Application of LSTM Neural Network in Forecasting Foreign Exchange Price

Yaxin Qu¹, Xue Zhao*⁺
¹College of Electrical and Control Engineering, North China University of Technology, Shijingshan District, Beijing, China
⁺Corresponding Author: Xue Zhao; email: 1219220019@qq.com; phone:18813055195

Abstract. LSTM neural network and RNN neural network models in deep learning are used to forecast the price of foreign exchange financial time series. The existing foreign exchange price and technical analysis indexes are taken as input parameters. By comparing the evaluation indexes of two deep learning models, the optimal neural network model is selected. The experimental results show that the LSTM neural network model has smaller root mean square error (RMSE) and mean absolute error (MAE) than the RNN network model, and the predicted price is more accurate.

1. Introduction
Quantitative trading regularizes, varies, serializes and models traditional trading concepts. It uses computer technology to select various "probabilistic" nodes from a large number of historical data, thus formulating new investment strategies, and forming a complete operating system, which is automatically executed by the computer in the real disk[1]. Due to the rapid development of quantitative trading and computer technology, quantitative models have become a powerful tool to predict the market and guide investment.

There are also some classical cases at home and abroad to predict prices by using neural network technology. Overseas, In 2014, Shinsuke Kimura, Kunikazu Kobayashi, Masanao Obayashi and others used restricted Boltzmann machines to construct time series prediction models for deep confidence networks[2]. In 2016, Matthew Dixon, Diego Klabjan, and Jin Hoon Bang et al. put forward the financial market prediction based on deep learning deep neural network[3]. In China, in 2015, Sun Rich used BP neural network, RNN neural network and LSTM neural network to forecast the stock price in short term and compared the accuracy of the model. At the same time, according to the different characteristics of Chinese and American stock markets, the practicability and accuracy of the model were verified[4]. In 2016, Bai Kaimin et al. proved that the single hidden layer neural network and the deep learning model have certain prediction ability for stock index futures, and the deep learning model has better prediction ability than the single hidden layer neural network. According to the results of the model, there are certain arbitrage opportunities[5]. In 2018, Ren Jun and others proved that the modified LSTM model of elastic network regularization has a good prediction effect on the Dow Jones index[6].

2. RNN Neural Network Model and Its Characteristics

2.1 Principle of RNN Neural Network Model
Circulating neural network (RNN) is a kind of neural network specially dealing with time series
problems. It can extract information of time series, allow information persistence, and use previous knowledge to infer follow-up patterns[7]. Figure 1 is a RNN model structure with a simplified left-hand structure on the right. Figure 2 shows the structure of RNN.

![Figure 1. RNN Neural Network Model Structure.](image1)

![Figure 2. RNN deployment structure.](image2)

The main characteristic of RNN is that it can deal with uncertain input and get certain output. However, due to the increase of information association, RNN will not be able to learn the relationship between information, and then lose all its influence. In order to deal with the forgetfulness of RNN, LSTM structure has been created.

3. LSTM Neural Network Model and Its Characteristics

3.1 LSTM Neural Network Model

In practical applications, RNN is difficult to deal with long-distance dependence. An improved cyclic neural network, Long Short Term Memory Network (LSTM), is proposed. It establishes a long time delay between input, feedback and gradient explosion prevention, and successfully solves the shortcomings of the original cyclic neural network.

In addition to the external RNN cycle, the LSTM cycle network also has an internal "LSTM cell" cycle (self-loop). Its most important component is the state unit. The time constants associated with it are controlled by the forget gate (time T and cell i). The sigmoid unit sets the weight between 0 and 1:

$$f_i^{(t)} = \sigma(b_f^{i} + \sum_j U_{i,j}^{f}x_j^{(t)} + \sum_j W_{i,j}^{f}h_j^{(t-1)})$$  \hspace{1cm} (1)

Among them is the current input vector, which is the current hidden layer vector, including the output of LSTM cells. They are bias, input weight and circular weight of forgetting gate. Therefore, the internal state of LSTM cells is updated in the following way, in which there is a condition of self-ring weight:

$$s_i^{(t)} = f_i^{(t)}s_i^{(t-1)} + \sigma(b_f^{i} + \sum_j U_{i,j}^{f}x_j^{(t)} + \sum_j W_{i,j}^{f}h_j^{(t-1)})$$  \hspace{1cm} (2)

Among them, the bias, input weight and cycle weight of forgetting gate in LSTM cells are calculated. The external input gate unit is updated in a way similar to the sigmoid (which obtains a value between 0 and 1), but with its own parameters:

$$g_i^{(t)} = \sigma(b_g^{i} + \sum_j U_{i,j}^{g}x_j^{(t)} + \sum_j W_{i,j}^{g}h_j^{(t-1)})$$  \hspace{1cm} (3)

The output gate of LSTM cells can also be closed (using sigmoid cells as gates):

$$q_i^{(t)} = \sigma(b_q^{i} + \sum_j U_{i,j}^{q}x_j^{(t)} + \sum_j W_{i,j}^{q}h_j^{(t-1)})$$  \hspace{1cm} (4)

Among them are bias, input weight and circular weight of forgetting gate. Among these variants, cell status can be selected as additional input (and its weight) into the three gates of the first unit, as shown in Figure 3, which will require three additional parameters[8].

