1832
$\theta = 0.009$	49.011	50	47.9821	47.0998	47.7273	46.4455
$\theta = 0.012$	43.7143	43.9306	43.5028	43.1193	44.6429	41.5094
Evaluation errors	SSEC			SZI		
	DS	CP	CD	DS	CP	CD
$\theta = 0$	50	51.8987	50.9221	51.1879	53.2164	52.2541

Table 6: Continued.

Evaluation errors	S&P 500			DJIA		
	DS	CP	CD	DS	CP	CD
$\theta = 0.003$	50.4459	50.646	50.2513	51.335	51.7677	50.9346
$\theta = 0.006$	51.0109	50.641	51.3595	52.1802	52.439	51.9444
$\theta = 0.009$	52.1154	52.5097	51.7241	52.847	53.7037	52.0548
$\theta = 0.012$	51.3382	50.2538	52.3364	50.211	53.7778	51.9481

to θ with large value. For example, when θ changes from the value 0.009 to 0.012, the values of DS, CP, and CD for stock indexes S&P 500 and DJIA decrease sharply.

5. Conclusion

In this study, we developed an improved Elman recurrent neural network by introducing the factorization machine. Through extensive numerical experiments on the data from stock indexes S&P 500, DJIA, SSEC, and SZI, we demonstrated the effectiveness of the FM-Elman neural network. The prediction accuracy for all financial time series shows that our proposed FM-Elman model outperforms the BP neural network and the original Elman NN. We select training and testing datasets under different volatility degrees, i.e., the threshold value θ varies, to predict. The prediction performance of the FM-Elman model will degrade as θ becomes larger. We also investigate the effect of the user-specified dimension on the prediction performance by the FM-Elman neural model.

The contribution of this work includes the following two points: (1) a technique FM combined with Elman NN to form an FM-Elman neural model for nonstationary analysis which enjoys benefits from both FM and Elman NN and (2) we demonstrate the prediction accuracy in various metrics. The numerical experiments show significant improvements in prediction accuracy over the existing methods. However, the limitation of this research is that the proposed model is data dependent which does not guarantee excellent predictions on all datasets. And further study on high-order interactions among the inputs is also a challenging work.

The power of combining FM with neural network to achieve better performance will likely exist for the area of classification and regression which will be useful for future studies. We believe that FM can be used in conjunction with other deep learning network such as LSTM to form the high quality predicting method. Various combinations in techniques and approaches can be investigated in the future to solve problems occurring in different applications.

Data Availability

The data used to support this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the R&D Program of Beijing Municipal Education Commission (grant no. KJZD20191000401). This research was also supported by the Program of the Co-construction with Beijing Municipal Commission of Education of China (grant nos. B20H100020 and B19H100010) and funded by the Key Project of Beijing Social Science Foundation Research Base (grant no. 19JDYJA001). Fang Wang is grateful for the support by the Beijing Laboratory of National Economic Security Early-Warning Engineering (grant no. B19H100030).

References

[1] K. Oh and K. J. Kim, "Analyzing stock market tick data using piecewise nonlinear model," *Expert Systems with Applications*, vol. 22, no. 3, pp. 249–255, 2002.

[2] Y. Wang, "Mining stock price using fuzzy rough set system," *Expert Systems with Applications*, vol. 24, no. 1, pp. 13–23, 2003.

[3] Y. Wang, L. Wang, F. Yang, W. Di, and Q. Chang, "Advantages of direct input-to-output connections in neural networks: the Elman network for stock index forecasting," *Information Sciences*, vol. 547, no. 8, pp. 1066–1079, 2021.

[4] F. Beritelli, G. Capizzi, G. Lo Sciuto, C. Napoli, and M. Woźniak, "A novel training method to preserve generalization of RBPNN classifiers applied to ECG signals diagnosis," *Neural Networks*, vol. 108, pp. 331–338, 2018.

[5] C. Chen, K. Li, S. G. Teo, X. Zou, K. Li, and Z. Zeng, "Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 14, no. 4, pp. 1–23, 2020.

[6] M. Woźniak and D. Połap, "Hybrid neuro-heuristic methodology for simulation and control of dynamic systems over time interval," *Neural Networks: The Official Journal of the International Neural Network Society*, vol. 93, pp. 45–56, 2017.

[7] J. Zhang, J. Li, and R. Wang, "Instantaneous mental workload assessment using time-frequency analysis and semi-supervised learning," *Cognitive Neurodynamics*, vol. 14, no. 5, pp. 619–642, 2020.

[8] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990.

[9] J. Y. Zheng, "Forecast of opening stock price based on Elman neural network," *Chemical Engineering Transactions*, vol. 46, pp. 565–570, 2015.

