TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 08/11 pp77–85
DOI: 10.26599/TST.2018.9010007
Volume 24, Number 1, February 2019

LSTM Based Reserve Prediction for Bank Outlets

Yu Liu, Shuting Dong, Mingming Lu, and Jianxin Wang*

Abstract: Reserve allocation is a significant problem faced by commercial banking businesses every day. To satisfy the cash requirement of customers and abate the vault cash pressure, commercial banks need to appropriately allocate reserves for each bank outlet. Excessive reserve would impact the revenue of bank outlets. Low reserves cannot guarantee the successful operation of bank outlets. Considering the reserve requirement is effected by the past cash balance, we deal the reserve allocation problem as a time series prediction problem, and the Long Short Time Memory (LSTM) network is adapted to solve it. In addition, the proposed LSTM prediction model regards date property, which can affect the cash balance, as a primary factor. The experiment results show that our method outperforms some existing traditional methods.

Key words: reserve prediction; time series prediction; Long Short Time Memory (LSTM) network; date property

1 Introduction

Strong business competition exists between commercial banks. To improve their profits, commercial banks should increase their cash liquidity. Therefore, commercial banks must control their reserve limit. In general, the reserve allocation depends on the previous daily cash balance of bank outlets. For instance, corporate banking business transactions occur intensively, which would rapidly deplete the later reserve requirements. Hence, the series of daily reserve requirements would be considered as a time series. The Recurrent Neural Networks (RNNs) adeptly address the time series prediction problem. However, RNNs cannot handle long term dependencies. To avoid this weakness, Long Short Time Memory (LSTM) network was proposed in Ref. [1], which is capable of learning long term dependencies. We adapted the LSTM network to predict the reserve requirements.

We observed the different reserve tendencies between workdays and holidays. Thus, we considered that the date property acts as a significant factor in reserve prediction. The date would be classified into four categories where the workdays are labeled as 0, the Saturdays are labeled as 1, the Sundays are labeled as 2, the holidays, which can cover any other date property, are labeled as 3. For instance, the date would be labeled as 3 if it is both a Saturday and holiday. In our prediction model, the date property is the primary factor. The prediction model aims to mine the relation between daily reserve requirements and the date property.

In this paper, we predicted the reserves of bank outlets based on the LSTM with date property. The prediction model adopts N days of daily cash balance of bank outlets to predict the next daily reserve requirement. Our prediction system adopts a real data set from the bank and is deployed on the test environment. To evaluate the validity of our method, we used four standards, namely, Mean Absolute Differences (MAD), Root Mean Squarer Error (RMSE), Mean Absolute Percent Error (MAPE), and the accuracy of classification. We compared our prediction method with two traditional methods, which are the Auto-Regressive Integrated Moving Average (ARIMA) and the mean-growth methods. The core thought of the mean-growth method involves the growth rate of cash balance and the average cash balance of

• Yu Liu, Shuting Dong, Mingming Lu, and Jianxin Wang are with the School of Information Science and Engineering, Central South University, Changsha 410083, China. E-mail: jxwang@mail.csu.edu.cn.

∗ To whom correspondence should be addressed.

Manuscript received: 2017-07-17; accepted: 2017-08-07

the corresponding business outlets. In the mean-growth method, the prediction value equals to the growth rate multiplied by the average value.

Our main contributions include the following.

(1) To the best of our knowledge, this study is the first work that applies the LSTM to reserve requirement prediction of bank outlets.

(2) The date property is regarded as a primary influencing factor of the prediction model. The dates are classified into workday, Saturday, Sunday, and holiday; holidays can cover any other date property.

(3) The real data from bank 453 outlets are applied to a simulation experiment. The experimental results show the high accuracy and strong scalability of our prediction model.

The remaining of the paper is organized as follows. Section 2 lists the related work on reserve management and time series prediction. Section 3 discusses the details of reserve prediction for bank outlets based on LSTM and its implementation. Section 4 presents the experimental settings and results. Section 5 summarizes the paper and discusses the future work.

2 Related Work

2.1 Reserve management

Commercial banks possess a certain amount of excess reserve every day to meet the requirements of the deposit or withdrawal for daily cash in business outlets[2]. Excessive reserves lead to waste of funds and impact the financial liquidity. On the contrary, less reserves are harmful to the reputation of banks and cause significant unknown risks[3–5]. The objective of reserve requirements could be divided into three categories, prudent decision, monetary control, and liquidity management[6]. Regarding the bank liquidity requirements, several scholars joined the research. A model was proposed that is consistent with the credit and liquidity friction in the interbank market[7]. The capital ratios and the cash ratios were coordinated as reasonable measures in Ref. [8]. The authors in Ref. [9] analyzed the influence of minimal capital and reserve requirements on solvency and liquidity shortages.