From the above LSTM information processing process, we can clearly see that it has a good memory of the previous information, and can effectively solve the problem of the disappearance of the traditional
RNN gradient. Since the new cell state can be regarded as some form of accumulation of the previous cell state, the derivative of the new cell state is also the accumulation mode, which solves the problem of the disappearance of the traditional RNN gradient. Because the final output of this paper is a numerical value, a full-connection layer is added after the LSTM hidden layer, and then the final output layer. The activation function of this layer is selected as a linear function.

4. Establishment of Forecast Model of Foreign Exchange Data Based on Deep Learning

4.1 Data Set Source and Preprocessing
This paper chooses EUR/USD as the data of financial foreign exchange. The data used in this paper are from June 1993 to March 2018. The total data sample is 6400. Each sample contains price data: open, high, low, Close, short-term moving average (MA3), medium-term moving average (MA20), long-term moving average (MA73), exponential smoothing with moving average (MACD), Boll, Bias, relative strength index (RSI) and other data sets of 18 dimensions. In this paper, we use the 18-dimensional data set of T-Time to predict the closing price of t+1 time. The data set is divided into training set and test set. The training set contains 80% of the data set and the test set contains 20% of the data set, which are used for model training and model testing respectively.

In order to eliminate the influence of the different dimensions of the 18 dimensions in this paper, as well as the influence of the variance and numerical value of each variable itself, the data set was standardized before the training model. The data set basically conformed to the mean value of 0, and the standard deviation was 1 standard normal distribution.

4.2 Evaluation Index of Prediction Model
Evaluation Index RMSE and MAE of Forecasting Model.

5. Summary and analysis of experimental results
Manual adjustment of hyperparameters requires understanding what hyperparameters do and how network models can achieve good generalization. The main parameters adjusted during model training include batch size, cell_num, layer_num, epoch, learning_rate, time_step and dropout. Only by finding the appropriate super-parameters, can the model have better performance. Prediction performance.

Firstly, LSTM neural networks with 1 to 4 hidden layers are tested, and the experimental results are shown in Fig. 3. The number of neurons in each hidden layer of these models is 20, and the activation function is ReLu function. The training algorithm uses Adam algorithm, and the initialization of parameters is the same: the initial learning rate is set to 0.001, the batch size of each updating parameter is 200, 100, 60, 40 samples, and the training iterations of each model are 1000 times. The mean of the model tests is the final result.

Figure 3. Indicators for evaluating different batches of samples under four hidden layers.
The results of Figure 3 show that the best effect of the model is batch_size=200, layer=2; batch_size=100, layer=2; batch_size=60, layer=2; batch_size=40, layer=2.

Choose batch_size correctly to find the best balance between memory efficiency and memory capacity. Batch_size is a training sample of gradient descent. Each batch of samples will calculate a gradient descent to optimize the objective function. It can determine the direction of gradient descent. When batch_size is properly increased, the number of training times (epoch) of the model will be reduced, the time consumption will be reduced, the accuracy of gradient descent direction will be improved, and the amplitude of training vibration will also be reduced. From the results of Figure 3, when the model parameter is batch_size=60, RMSE is the smallest and the best parameter selection.

Hidden layer selection plays an important role in model prediction. If the number of layers is too small, the model can not extract data features very well, and its effect will be poor. If the number of layers is too large, the model will perform well in the training set, but poorly in the test set. From the results of Figure 3, we can see that RMSE is the smallest when the model parameter is layer=2, which is the best parameter choice.

Figure 4 shows the results of RMSE and MAE of LSTM network model under different number of neurons. As shown in Figure 4, when the number of neurons is 10, RMSE and MAE are the smallest and the prediction model is the best. The selection of the number of hidden layer neurons plays an important role in the prediction effect of the model. If the number of neurons is too small, the model will not be able to extract data features well, and its effect will be poor. If the number of neurons is too large, the model will perform well in the training set, but it will perform poorly in the test set.

From the results of Figure 5, we can conclude that the best effects of the model are: LR = 0.01, layer = 2; LR = 0.001, layer = 2; LR = 0.0001, layer = 2.

