# Chinese Futures Forecasting System Based on Recurrent Neural Network

**By**

**Youwei Huang**

**S1296394@monmouth.edu**

**Available[Online]:**

<https://github.com/devilyouwei/WallStreet-New>

**Advisor**

**WenJin Zhang**

**An essay for the final project of**

**[Deep Lrn with Python](https://monmouth.desire2learn.com/d2l/home/273200" \t "/home/devil/Documents\\x/_blank)**

**Department of Computer Science and Software Engineering**

**Monmouth University**

**West Long Branch, New Jersey**

**August 2020**



# Abstract

The purpose of financial forecasting is to analyze the financial historical data, build a prediction model, and predict the trend of future data. In this paper, the latest deep learning results and financial forecasting are combined systematically and innovatively, and the method of using recurrent neural network to predict the change of financial data is proposed. This paper first introduces the breakthrough achievements of artificial intelligence in recent years, designs the system based on RNN related technology, and then shows the prediction effect of the system through the experimental group, and improves the model algorithm through the evaluation of the experimental results and the actual target results. The system evaluates the prediction results of the model trained by the historical data of each commodity separately, which is close to the actual data and has good prediction effect. It is concluded that it is feasible to use the recurrent neural network to study and analyze financial data for trend prediction, which has research value and development potential.

**Key words: deep learning; neural network; RNN; big data; financial forecasting**

# Introduction

The concept of artificial intelligence was proposed as early as 1956. It has experienced more than half a century of development, until the artificial neural network theory was put forward, and Google first published deep learning technology, and through the form of "man-machine war" [1], people can really see the powerful ability of deep learning.

Deep learning is different from the traditional artificial intelligence method [3]. It adopts "reverse deduction thinking", based on huge data, with the help of the characteristics of neural network autonomous learning (the process of neural network building connection relationship), through repeated learning of historical data, and continuous improvement and optimization, an optimal model is fitted. Among them, RNN (recurrent neural network) is suitable for the learning of data characteristics on time series, and financial data has sequence. The historical data characteristics can be fitted by RNN as the prediction model, and the model can be continuously optimized in the prediction process. With the accumulation of data, the model tends to be optimal in theory.

Internationally, financial forecasting has been highly valued in the academic, financial and people's daily life. The Nobel Prize for economics was awarded to two economists who successfully applied the autoregressive condition difference and vector autoregressive prediction model to financial time series prediction in 2003 and 2011 respectively.

# Related Work

In China, there are not only financial forecasting based on economics and its improved forecasting methods, such as the financial forecasting method based on grey linear regression combination model, which has the characteristics of less data demand and accurate prediction model [5], Markowitz combination theory has weakened the mystery of market risk [6], but also prediction methods and models based on Mathematics and statistics, such as hidden Markov The improvement of Markov model and its application in financial forecasting [7], financial forecasting based on Bayesian maximum likelihood estimation [8].

In recent years, artificial intelligence method combined with computer technology has been proposed for financial forecasting. Among them, financial forecasting technology based on deep learning is the most active, and most of them are based on convolutional neural network (CNN) Network, referred to as CNN [9] and its improved algorithms, such as financial forecasting method based on the coupling of genetic algorithm and neural network [10], fepa model [11], financial forecasting model based on generalized regression neural network [12], and so on.

There are also applications of traditional forecasting methods, such as SVM method [13], EMD method [14], etc. Both the traditional forecasting method and the financial forecasting method based on convolutional neural network are very innovative, and their prediction accuracy and efficiency can achieve the expected effect in most environments.

In the above situation, through the traditional prediction methods and CNN prediction, such as building a model on the historical data of long time series, the model has a low dependence on the time relationship of data. This system will creatively adopt LSTM (long-term and long-term memory neural network) of RNN to realize financial forecasting model. Its characteristics are: the prediction model is to fit the change characteristics of historical data influenced by each other on the time line, rather than the data itself, that is, the change of the current day's data is related to yesterday's data, yesterday's data is related to the previous day's data, and the previous day's data is also related It is more related to the recursive analysis. This method makes full use of the characteristics of time series, correlation and recurrence of financial data changes.