2.2 Time series prediction

Given that the reserve data usually depend on previous data and contain considerable noise, the reserve data are essentially time series data. Therefore, the reserve prediction problem is a time series prediction problem. In recent years, a multitude of studies have focused on different time series problems. A hybridization of Artificial Neural Network (ANN) and ARIMA models for time series prediction was proposed in Ref. [10], and the experimental results proved that the hybrid model is superior to traditional hybrid models. RNNs were used to handle the time series problems[11, 12]. The author in Ref. [13] presented a competitive cooperative coevolution method for training recurrent neural networks for chaotic time series prediction. The time series prediction problem was disposed by the model of Auto-Regressive Moving Average (ARMA) with minimal assumption[14]. The authors in Ref. [15] applied the Bayesian evidence framework to the Least Squares Support Vector Machine (LS-SVM) regression in order to infer nonlinear models for predicting a financial time series and the related volatility. Several of the above studies rely on RNNs. Given their recurrent character, RNNs conveniently solve the time series prediction problem. However, as the RNNs perform poorly at handling long term dependencies, the LSTM was introduced in Ref. [1]. By default, the LSTM remembers information for long periods of time. With its default behavior, the LSTM serves as an excellent model for solving the time series prediction problem.

3 LSTM Based Reserve Prediction for Bank Outlets

We propose the novel idea of using the LSTM model for reserve prediction of bank outlets. We also regarded daily reserve data as the time series data. The reserve requirement of bank outlets is affected by the previous by past daily cash balance series. The tendency of daily cash balance series can be discovered by LSTM with multiple hidden layers. In this section, we present our LSTM based reserve prediction for bank outlets as Fig. 1.

As shown in Fig. 1, each transaction record would be saved in the database. Through data processing, a daily cash balance would be shaped as a feature vector and a

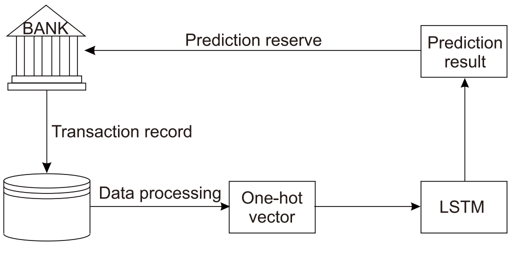

Fig. 1 Work flow of prediction module.

one-hot vector. The feature vector is used as the input of the LSTM prediction model, whereas the one-hot vector is used to compare with the prediction result and calculate the loss of prediction model. Given the date property, which is the primary factor of the prediction model, each bank outlet would build a prediction model independently. After training, the LSTM prediction model would yield an output for the prediction interval, and a random value in the interval is regarded as the prediction result for a specified date. Finally, the prediction result is introduced to the bank.

3.1 Data preprocessing

3.1.1 Data tagging

Achieving accurate predictions presents extreme difficulty. On the other hand, the LSTM is an expert in classification issues. We considered the prediction of intervals to be more significant than accurate predictions. Therefore, in the model building processing, we reduced the prediction problem to a classification problem. The 99% confidence interval of the training data would be computed according to its average value and its Standard Deviation (SD). Then, the confidence interval would be classified into N_CLASS intervals. The daily cash balance would be tagged at the category label.

3.1.2 Normalization

To train the model parameters more sufficiently and converge them to an appropriate value, we randomly divided the training data set into N batches and to only contain a partial training set. Then, we performed normalization for each batch.

Normalization refers to normalizing of data dimensions to transform them to approximately the same scale. The reserve prediction system for bank outlets uses the zero-score, which divides each dimension by the average value. The zero-score transform function is shown in Eq. (1), where x refers to the average value of the sample, and σ denotes the SD of the sample.

$$x^{\star} = \frac{x-\mu}{\sigma} \tag{1}$$

Therefore, after calculating every feature dimensions average value and SD for all the training sets, normalization of the entire data is completed.

3.2 Model construction

Figure 2 shows our LSTM based reserve prediction for bank outlets. Let x and h represent the input and output, respectively. The key to LSTM is the cell state, which is kind of a conveyor belt. The cell state runs straight down the entire chain, with only several minor linear interactions. Information easily flows unchanged along the chain. The LSTM used three gates to remove or add information to the cell state.