Comparing the above three cases, we can find that when LR = 0.001 and layer = 2, the RMSE value of evaluation index is the smallest. The learning rate directly affects the convergence speed of the model to the local minimum. If the learning rate is too low, the network may fall into local optimum, but if it exceeds the extreme value, the loss will not decrease any more and will oscillate back and forth in the local area. Therefore, choosing a suitable learning rate will train the model in a shorter time. In this experiment, when the learning rate is 0.001, the model has the best prediction effect and is the best parameter value.

The time_step of the model reflects how long the model can remember at most. The step size is smaller and the generalization ability is better. As for the influence of time step on the prediction of model effect, this paper compares the time step of LSTM neural network and RNN neural network from 10 to 60. Table 1 Comparisons of evaluation indices of model test sets with different time steps.

| Time_step | LSTM | RNN | LSTM | RNN |
|-----------|------|-----|------|-----|
| 10        | 84.44| 65.61| 92.75| 73.84|
| 20        | 75.17| 57.54| 89.91| 70.28|
| 40        | 82.15| 63.81| 92.70| 73.21|
| 60        | 86.08| 66.04| 87.71| 68.46|

Table 1 shows that RMSE and MAE on the test set are also the smallest when the time step is 20.
This shows that increasing the time step properly will make the horizontal depth of the model longer, and the longer the memory of the model, the more helpful to improve the prediction accuracy of the model. Compared with RNN model, LSTM model is better than RNN model in the same time step.

| Hidden Layer Number | LSTM | RNN |
|---------------------|------|-----|
|                     | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE | RMSE | MAE |
| LR=0.0001           |      |     |      |     |      |     |      |     |      |     |      |     |
| 1                   | 86.95| 65.39| 81.78| 62.48| 83.21| 63.49| 108.57| 154.28| 89.91| 70.28| 92.08| 75.39|
| 2                   | 81.69| 60.42| 79.28| 59.87| 81.35| 62.25| 405.38| 320.21| 81.72| 62.81| 84.21| 64.71|
| 3                   | 99.80| 78.21| 91.28| 70.29| 86.61| 64.67| 541.24| 414.12| 102.02| 81.52| 106.87| 83.13|
| 4                   | 124.12| 97.70| 95.82| 84.88| 91.52| 70.59| 713.27| 587.04| 128.78| 101.17| 110.48| 85.71|

Comparing the results of Table 2, both models have the smallest evaluation index RMSE and MAE when the learning rate LR is 0.001 and the number of hidden layers is 2. The results show that the prediction effect of LSTM model is better than that of RNN model under the same number of hidden layers and different learning rates, and the prediction effect of LSTM model is better than that of RNN model under the different number of hidden layers and the same learning rate.

The experimental results in Tables 2 and 3 above show that LSTM model is more accurate than RNN model in predicting the exchange rate price, so LSTM neural network model with memory ability is more suitable to be applied to the time series data with connection before and after.

6. Conclusion

In this paper, through manual parameter adjustment and many experiments, by comparing the size of the two evaluation indicators, suitable Super-parameters are found for RNN and LSTM models, so that the model has good performance. Using RNN and LSTM network model to forecast exchange rate price, the experimental results show that LSTM prediction effect is better than RNN, which proves that the unique LSTM network model can better learn the past exchange rate price data and find out the relationship between time series. It can also use the function of selective memory to further excavate the inherent law of exchange rate price, so as to have better forecasting effect.

Acknowledgments

Here I would like to express my heartfelt thanks to my master's supervisor, Mr. Yaxin. Qu. Thank him for his guidance in my thesis and for his help and teaching in the past three years.

Reference

[1] Aonan Liu. (2018) The Development of Quantitative Transaction. J. International Perspective.
[2] Kuremoto T, Kimura S, Kobayashi K, et al.(2014) Time series forecasting using a deep belief network with restricted Boltzmann machines. J. Neurocomputing, 137(15): 47-56.
[3] Dixon M F, Klabjan D, Bang J H.(2016) Classification-Based Financial Markets Prediction Using Deep Neural Networks. J. Social Science Electronic Publishing, 3: 28-37.
[4] Ruiqi Sun.(2015) Research on Prediction Model of US Stock Index Price Trend Based on LSTM Neural Network. D.Capital University of Economics and Trade.
[5] Kaimin Bai.(2017) Application of Neural Network and Deep Learning in Quantitative Investment. D. Shandong University.
[6] Jun Ren.(2018) Stock Index Prediction Based on Regularized LSTM Model. N. Computer Application and Software. 4:35(4).
[7] Kaile Dang.(2016) Research on Prediction Algorithms of Social Network Information Dissemination Based on Neural Network. D.Nanjing University of Posts and Telecommunications.
[8] Shenjian Zhao.(2017) Deep Learning. People's Posts and Telecommunications Publishing House, Beijing. pp. 249-250.
[9] Yongwen Bao.(2011) Statistical Analysis of the Income Gap between Urban and Rural Residents in Gansu Province J. Lanzhou University.