# System Design

RNN (cyclic neural network) model design includes data loading, model construction, data training and evaluation model design.

## data loading

Data type: financial data. For example, futures and stocks usually offer the following four prices: the highest price, the lowest price, the opening price and the closing price. Since the highest price and the lowest price already include the opening price and closing price, and because the opening price and closing price are accidental prices in one day's data, the training data should choose the highest price or the lowest price as the input data. In order to speed up the data training process, the input data type is set to use the highest price.

Data window: corresponding to the sliding window in RNN, i.e. to predict the sequence length of past values to be input. The window size needs to be adjusted according to the RNN model. Theoretically, if the window is too large, the data change characteristics are more sensitive to the market random noise over time, and the data that is too far away from the predicted target value has less significance to the target; if the window is too small, it will lead to over fitting of deep learning, and the prediction results have no actual reference value.

The loadable data types are shown in Table 1.

Table 1 available data types of input layer

|  |  |
| --- | --- |
| Attributes | Meaning |
| DATE | Index |
| CLOSE | Closing Price |
| HIGH | Highest Price |
| LOW | Lowest Price |
| OPEN | Open Price |
| VOLUME | Volumes |

## Model construction

RNN includes input layer, output layer and hidden layer. Its characteristic is that each layer has time feedback loop, and the layers are composed of superposition [16]. Its network structure is shown in Fig. 1.

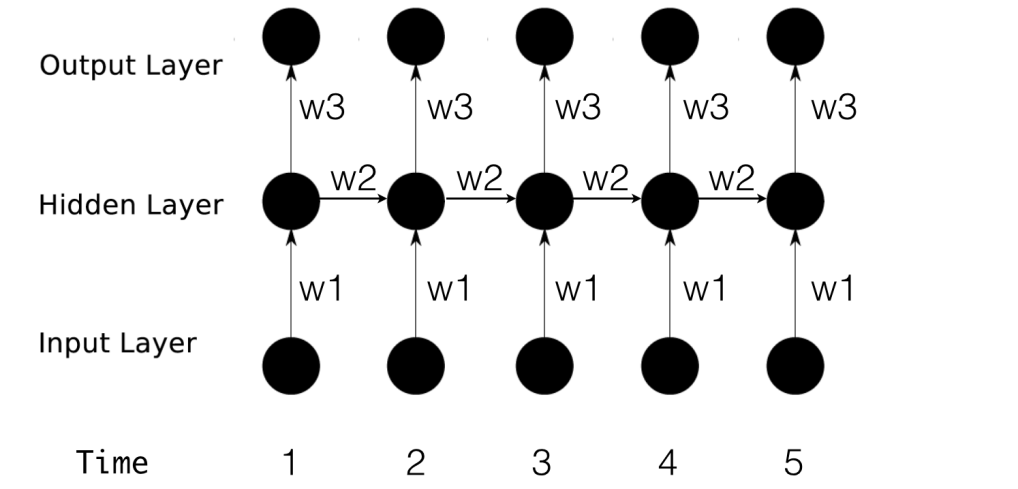


Fig. 1 Relationship of RNN multilayer neural networks

According to figure 1, the neural network structure is designed as follows: set the date (serial number) as T and the price as P [t], and the model is divided into three steps to predict P [T + w]. Slide window P [t T + W-1], where t ∈ (1, tmax-w + 1), and t ∈ n, W is the set window size, P matrix is used as input matrix, which is recorded as input (T); the last hidden layer hide (t-1) is connected; hide (T) of current hidden layer is generated, and the predicted value p [T + w] is taken as the input of output layer, and the predicted result is output (T).

Dropout design: in the process of deep learning, over fitting is a serious problem. Dropout is a technology to solve this problem. The key idea of dropout is to randomly discard units (and their connections) from neural networks during training [17]. Financial data, especially futures and stocks, are likely to rise and fall sharply due to market environment. Therefore, dropout function should be added to the output layer to remove the impact of huge price fluctuations. At the same time, because the loop in RNN will amplify the noise and disturb the learning, dropout is used for acyclic connection.