LSTM can be unfolded into a full network as shown in Fig. 2. Suppose that we consider three states at time $t-1$, t, and $t+1$. x_t and h_t $(t-1 \leqslant t \leqslant t+1)$ denote the input and output, respectively. E denotes the output value of each gate. The first step in our LSTM prediction model to determine the information would be removed from the cell state. The decision is made by a sigmoid layer called the "forget gate layer". Formally, the calculation is shown in Eq. (2).

$$E_{f_t} = \delta(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

where δ indicates the sigmod function, h_{t-1} represents the output of the previous cell at the same sequence step, x_t stands for the input at state t, and W_f and b_f denote weight and bias values of the forget gate, respectively. This step yields an output, which is a number between 0 and 1 for each number in the cell state C_{t-1} according to h_{t-1} and x_t.

The second step decides which new information would be stored in the cell state. This step includes two parts. First, a sigmoid layer called the "input gate

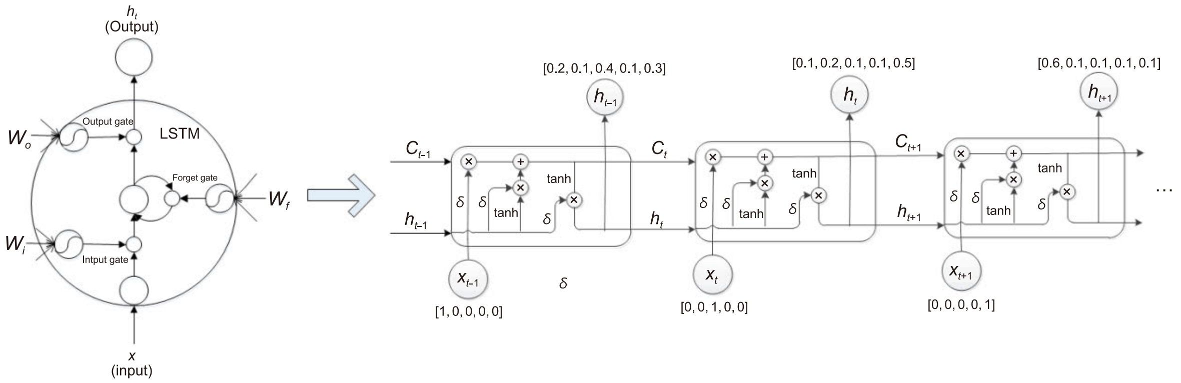

Fig. 2 LSTM based reserve prediction for bank outlets.

layer" decides which values would be updated. Then, a tanh layer creates a vector of new candidate values, $\widetilde{C}_t$, that could be added to the state. The computational processes of the step are shown in Eqs. (3) and (4).

$$E_{i_t} = \delta(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$\widetilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{4}$$

where W_i and b_i denote weight and bias values of the input gate, respectively. W_C and b_C denote intermediate weight and bias of the input gate for creating the new candidate values. The previous cell status C_{t-1} is updated into the new cell state C_t. The computational formula is shown in Eq. (5).

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C}_t \tag{5}$$

Finally, LSTM would yield an output based on the cell state. First, a sigmoid layer decides the parts of the cell state that would be run as output. Then, the cell state is processed by tanh and sigmoid layers to give the final output h_t. The computational formulas are shown in Eqs. (6) and (7).

$$E_{o_t} = \delta(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{7}$$

Each time state of LSTM includes *N_HIDDEN* hidden layers and each layer features Q neurons. We assume that all layers adopt the same number of neurons. Inside a state, neurons in the j-th hidden layer connect to the neurons in the $(j+1)$-th hidden layer. When two states are consecutive, the neurons in the j-layer of time state t link to the neurons in the j-layer of time state $t+1$.

Finally, the output layer would output a probability vector $\boldsymbol{O}_{\text{out}} = [o_0, o_1, ..., o_{N\_CLASS-1}]$. o_m represents the probability of prediction reserve belonging to the m-th interval.

3.2.1 Weight initialization

Before training the LSTM prediction model, we must initialize its parameters. On the one hand, the weights should be very close to zero. On the other hand, the same gradients exist during backpropagation if every neuron in the network computes the same output. Therefore, we selected small random numbers. As the weights of the neurons to small numbers and to break the symmetry. Initially, all neurons are random and unique. Thus, they will be used to compute the distinct updates and integrate them as diverse parts of the full network. The weight value of each neural network node is initialized by following formula:

$$W_{\text{each}} = 0.01 \cdot r \cdot \text{sqrt}(2/n) \tag{8}$$

where r is a random value and n is the number of nodes. With this formulation, the weight vector of each neuron is initialized as a random vector sampled from a multi-dimensional Gaussian. Thus, the neurons follow a random direction in the input space.

