Proceedings of the 2nd International Conference on Financial Technology and Business Analysis
DOI: 10.54254/2754-1169/57/20230771

Google Stocks Prediction by Machine Learning of RNN and LSTM Techniques

Ruimin Tian[1,a,*]

[1]DeGroote School of Business, McMaster University, Hamilton, Canada
a. email, tianr16@mcmaster.ca
**corresponding author*

Abstract: The objective of this study is to utilize a combined model of two algorithms, namely Long Short-Term Memory network and Recurrent Neural Network, to forecast the stock price of Google. Using Google stock price data from 2010 to 2022 as the training set and performed data preprocessing and feature engineering. This then build a deep neural network model consisting of multiple LSTM and RNN layers and train it by the backpropagation algorithm. During training, this paper employs an appropriate loss function and optimizer to minimize the prediction error. In conclusion, the performance of the model was assessed by employing Google stock price data from 2023 as a test set. By comparing the error between the actual stock price and the predicted value of the model, it can evaluate the accuracy and stability of the model. The experimental results show that the superposition model using LSTM and RNN algorithms can effectively predict the Google stock price with high accuracy and stability. This research presents a practical approach that can enhance the predictive capabilities of investors, financial institutions, and other related domains, enabling them to make well-informed investment decisions in the stock market.

Keywords: Google, RNN, LSTM

1. Introduction

The stock market is a complex system full of volatility and uncertainty, which is affected by many factors, such as the economic environment, political situation, and company performance. Accurately predicting stock prices is a challenging task for investors and traders. Traditional statistical methods and machine learning algorithms have certain limitations when dealing with nonlinear and dynamically changing stock price data. Deep learning technology has made great breakthroughs in recent years and has shown powerful capabilities in many fields. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks are extensively utilized in the field of deep learning for effectively processing sequential data. These models excel at capturing the temporal dependencies and long-term memory inherent in sequential data.

Currently, scholars worldwide have conducted extensive research on utilizing a composite model combining LSTM and RNN algorithms to predict the stock price of Google. For example, Shah et al. used LSTM recurrent neural networks to predict the stock market. They have successfully implemented predictions of future stock prices by feeding historical stock price sequences into the LSTM model [1]. Kumar et al. used the Long Short-term Memory network to predict Google stock price. They used the LSTM model to analyze the historical price data of Google's stock and got a

DOI: 10.54254/2754-1169/57/20230771

good prediction result [2]. Hochreiter and Schmid Huber proposed the architecture of long short-term memory (LSTM). They solve the gradient disappearance problem in traditional recurrent neural networks by introducing a gating mechanism and achieve significant performance improvements on several sequence modeling tasks [3]. Elman introduced a recurrent neural network structure called Elman network, which has a simple feedforward connection and hidden layer state transfer and is widely used in time series modeling tasks [4]. Zhang et al. reviewed the research status of artificial neural networks in the field of prediction. They point out that artificial neural networks have become an effective and flexible modeling method for time series prediction tasks [5].

These papers demonstrate the widespread utilization of LSTM and RNN in stock market forecasting and time series modeling. These studies show that LSTM model has strong memory ability and adaptability, can capture long-term dependencies in time series, and achieve satisfactory results in prediction tasks. However, it is important to note that the model's performance is still influenced by factors such as data quality and feature selection. Therefore, further exploration and improvement are necessary in practical applications.

The objective of this study is to investigate the application of LSTM and RNN algorithms in stacked models for predicting Google's stock price, with the aim of enhancing the accuracy and stability of the prediction outcomes. The empirical process can be summarized as follows: First, construct a stock price prediction framework based on LSTM and RNN superposition model. Second, Collect and sort out the historical trading data of Google stock, and perform data preprocessing, including feature engineering and normalization. Third, Train the overlay model and optimize the model parameters, such as choosing the appropriate network structure, activation function and loss function. In order to evaluate the predictive performance of the stacked model, the test dataset was utilized. Evaluation metrics, including mean square error and mean absolute error, were utilized to gauge the accuracy of the predictions. Fifth, Analyze the advantages and disadvantages of the superposition model, and propose improvement measures, such as introducing other features, adjusting model hyperparameters, etc. Finally, the results show that the algorithm used in this paper performed well.

2. LSTM and RNN

2.1. LSTM

LSTM is a specialized form of recurrent neural network that effectively addresses issues such as gradient vanishing and exploding gradients that arise in traditional RNN models when processing sequential data. By introducing a gating mechanism, LSTM can selectively retain and forget input information, and can learn long-term dependencies.