[10] B. Wu and T. Duan, "A performance comparison of neural networks in forecasting stock price trend," *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 336–346, 2017.

[11] B. Wu and T. Duan, "The fractal feature and price trend in the gold future market at the Shanghai Futures Exchange (SFE),"

Physica A: Statistical Mechanics and Its Applications, vol. 474, no. 15, pp. 99–106, 2017.

[12] R. H. J. Rani and T. A. A. Victoire, "A hybrid Elman recurrent neural network, group search optimization, and refined VMD-based framework for multi-step ahead electricity price forecasting," *Soft Computing*, vol. 23, no. 18, pp. 8413–8434, 2019.

[13] M. Tripathy, "Power transformer differential protection using neural network principal component analysis and radial basis function neural network," *Simulation Modelling Practice and Theory*, vol. 18, no. 5, pp. 600–611, 2010.

[14] H. Niu and J. Wang, "Financial time series prediction by a random data-time effective RBF neural network," *Soft Computing*, vol. 18, no. 3, pp. 497–508, 2014.

[15] S. Yousefi, I. Weinreich, and D. Reinarz, "Wavelet-based prediction of oil prices," *Chaos, Solitons & Fractals*, vol. 25, no. 2, pp. 265–275, 2005.

[16] L. Huang and J. Wang, "Global crude oil price prediction and synchronization based accuracy evaluation using random wavelet neural network," *Energy*, vol. 151, no. 15, pp. 875–888, 2018.

[17] M. Bildirici, E. A. Alp, and Ö. Ö. Ersin, "TAR-cointegration neural network model: an empirical analysis of exchange rates and stock returns," *Expert Systems with Applications*, vol. 37, no. 1, pp. 2–11, 2010.

[18] R. Dash, S. Samal, R. Dash, and R. Rautray, "An integrated topsis crow search based classifier ensemble: in application to stock index price movement prediction," *Applied Soft Computing*, vol. 85, pp. 1–14, 2019.

[19] K. J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1-2, pp. 307–319, 2003.

[20] X. Y. Qian and S. Gao, "Financial series prediction: comparison between precision of time series models and machine learning methods," 2018.

[21] G. Armano, M. Marchesi, and A. Murru, "A hybrid genetic-neural architecture for stock indexes forecasting," *Information Sciences*, vol. 170, no. 1, pp. 3–33, 2005.

[22] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock market index using fusion of machine learning techniques," *Expert Systems with Applications*, vol. 42, no. 4, pp. 2162–2172, 2015.

[23] J. Wang, H. Pan, and F. Liu, "Forecasting crude oil price and stock price by jump stochastic time effective neural network model," *Journal of Applied Mathematics*, vol. 2012, Article ID 646475, 15 pages, 2012.

[24] J. Wang and J. Wang, "Forecasting energy market indices with recurrent neural networks: case study of crude oil price fluctuations," *Energy*, vol. 102, no. 1, pp. 365–374, 2016.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[26] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, http://arxiv.org/abs/1409.0473.

[27] G. Hinton, L. Deng, D. Yu et al., "Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[28] S. Rendle, "Factorization machines," in *Proceedings of the 2010 IEEE International Conference on Data Mining*, pp. 995–1000, Sydney, Australia, December 2010.

[29] X. He, H. Zhang, M. Y. Kan, and T. S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 549–558, Pisa, Italy, July 2016.

[30] Y. Liu, W. Guo, D. Zang, and Z. Li, "A hybrid neural network model with non-linear factorization machines for collaborative recommendation," *Lecture Notes in Computer Science*, vol. 11168, pp. 213–224, 2018.

[31] Q. Wang, F. Liu, S. Xing, X. Zhao, and T. Li, "Research on CTR prediction based on deep learning," *IEEE Access*, vol. 7, pp. 12779–12789, 2018.

[32] W. Ni, T. Liu, Q. Zeng, X. Zhang, H. Duan, and N. Xie, "Robust factorization machines for credit default prediction," in *Proceedings of the PRICAI 2018: Trends in Artificial Intelligence*, pp. 941–953, Nanjing, China, August 2018.

[33] F. Zhou, H. Zhou, Z. Yang, and L. Yang, "EMD2FNN: a strategy combining empirical mode decomposition and factorization machine based neural network for stock market trend prediction," *Expert Systems with Applications*, vol. 115, pp. 136–151, 2019.

[34] M. Khashei and M. Bijari, "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting," *Applied Soft Computing*, vol. 11, no. 2, pp. 2664–2675, 2011.

[35] L. Cao and F. E. H. Tay, "Financial forecasting using support vector machines," *Neural Computing & Applications*, vol. 10, no. 2, pp. 184–192, 2001.