3.2.2 Regularization

To control the capacity of neural networks and to prevent overrating, we should use regularization. In our prediction system, we selected dropout as the regularization method. During training, the dropout was implemented by only keeping an active neuron active with a probability P. We set 0.7 as the default dropout value in the training process.

3.2.3 Activation function

In the computational model, we need an activation function, which is a non-linear learning function. In general, every activation function takes a single number and performs a certain fixed mathematical operation on it. We selected the tanh non-linearity function as the activation function. This function squashes a real-valued number to the range $[-1, 1]$. The output of tanh non-linearity function is zero-centered. Equation (9) calculates the tanh non-linearity function.

$$\tanh(x) = 2\sigma(2x) - 1 \tag{9}$$

We managed to finish the model construction process.

We selected the Stochastic Gradient Descent algorithm (SGD) to learn the weights of the LSTM network. We implemented the gradient checking to verify the correctness.

4 Experiment

4.1 Data set

The data set consists of real daily deposit and daily withdrawal records of 453 bank outlets from December 28, 2012 to June 9, 2015. To identify the effect of the date property on the daily cash balance of bank outlets, we set a date property label. We marked the workdays as 0, the Saturdays as 1, the Sundays as 2, and the holidays as 3.

4.2 Data sample setting

First, we divided the training set and test set at a $3:1$ ratio. The training sets were used to train the prediction model, whereas the test set was used to verify the prediction model. The ratio could be changed in the experiment.

Second, the sample features N days of data, the default value of N is 10, and the format of sample is shown as $[x_0, x_1, \ldots, x_{n-1}]$.

Every sample contains D dimensions, the default value of D is 5, and the format is as [*month*, *date*, *daily deposit*, *daily withdrawal*, *date label*].

We sorted the sample data according to the value of days and feature dimensions. Finally, the sample is shaped as $\boldsymbol{X}_{N\times D}$. For example, when we set $N = 10$ and $D = 5$, the sample shape is shown as Eq. (10).

$$\boldsymbol{X} = \begin{pmatrix} 4 & 25 & 278 & 302 & 0 \\ 4 & 26 & 216 & 196 & 0 \\ 4 & 27 & 430 & 330 & 1 \\ 4 & 28 & 444 & 498 & 2 \\ 4 & 29 & 170 & 193 & 3 \\ 4 & 30 & 233 & 193 & 3 \\ 5 & 1 & 248 & 209 & 3 \\ 5 & 2 & 293 & 207 & 0 \\ 5 & 3 & 412 & 652 & 0 \\ 5 & 4 & 192 & 154 & 1 \end{pmatrix} \tag{10}$$

4.3 Steps for data tagging

(1) The daily cash balance of each sample should be selected. As shown in Fig. 3, we observed the normal distribution of the daily cash balance data.

(2) We supposed that the data to be predicted satisfy the normal distribution, and confirm the boundaries of the range of values. We calculated the mean value (MEAN) and SD of the training set. Then, the 99% confidence interval ([MEAN − 2.576×SD, MEAN + 2.576 × SD]) is used as the distribution range of all tagged data. This range would be classified into *N_CLASS* intervals of which the left margin is closed, and the right margin is open.

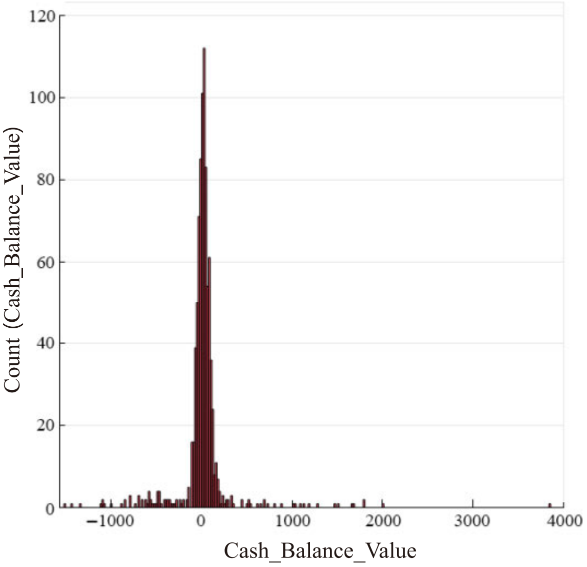

Fig. 3 Distribution of daily cash balance.