LSTM comprises three essential components: the input gate, output gate, and forget gate. The input gate regulates the significance of new input information, while the forget gate determines whether to retain prior memory states. Lastly, the output gate controls the impact of output information.

To provide further elaboration, at each time step t, the LSTM model takes in the current input xt and the hidden state ht-1 from the previous time step t-1. Subsequently, computations are carried out to derive the candidate cell state, as well as the input gate it, forget gate ft, and output gate ot for the current time step t. Finally, utilizing these calculated values, both the cell state Ct and hidden state ht at time step t are updated accordingly.

2.2. RNN

RNN, or Recurrent Neural Network, is a type of neural network that has feedback connections, allowing it to process sequential data and retain information from previous steps. The basic structure

DOI: 10.54254/2754-1169/57/20230771

of an RNN involves combining the input at the current time step with the hidden state from the previous time step, and then obtaining the hidden state for the current time step using an activation function.

To explain in more detail, at each time step t, the RNN takes in the current input xt and the hidden state ht-1 from the previous time step t-1. The hidden state ht for the current time step t is computed through specific calculations and subsequently outputted for use in the next time step.The RNN model has problems such as gradient vanishing and gradient explosion, which makes it less effective when dealing with long sequence data.

3. Data Collection and Preprocessing

3.1. Data Collection

For the purpose of conducting a study on predicting Google stock prices, this paper utilized Google stock price data spanning from 2010 to 2023. The dataset was acquired from the publicly available Kaggle dataset. In this study, the data from 2010 to 2022 were employed as the training set, while the data from 2023 were reserved for testing purposes. Basic information of in-sample period is shown in Table 1

Table 1: Basic information of the Google price from 2010 to 2022.

	Open	High	Low	Close	ADJ_Close	Volume
Max	151.25	151.55	148.90	149.84	149.84	592399008
Min	10.97	11.07	10.85	10.91	10.91	9312000
std	35.81	36.22	35.39	35.80	35.80	49576092
mean	49.16	49.65	48.65	49.16	49.16	60175405

3.2. Data Preprocessing

Before training the model, it is crucial to preprocess the raw data to improve the accuracy and reliability of the model. This paper employed several common data preprocessing methods, which are outlined below.

3.2.1. Feature Selection and Analysis

The training set samples may contain numerous features, and in order to streamline the model and enhance computational efficiency, this paper aims to carefully select and analyze the most relevant features. This paper can use domain knowledge and experience to filter out the most influential features.

3.2.2. Visual Processing

Before performing feature selection, this paper can analyze the original data through visualization methods. Observe the relationship between Google's stock price and other features by plotting time series plots, scatter plots, and so on, and check for outliers or missing values. The specific visualization process is as follows:

In the box plot, the year is represented on the horizontal axis, while the Google stock price is depicted on the vertical axis. The boxes in the box plot represent the distribution of the data set between 25% and 75%, and the horizontal line in the middle represents the median. Therefore,this paper can clearly find that the upper and lower edges represent outliers or extreme values. The visualizations is shown in the following Figures 1-5.