## Training and evaluation program

The RNN neural network model built in Chapter 1.2 is set as LSTM (), and LSTM () is used for data prediction training and model evaluation scheme is given. The specific steps are as follows:

(1) The training data group and the test data group were separated.

(2) The separated data group is divided into window and target.

(3) Configuration of training parameters, including learning rate, cycle times and batch batch.

(4) Design optimization scheme: define the loss function and optimizer. The optimizer adopts Adam, and the loss function formula is as follows:

（1）

(5) LSTM (window array, target value, training times, batch) was trained.

(6) The evaluation algorithm is used to evaluate the error between the actual value in the test data group and the predicted value given by LSTM (). The lower the error is, the better the model is.

(7) Save the model file.

The training process design is shown in Figure 2. The training prediction value and LSTM RNN in the figure represent the self optimization operation in the training process. After the predicted value generated by self training is recovered, the scheme optimization is carried out in step (4).

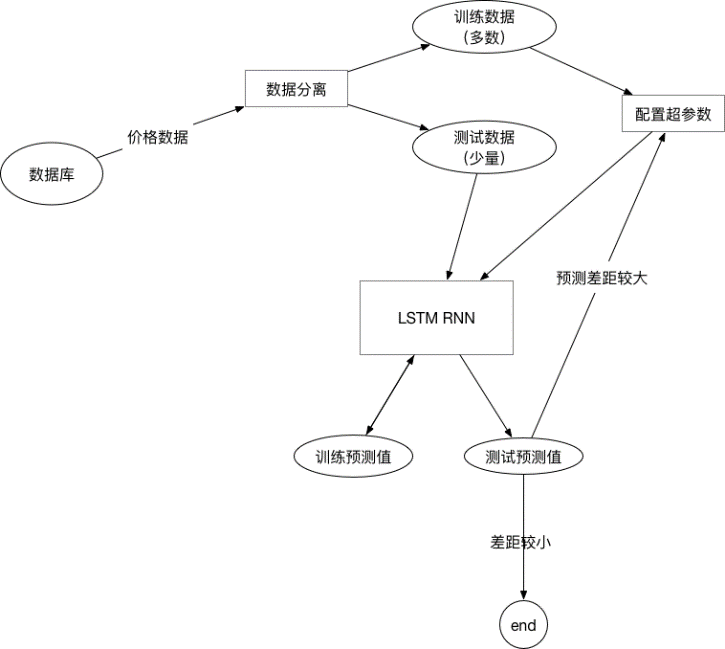


Fig. 2 training process

The mapping relationship between window data and target data is shown in Figure 3.

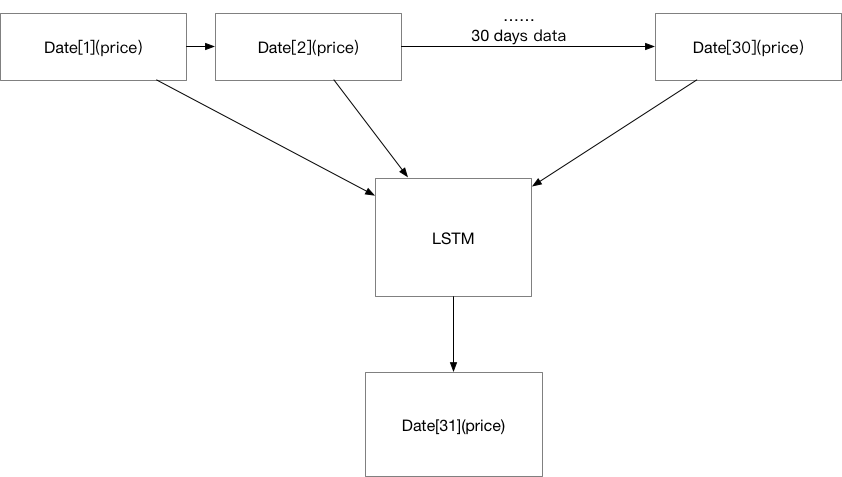


Fig. 3 mapping relationship between window and target

# Key technologies

## Prediction of technical feasibility

There are three assumptions that financial forecasting technology feasibility needs to satisfy:

1. market behavior digests all behaviors;
2. price will evolve in the form of trend;
3. history will repeat itself in the end. Only when the above conditions are met can the financial forecast be studied.