(3) After dividing the daily cash balance into the *N_CLASS* intervals, the intervals would be assigned from 0 to *N_CLASS* − 1. The tagged data would be assigned the number according to the located interval. Additionally, the data which are not in the *N_CLASS* intervals would be partitioned into the nearest interval.

(4) Finally, the classified tagged data would be transformed to a one-hot vector. The one-hot vector is used to compare with the prediction result and calculate the cross-entropy value.

4.4 Model construction

The reserve prediction system of bank outlets based on LSTM would construct an LSTM network in which the hidden layer consists of *N_HIDDEN* LSTM cells. The input data would be unfolded on the time dimension, mapped into the hidden layer, and used as the output of the upper hidden layer. We would obtain the predicted intervals after a series of matrix transformation.

4.5 Experimental evaluation

To evaluate the proposed LSTM prediction model, we deployed it on the test environment. The LSTM prediction model was trained on a server equipped with Xeon 2.4 GHz 20-core CPUs, 128 GB memory, and Ubuntu 14.04TSL OS. LSTM was configured to run on the CPU mode. During the training process, 75% data are used for training, and 25% data are used for testing. Therefore, we predicted the reserve requirement from January 1, 2015 to June 9, 2015. A total of 4 to 6 min are required for the outlets to construct the prediction model and yield as output the prediction results.

By default, the LSTM prediction model maintains 10 states. As presented in Fig. 4, five hidden layers exist. Each hidden layer contains 50 neurons. The output vector consists of the probabilities of a prediction reserve that would fall into each interval. For each bank outlet, we split the training data into multiple samples

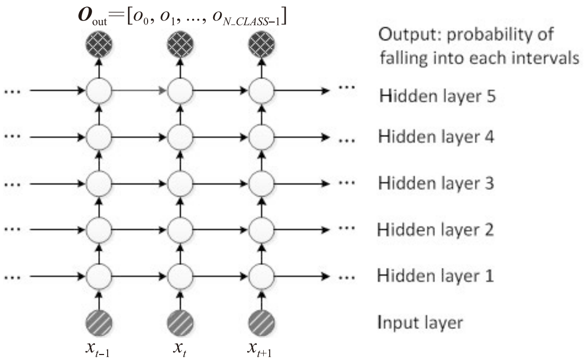
$\boldsymbol{O}_{\text{out}}=[o_0, o_1, \ldots, o_{N\_CLASS-1}]$
Output: probability of falling into each intervals
Hidden layer 5
Hidden layer 4
Hidden layer 3
Hidden layer 2
Hidden layer 1
Input layer
x_{t-1}
x_t
x_{t+1}

Fig. 4 Illustration of LSTM prediction model.

that share the same configuration. Table 1 lists the default settings of the experiment.

During the testing process, the output of the prediction model is exactly one interval for one date. If a prediction interval contains the real reserve value, we considered the prediction interval as correct. The computational formula of the accuracy is shown as Eq. (11).

$$\text{Accuracy} = \frac{P(D)}{|D|} \tag{11}$$

where D denotes the length of prediction period and $P(D)$ returns the number of correct prediction. In this experiment, we set *N_CLASS* to 5. Based on this, the confidence interval is divided into five equal intervals. The predicted value of the daily reserve requirement is classified into one of the five intervals. We record the number of correct predictions that the predicted value belongs to the same interval as the actual value. We test the reserve requirement from January 1, 2015 to June 9, 2015, and the prediction accuracy is 0.850 697. As shown in Fig. 5, most of the actual values are in the prediction intervals and a little extremely fluctuating values were also observed. In general, the extremely fluctuated values originated from the corporate businesses. However, bank outlets follow a reservation protocol to cope with the corporate business. Generally, the extreme fluctuated value is caused by the corporate business. However, there is reservation protocol for bank outlets to cope with the corporate business.

The prediction model is trained iteratively. In each iteration, the model parameters would be updated by using all the training samples. Each iteration is called an epoch in our experiment, and hundreds of epochs may be needed before the model converges. For comparison, we also applied the data to the ARIMA approach, a widely used time series prediction algorithm. We also used the mean-growth method which is commonly adopted in the reserve prediction of bank outlets. Finally, the comparison results are shown on the web page based on EChart package and Bokeh package.

Table 1 Experiment setting.