DOI: 10.54254/2754-1169/57/20230771

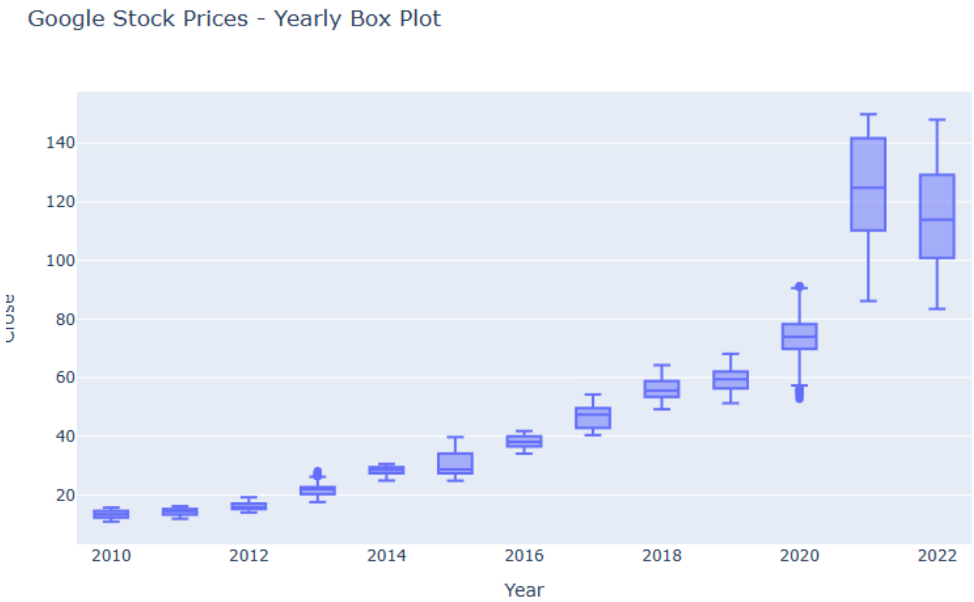

Figure 1: Annual box plot of Google's stock price.

The histogram displays the rate of return on the horizontal axis and the number of samples or frequencies within each interval on the vertical axis.

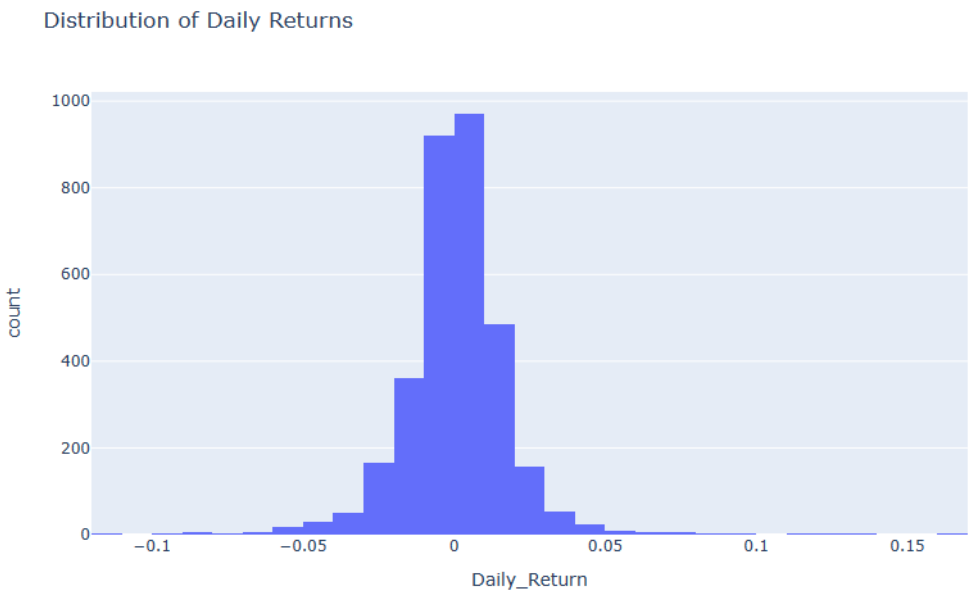

Figure 2: Histogram of daily returns.

In the scatter plot, the close price is depicted on the horizontal axis, while the volume is represented on the vertical axis. Each data point corresponds to a trading day and represents both the closing price and volume.

The scatter plot can be further manipulated as needed, such as adding trend lines, unusual colors for different time periods, and so on.