## LSTM Technology

LSTM (long-term and long-term memory network) is an improvement to solve the long-term dependence problem based on RNN. Cell is introduced into LSTM and encapsulated in block. Each block contains input gate, forgetting gate, output gate and cell [19]. The function of LSTM block is to prevent forgetting long-term nodes in the sequence and increase the depth of NN. The structure of LSTM block is shown in Fig. 4.

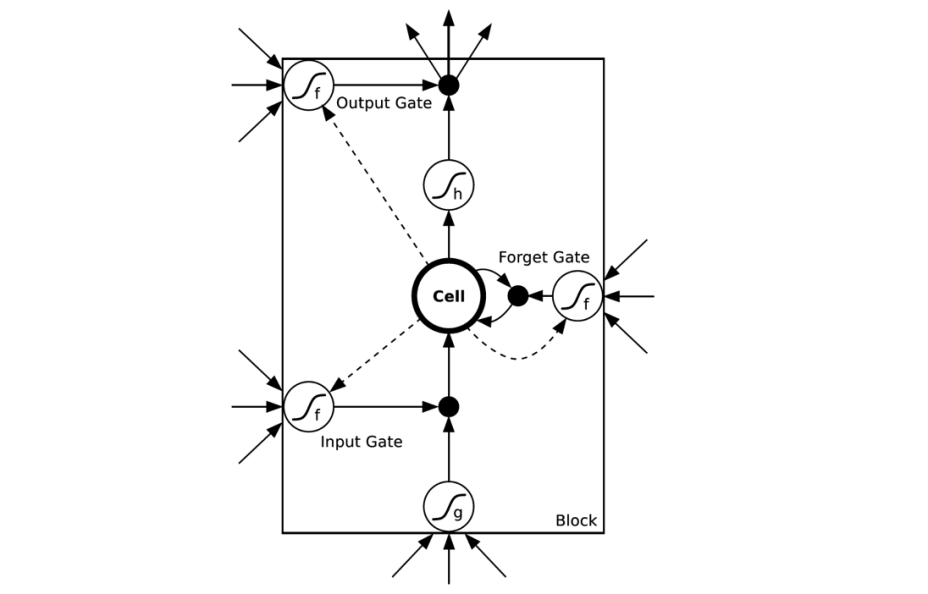


Fig. 4 LSTM block structure

The characteristics of large amount of financial data, rapid change, large and many sliding windows require the use of LSTM technology.

## Data visualization technology

For the neural network training process, visual graphics are made to show the original data, training prediction data and test prediction data by linear graph. Python data analysis library pandas or web technology is used to visualize the data to browser canvas.

## Data crawler technology

The model training of recurrent neural network is infinite. Internet finance will produce new commodity price, transaction volume and other data every day. The system needs to update the historical data in the database regularly to achieve better prediction effect. In this paper, through python node.js It also provides the visual crawler management function and data management function, responsible for data acquisition, sorting and storage.

# Core algorithm

The implementation of neural network based on RNN needs to load the data crawled from the Internet and remove the invalid data. Before training, the data is separated, including training data and simulation prediction data, as well as the window data and the target data to be predicted in the training process. The window data is a two-dimensional array and the target data is a one-dimensional array. In the evaluation model stage, the target array is compared with the prediction array, and the evaluation results are given.