Parameter	Default value
Number of hidden layers in LSTM	5
Number of states in LSTM	10
Number of neurons in LSTM	50
Activation function	tanh
Loss function	Cross-entropy loss
Initial learning rate	0.001
Batch size in LSTM	200
Iteration times	100 000
Weight initialization function	Random small value

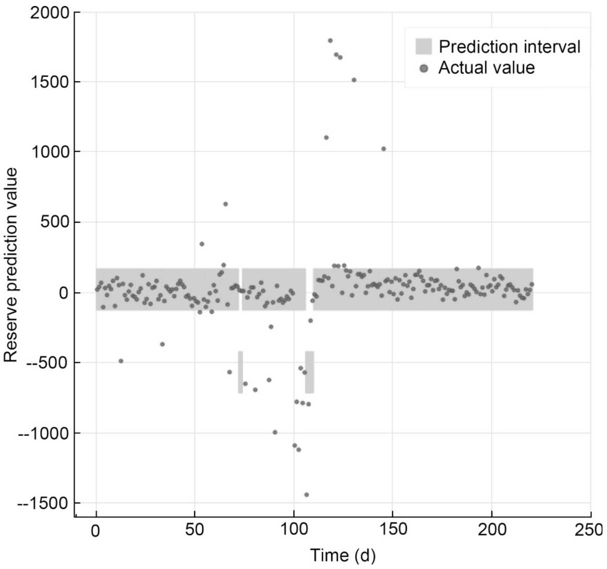

Fig. 5 Accuracy of prediction results.

4.6 Effect of interval number

During the learning process, the prediction values of daily reserve requirement are classified to a certain interval. We set *N_CLASS* as the number of prediction intervals. In training process, we usually classified the extreme values that are out of the confidence interval to the nearest boundary intervals. To find an appropriate *N_CLASS* value, we compared the accuracy of different *N_CLASS* values. Figure 6 shows the results. We can learn that the accuracy features a reduction trend with increasing interval numbers. In general, we set 5 as the default *N_CLASS* value.

4.7 Prediction value

Considering that the reserve requirement of bank outlets is an exact value, it is insufficient to take an interval as the final prediction. From the statistical data, we observed people with a random behavior of deposit requirements or withdrawal requirements, except the corporate business. Given the predicted interval, we generated a random value in the interval and considered it as the predicted value of the special date. The prediction value is more stable, whereas the *N_CLASS*

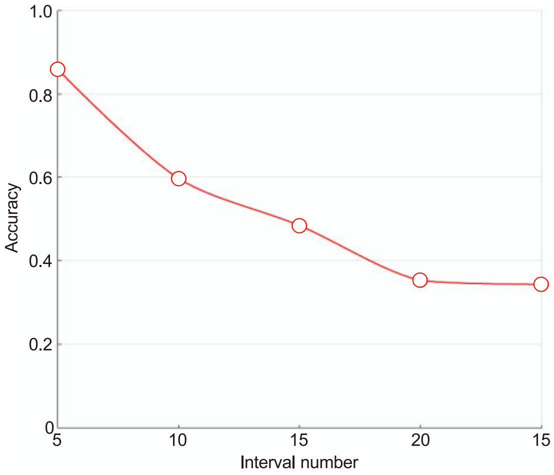

Fig. 6 Effect of interval number.

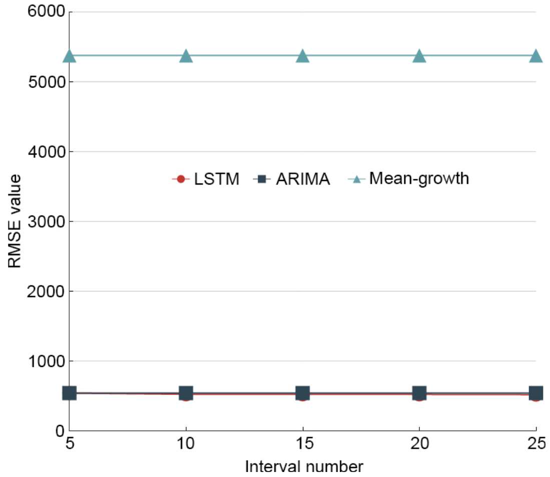

Fig. 8 Comparison of RMSE values.

is larger.

To measure the prediction effect, we used three classical standards, MAD, RMSE, and MAPE. After obtaining the final prediction value, we compared the values of the standards for LSTM prediction model to the standard values for ARIMA and the mean-growth methods.