DOI: 10.54254/2754-1169/57/20230771

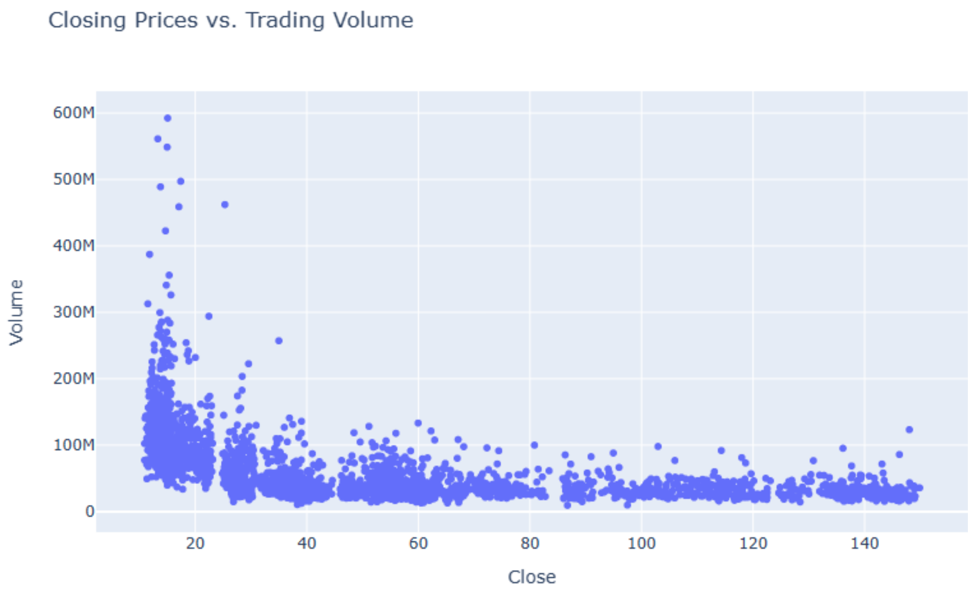

Figure 3: Scatter plot of closing price versus volume.

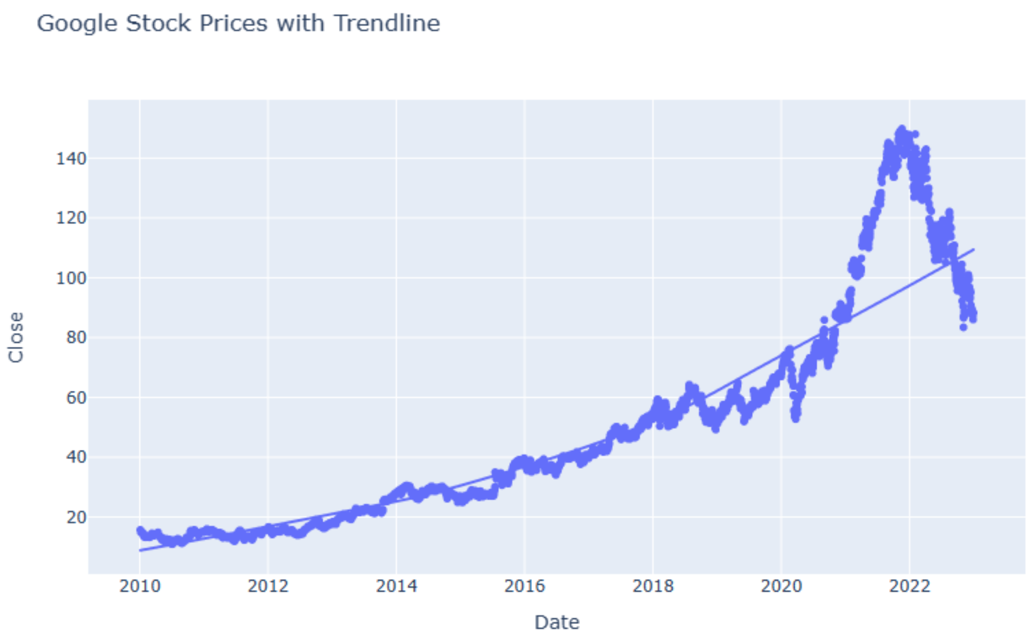

Figure 4: Scatter plot of closing prices (trendline).

By creating a scatter plot of the closing prices and incorporating trendlines, this paper aims to gain a deeper understanding of the distribution of Google stock's closing prices and its long-term trend. This helps us to judge market movements and predict probable future price movements.

To better understand the trend of trading volume in Google stock price, this paper use data visualization method to show the change of daily trading volume, make a line chart and add range sliders.