## Optimal crawler algorithm

The crawler module adopts the optimal crawler algorithm, that is, only the commodity data within the specified range can be crawled, so as to minimize the amount of redundant data. Assuming that the data type and data volume have been standardized, the optimal crawler algorithm is used to filter out the specified goods, commodity attributes and their data types, and crawl and store them. The following are the key codes to implement the algorithm:

1. Read database restricted items.

Define object array: Commodity [];

Product [] = SQL query (select \* from goods);

While commodity in commodity []

Data = waiting for execution: crawler (commodity);

If (! Data | data. Length = = 0) skips the crawler;

Else wait for execution: save (data); save (data);

Return, end;

(2) Crawler function implementation.

Define address: address = the optimal address specified;

Define function: asynchronous: crawler (commodity)

Data = SuperAgent (address). Query (product. API number). Httpget();

Return data;

The super agent in the algorithm is Node.js SQL database query is driven by MySQL database.

## Incremental crawler strategy

The crawler module uses incremental crawler, which requires that duplicate data can not be crawled. Therefore, it is necessary to implement a re access strategy for the crawler system, which is incremental rather than reset mode. In order to realize incremental data comparison, the key codes are as follows:

Array: local data volume = SQL query ("select count (\*) as count from futures where G\_ id=? order by date asc",g\_ id);

Array: API data [] = crawler (commodity);

Variable: API data volume = API data []. Length;

Variable: update number = 0;

If (local data volume = = API data volume) skips the crawler;

else

While data in API data []

If (the number of cycles is less than or equal to the local data volume), skip saving;

else

To save (data); wait for execution;

Update number + +;

end;

In the above code, the API data volume and the local data volume are integer data. After comparison, if the two are equal, it means that the data in the local database has been updated and the crawler is skipped. If it is found that the data is not equal, it is ready to insert data, traverse the API data array, and save until the number of cycles is greater than the local data volume, so as to avoid data duplication :

API data volume - local data volume > 0, crawling

API data volume – local data volume < = 0 | API data volume = 0, skip

## LSTM neural network algorithm

In Section 2.2, LSTM is an improved RNN algorithm of threshold type, and the block structure diagram is given. Its nodes include input gate, output gate and forgetting gate, which can change the weight of self circulation. When the model parameters are fixed, the integral size at different times can be changed dynamically, thus avoiding the problem of gradient disappearance or gradient explosion. The calculation formula included in LSTM cell is as follows:

（2）

（3）

（4）

（5）

（6）

（7）

Formula (2) is the forgetting threshold calculation formula, formula (3) is the input threshold calculation formula, formula (4) is the previous cell state calculation formula, formula (5) is the current cell state calculation formula, formula (6) is the output threshold calculation formula, formula (7) shows the calculation formula of the final output result of the current cell. All formulas are encapsulated in the node of LSTM. When the financial data from the system passes through the node, the relevant formula will be used for calculation. According to the size and number of windows, multiple corresponding cell states are generated in time series, and the former state can affect the next state. Where tanh in the formula is the activation function.

The output function is changed to the next nonlinear level.

By training multiple groups of data, W and B variables between nodes can be optimized. This process needs to be realized by cost function.

Cost function: evaluate the specific input value quantitatively, calculate the difference between the value and the real value, so as to adjust the weight between layers and reduce the loss value. This process is called reverse courier, and the smaller the loss value, the more accurate the result [20].

## Data loading implementation

The "data loading" of neural network refers to reading and sorting the database. The specific steps are as follows: introducing the driver, loading the database configuration file, obtaining the specified commodity price, removing the interference data (non trading price), and temporarily storing it in the array in the form of float32

## Window and target data separation algorithm

The training implementation requires separating window data and target data from the array data obtained in SQL. The specific algorithm is as follows:

Definition function: window separation (data set, window size)

Window array, target array = array []], array [];

Loop (Times: data length window size)

The window array is increased (dataset [cycles] to dataset [cycles + window size]);

The target array is increased (dataset [cycles + window size + 1]);

Ends;

Return window data array and target data array;

## Separation algorithm of training and test data

In order to compare the test data and the training data, the specific algorithm is as follows:

Define constant: training data percentage:

Definition: training data length = data set length × training data percentage

Test data length = data set length - training data length

Training array, test array = data set [start position to training data length], data set [training array length to data set length]

Definition constant: window size = (system uses 30 days)

Training window array, training target array = window separation (training array, window size)

Test window array, test target array = window split (test array, window size)

## LSTM data training algorithm

For the above input data, LSTM requires the matrix operation format algorithm as follows:

Training data = format (training window data, training window data length, 1, window size);

The specific functions and algorithms are as follows:

Definition function: LSTM (training data, target data, number of cycles, batch);

Define variable: input weight;

Define variables: input bias;

Input operation = training data \* input weight + input bias;

Create LSTM neurons (number);

Initialize zero value (batch);

Output data = received tensor;

Define variable: output weight;

Define variables: output bias;

Predicted value = output data \* output weight + output bias;

DROPOUT()；

Return the predicted value;

## Model evaluation algorithm

After training, the model is evaluated and improved, and the evaluation algorithm is as follows:

Training prediction array = LSTM () = > prediction (training window array);

Test prediction array = LSTM () = > prediction (test window array);

Format data (training prediction array, test prediction array);

Training data evaluation =;

Test data evaluation =;

Print (training data evaluation, test data evaluation);

Visualization;

# Algorithm improvement experiment

In this paper, different parameters in the algorithm are adjusted, and a number of experiments are carried out, and the algorithm is improved through comparative experiments.

## Training times control experimental group

With the increase of training times, the prediction results of RNN (recurrent neural network) will be closer to the target results, but with the increase of training times, the prediction performance will not improve to a certain extent and tend to saturation. As shown in Fig. 5-8, the prediction results of different training times are given.

|  |  |
| --- | --- |
| Fig. 5 training once | Fig. 6 training for 10 times |
| Fig. 7 100 times of training | Fig. 8 1000 training times |

In Figure 5 to figure 8, the blue line represents the original price, the yellow line represents the training forecast price, and the green line represents the simulation test forecast price.

Among them, when the training times are 1, the prediction results are seriously divorced from the target data; when the training times are 100, the prediction results are closer to the target results, which is significantly improved compared with one training; when the training times are 1000, the prediction effect is not significantly improved compared with 100 training times, indicating that the prediction effect has been saturated and there is no need to increase the training amount.

## Sliding window experimental control group

In LSTM, too large sliding window will lead to the decline of prediction accuracy. The reason is that the noise generated by too far data is too much, and the mapping relationship with the target data is small. Moreover, too large window will lead to too few training windows and insufficient training amount, which will lead to over fitting problems. However, if the sliding window is too small, the correlation between the data in the time series will be reduced, which will lead to the problem of over fitting of training, making the prediction curve too close to the target data curve, so that the prediction results have no practical reference value. As shown in Figure 9-11, the results of sliding window for 100 days, 1 day and 30 days are shown respectively.

|  |  |
| --- | --- |
| Fig. 9 window: 100 days | Fig. 10 window: 1 day |
| Fig. 11 window 30 days |  |

The experimental results show that 100 days as the sliding window, the prediction effect is significantly reduced; 30 days as the sliding window; the prediction results have certain reference value; taking 1 day as the sliding window, the predicted data almost coincide with the actual target data, resulting in the prediction over fitting problem, and the predicted data has no actual reference value. It is concluded that 30 days as the sliding window to predict the next day's price is more appropriate, the forecast trend curve is basically close to the real curve, and the price difference is low, which has certain reference value.

# Conclusion

In this paper, RNN is used to construct and train the financial big data prediction model. Finally, the experimental results are obtained: 30 days as the sliding window, the price of the 31st day is predicted, the training times are 100, and the training data volume is 60%. Different financial products are trained separately to generate training memory files. After the detection of evaluation algorithm, the predicted values of some commodities are close to the target value. For example, the evaluation value (average price difference) of gold prediction model is 2-5 yuan, which is more appropriate. However, at present, the evaluation value (average price difference) of some commodities is as high as 100-1000 yuan, so it is lack of reference. In the future, it is considered to adjust the super parameters separately according to the different characteristics and data volume of different commodities, and consider introducing more input parameters, such as transaction volume, commodity trade data, relevant industry data, etc., which will affect the trend of commodity price and improve the accuracy of prediction.