From Fig. 7, the LSTM prediction model features the minimum MAD value compared with the ARIMA and the mean-growth methods. In addition, the MAD value of the LSTM prediction model decreases with increasing interval number. However, owing to a number of extremely fluctuating values, the three methods yielded large MAD values.

As shown in Fig. 8, the RMSE value of the LSTM prediction model is almost equal to that of the ARIMA prediction model. However, the RMSE value of the mean-growth method is 4 times as much as of that of the LSTM prediction model. On the other hand, Fig. 8 shows that the interval number exerts little effect on the RMSE value for the LSTM prediction model.

As shown in Fig. 9, the LSTM prediction model yields a minimum MAPE value compared with the other two methods. The MAPE value of the mean-growth model is 7 times, whereas that of the ARIMA method is 3 times as much as the LSTM prediction model. In addition, the MAPE value of the LSTM prediction model decreases with growing interval number. When the interval number equals 25, the MAPE value of the LSTM prediction model is almost equal to 0.1.

As shown in Fig. 10, from January 25th to February 25th, the mean-growth method exhibited several exceeding prediction values, whereas both the prediction results of LSTM and ARIMA are close to the real values. Therefore, the prediction results of the mean-growth method are too extreme to present the real reserve values.

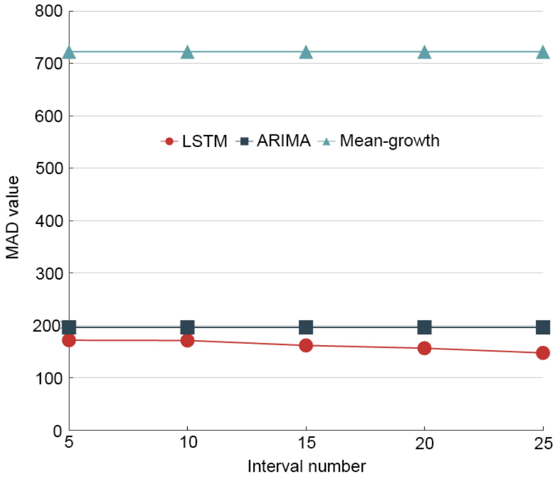

Fig. 7 Comparison of MAD values.

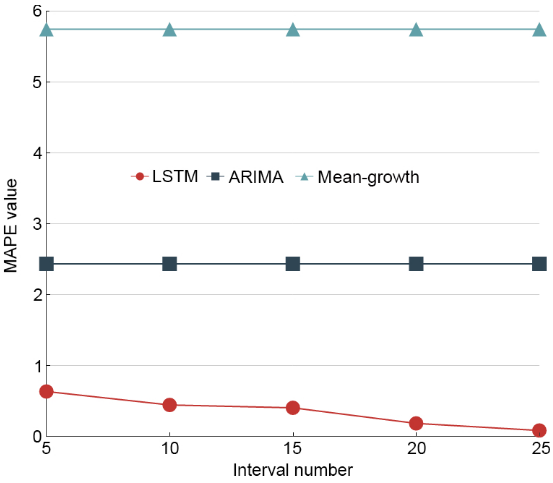

Fig. 9 Comparison of MAPE value.

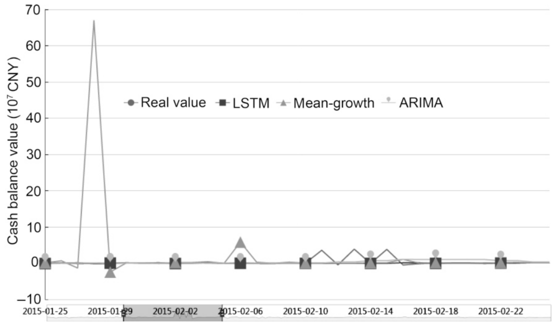

Fig. 10 Comparison of predictions.

As depicted in Fig. 11, from March 1st to March 31th, although some fluctuations were hardly predicted, the prediction results of the LSTM are the closest to the real value than those of ARIMA and mean-growth. In addition, ARIMA features a stable tendency and thus hardly reflects the trend of real values. Besides, the prediction results of mean-growth appear more randomly to the real reserve values and difficult to represent the real reserves.

Overall, the LSTM prediction method shows a more excellent performance than the ARIMA and the mean-growth methods.