DOI: 10.54254/2754-1169/57/20230771

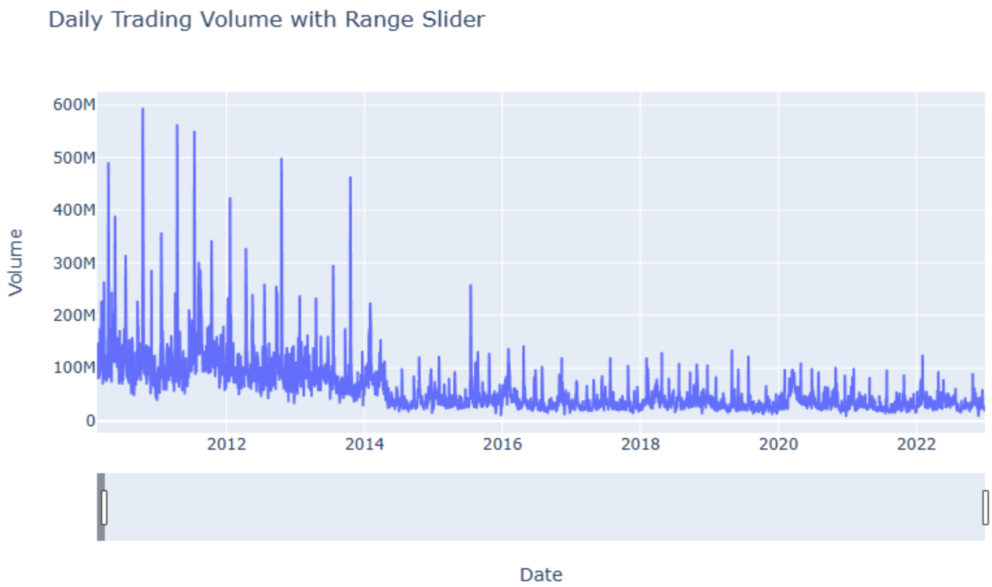

Figure 5: Daily volume chart (with range slider).

By plotting a line chart of the daily trading volume and adding a range slider, this paper can get a better understanding of the trend of the trading volume of Google stock. This helps us judge the level of market activity and investor sentiment and provides a reference for subsequent model training and prediction.

3.2.3. Managing Missing Values

During the data collection process, there may be missing values. This paper can deal with missing values in the following ways: first, this paper can delete samples that contain missing values; The second is to use the interpolation method to fill the missing values. The third is to make reasonable estimations based on domain knowledge.

3.2.4. Data Normalization

Since the range of values may be different for unique features, this paper needs to normalize the data to avoid some features having too much influence on the model training. Commonly employed normalization techniques consist of Z-score normalization and Min-Max normalization. For this particular study, the paper opts for Min-Max normalization as the preferred method to process the data.

4. Model Evaluation and Results Analysis

4.1. Model Structure Design

First, this paper needs to determine the network structure of LSTMS and RNNS. The LSTM model consists of three primary layers: the input layer, the hidden layer, and the output layer. The input layer receives historical stock price data as input features, and the hidden layer contains multiple LSTM units that process time series data and produce output results through the output gate. Finally, the output layer maps the forecast results to the actual share value [6].

DOI: 10.54254/2754-1169/57/20230771

Like the LSTM model, the RNN model is also composed of an input layer, a hidden layer, and an output layer. However, the key distinction lies in the structure of the hidden layer. In contrast to the LSTM model, the hidden layer in an RNN contains a single recurrent unit that connects In each time step, the input of the current time step is combined with the hidden state from the previous time step. This enables information transfer and sequential processing within the network.

4.2. Data Preparation

Before training the model, this paper needs to prepare the data. First, collect historical data on Google's stock price and sort it in chronological order. Then, select the appropriate characteristics as input variables based on the demand, such as the stock price over the past few days, trading volume, and so on. At the same time, it is also necessary to normalize the data to eliminate the scale differences between unique features.

4.3. Model Training

During the model training process, this paper partitions the dataset into a training set and a validation set. To begin, the parameters of the LSTM or RNN model are initialized. Then, through backpropagation algorithm and gradient descent optimization algorithm, the model parameters are constantly adjusted to minimize the prediction error. In each training iteration cycle, this paper uses the training set for forward propagation calculations and calculate the value of the loss function. The model parameters are then updated according to the value of the loss function. This process is repeated until a set stopping condition is reached.

To prevent the overfitting phenomenon from occurring, this paper can adopt some regularization techniques, such as dropout, L1/L2 regularization, and so on. Simultaneously, certain early stopping strategies can be employed to regulate the model's complexity and enhance its generalization ability throughout the training process.

4.4. Model Evaluation and Tuning

After the model is trained, this paper needs to evaluate and tune it. Initially, the model is assessed by employing the validation sets and computing error metrics, such as root-mean-square error (RMSE) and mean absolute percentage error (MAPE), to quantify the disparity between the predicted results and the actual share price. Based on the evaluation results, this paper can adjust the hyperparameters and network structure of the model to further improve the prediction performance.

Furthermore, cross-validation techniques can be utilized to assess the model's stability and its ability to generalize. The dataset is partitioned into multiple subsets, with one subset being designated as the validation set. The remaining subsets are utilized for training and testing the model. Finally, the average error index is computed. Related results are shown in the following Figure 6.

DOI: 10.54254/2754-1169/57/20230771

Figure 6: Line chart visualization of prediction results.