Through the research on the prediction method of financial data by using the recurrent neural network, we can deepen the understanding of the principle of deep learning, and the conclusion of the experiment makes people realize the applicability and practicability of RNN in the prediction of financial data. In the future, with the development of computer hardware technology, the performance improvement will promote the improvement of deep learning efficiency. Therefore, more dimensional data types and larger amount of data can be used as the input parameters of neural network modeling process to provide more reliable, stable and accurate prediction ability for neural network. The data analysis method combining deep learning and big data will gain higher recognition in the financial field.

# Citations

[1] Li Haowei. Man machine war and the future development of artificial intelligence [J]. Examination weekly, 2018（40）：192-193.

[2] Yang Liu. Big data coordination in the Internet era [J]. China's strategic emerging industries, 2018（20）：112.

[3] Dong Xin. Construction of future artificial intelligence autonomous learning network [J]. Electronic technology and software engineering, 2018 (3): 252

[4] Zhang guiyong. Application of improved convolutional neural network in financial forecasting [D]. Zhengzhou: Zhengzhou University, 2016

[5] Lu Yang. Financial forecasting method based on grey linear regression combination model [J]. Statistics and decision making, 2017 (10): 91-93

[6] He Jian. Application of statistical methods in risk management and financial forecasting [J]. Shopping mall modernization, 2006（32）：368-369.

[7] Xu Zhujia, Xie Rui, Liu Jia, et al. Improvement of hidden Markov model and its application in financial forecasting [J]. Journal of engineering mathematics, 2017,34 (5): 469-478

[8] Shen bin, Wen Tao. Financial forecasting based on Bayesian maximum likelihood estimation [J]. Statistics and decision making, 2018,34 (7): 85-88

[9] Zhou Feiyan, Jin Linpeng, Dong Jun. a review of convolutional neural networks [J]. Acta Sinica Sinica,

2017，40（6）：1229-1251.

[10] Song Xiaoyong, Chen niansheng. Financial forecasting system coupled with genetic algorithm and neural network [J]. Journal of Shanghai Jiaotong University, 2016,50 (2): 313-316

[11] Zhang Chengzhao. A deep learning model for financial market forecasting: fepa model [D]. Chengdu: University of Electronic Science and technology, 2016

[12] Liao Wei, Feng Xiaobing, Cao Weiying, et al. Research on financial forecasting model based on generalized regression neural network [J]. Business times, 2010 (7): 42-43

[13] Li Xiangfei, Zhang Zaisheng. SVM financial time series prediction method based on error synchronous prediction [J]. Journal of Tianjin University: Natural Science and Engineering Technology Edition, 2014,47 (1): 86-94

[14] Wang Fang. Research on Financial Volatility Based on EMD and STSA [D]. Tianjin: Tianjin University,

2016.

[15] Zhu Xingjia, Li Honglian, LV Xueqiang, et al. An improved attention based LSTM feature selection model [J]. Journal of Beijing University of information technology: Natural Science Edition, 2018,33（2）：54-59.

[16] Yang Yiyue, Fu Qian, Wan Dingsheng. Time series prediction model based on deep cycle neural network [J]. Computer technology and development, 2017,27 (3): 35-38,43

［17］ SRIVASTAVA N，HINTON G，KRIZHEVSKY A，et al. Dropout：a simple way to prevent neural networks from overfitting［J］. Journal of Machine Learning Research，2014，15（1）：1929-1958.

[18] Huang Shenggen. Analysis of futures investment technology [J]. Capital market, 2013 (6): 126-127

[19] Tang Yin. Improvement of long and short term memory neural network model [J]. Times finance, 2016(24): 1281-282.

[20] Li Jiaxuan. Tensorflow technology analysis and actual combat [M]. Beijing: People's Posts and Telecommunications Press,