In a line chart, the horizontal axis represents the time of day (trading day), while the vertical axis corresponds to the stock price. This paper plots the predicted broken lines of both LSTM and RNN algorithms and add the actual 2023 Google stock price as well as the predicted broken lines of the reference algorithm GRU to the chart. To further process the line chart, this paper can set colors, add labels, etc., as needed.

Using the Mean Squared Error (MSE) metric to evaluate Google stock price forecasts, this paper conducted a performance comparison between the RNN and LSTM algorithms. The results show that the MSE index is 0.0019 when LSTM algorithm is used for prediction, while the MSE index is 0.0023 when RNN algorithm is used. Therefore, in this task, the LSTM algorithm performs better than the RNN algorithm.

The comparison results reveal that the LSTM algorithm outperforms the RNN algorithm in terms of the MSE index for predicting Google stock prices, that is, smaller average prediction error. This suggests that the LSTM model can more accurately capture the complex relationships of Google's stock price data and provide more accurate predictions [7-10].

In summary, the LSTM algorithm shows better performance than the RNN algorithm in the Google stock price prediction task (See Table 2).

Table 2: Mean Squared Error loss of RNN and LSTM algorithms.

Epoch	50	100	150	200	250	300
RNN Training Mean Squared Error	0.0033	0.0014	0.0013	0.0012	0.0011	0.0011
LSTM Training Mean Squared Error	678.608μ	510.4025μ	423.6636μ	385.8613μ	367.109μ	365.4234μ

5. Conclusions

This paper aims to use LSTM and RNN models to predict Google stock price. Through experiments and analysis, we draw the following conclusions: LSTM and RNN models perform well in the Google stock price prediction task. Compared with traditional time series models and other machine learning algorithms, LSTM and RNN can better capture the long-term dependence and nonlinear characteristics of stock prices and improve the accuracy of prediction. In the process of model training, proper selection of network structure, hyper parameters, and optimization algorithm is crucial to improve prediction performance. Further optimization of the model's performance can be achieved by adjusting parameters such as the number of nodes in the hidden layer, the learning rate, and the number of iterations. The selection and processing of data also has an important impact on the prediction results. In this study, we selected multiple factors including historical stock price, trading volume, market index and other factors as input features, additionally, data normalization was performed to enhance the model's generalization capability.

While LSTM models perform better in some cases, RNN models are still a valid and reliable choice. According to the characteristics of the specific problem and data set, the appropriate model can be flexibly selected for prediction.

In conclusion, this paper utilizes LSTM and RNN models for the prediction of Google's stock price, yielding favorable outcomes in the experimental analysis. These research results have important reference value for financial market participants, investors, and related researchers, and provide certain guidance and reference for future stock price prediction. However, there are still many aspects that can be further improved and expanded, such as introducing more features, optimizing the model structure, etc. It is hoped that the research results of this paper can provide latest ideas and methods for further research in related fields.

References

[1] *Shah, A., Zhang, Z., and Ahmad, N. (2017) Stock market prediction using LSTM recurrent neural network. International Journal of Business Information Systems, 25(3), 366-380*
[2] *Kumar, A., Bharti, S. K., and Rani, R. (2018) Google stock price forecasting using long short-term memory (LSTM) networks. Journal of Computational and Theoretical Nanoscience, 15(12), 6625-6631.*
[3] *Hochreiter, S., and Schmid Huber, J. (1997) Long short-term memory. Neural computation, 9(8), 1735-1780.*
[4] *Elman, J. L. (1990) Finding structure in time. Cognitive science, 14(2), 179-211.*
[5] *Zhang, G., Patuwo, E.B., and Hu, M.Y. (1998) Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, 14(1), 35-62.*
[6] *Yao, X., and Liu, Y. (2000) A new evolutionary system for evolving artificial neural networks. IEEE Transactions on Neural Networks, 11.*
[7] *Huang, G.B., and Babri, H.A. (2014) An empirical study on CAC40 and Dow Jones Industrial Average using extreme learning machine. Applied Soft Computing, 18.*
[8] *Wang, J., and Xu, X. (2014) Forecasting stock indices using radial basis function neural networks optimized by an improved particle swarm optimization algorithm. Expert Systems with Applications, 41.*
[9] *Guo, H., and Zhang, X. (2015) A novel hybrid model for stock price prediction. Applied Soft Computing, 37.*
[10] *Chen, J., and Huang, L. (2013) The prediction of stock markets based on fuzzy time series. Applied Soft Computing, 13.*