5 Conclusion and Future Work

In this paper, we presented the LSTM based reserve prediction for bank outlets and has successfully deployed the prediction system to the test environment. In our prediction model, the daily cash balance of bank outlets is regarded as their daily reserve requirement. We extracted the cash balance regulations of the bank outlets and predicted the later daily reserve requirement by constructing the LSTM network with five hidden layers. Our results on real data set showed that the LSTM prediction approach significantly outperformed the previous ARIMA prediction and the mean-growth prediction methods. Although our prediction model primarily considers the date property, our prediction model is extensible.

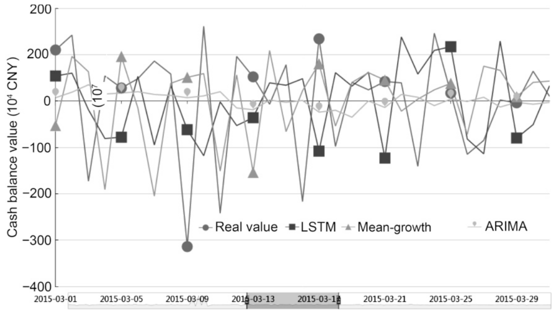

Fig. 11 Comparison of predictions from March 1st to March 31th.

In the future, our main focus is on the other factors such as the address of bank outlets and the weather of the bank outlets location. We will aggregate these factors and construct a general prediction model to the resolve reserve prediction problem for bank outlets. We also will consider how to use new techniques to improve the predicting performances, such as parameterized algorithm[16, 17] and matrix completion[18, 19].

References

[1] S. Box and J. Schmidhuber, Long short-term memory, *Neural Computation,* vol. 9, no. 8, pp. 1735–1780, 1997.

[2] A. K. Kashyap, R. Rajan, and J. C. Stein, Banks as liquidity providers: An explanation for the coexistence of lending and deposit-takin, *The Journal of Finance,* vol. 57, no. 1, pp. 33–73, 2002.

[3] D. Hancock and J. A. Wilcox, Intraday management of bank reserves: The effects of caps and fees on daylight overdrafts, *Journal of Money, Credit and Banking,* vol. 28, no. 4, pp. 870–908, 1996.

[4] D. Heller and Y. Lengwiler, Payment obligations, reserve requirements, and the demand for central bank balances, *Journal of Monetary Economics,* vol. 50, no. 2, pp. 419–432, 2003.

[5] L. Wang, Y. Chai, and Y. Liu, Analysis of specialized production of transaction services based on essential services quantity, *Tsinghua Science and Technology,* vol. 22, no. 5, pp. 529–538, 2017.

[6] S. Gray, *Central Bank Balances and Reserve Requirements*. Washington, DC, USA: International Monetary Fund, 2011.

[7] A. Ashcraft, J. McAndrews, and D. Skeie, Precautionary reserves and the interbank market, *Journal of Money, Credit and Banking,* vol. 43, no. 370, pp. 311–348, 2011.

[8] C. W. Calomiris, Getting the right mix of capital and cash requirements in prudential bank regulation, *Journal of Applied Corporate Finance,* vol. 21, no. 1, pp. 33–41, 2012.

[9] A. Krause and S. Giansante, Liquidity and solvency shocks in a network model of systemic risk: The impact of minimum capital and reserve requirements, in *Proc. 25th Australasian Finance and Banking Conf.*, Sydney, Australia, 2012, p. 381.

[10] 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.

[11] X. Cai, N. Venayagamoorthy, and G. K. Venayagamoorthy, Time series prediction with recurrent neural networks trained by a hybrid PSO–EA algorithm, *Neurocomputing,* vol. 70, nos. 13–15, pp. 2342–2353, 2007.

[12] C. L. Giles, S. Lawrence, and A. C. Tsoi, Noisy time

series prediction using recurrent neural networks and grammatical inference, *Machine Learning,* vol. 11, nos. 1&2, pp. 161–183, 2001.

[13] R. Chandra, Competition and collaboration in cooperative coevolution of Elman recurrent neural networks for time-series prediction, *IEEE Transactions on Neural Networks and Learning Systems,* vol. 26, no. 12, pp. 3123–3136, 2015.

[14] O. Anava, E. Ning, S. Mannor, and B. A. Pescosolido, Online learning for time series prediction, in *Proc. 26th Annual Conf. on Learning Theory*, 2013, pp. 172–184.

[15] T. Van Gestel, J. A. K. Suykens, and D. E. Baestaens, Financial time series prediction using least squares support vector machines within the evidence framework, *IEEE Transactions on Neural Networks,* vol. 12, no. 4, pp. 809–821, 2001.

[16] Y. Ke, Y. Cao, X. Ouyang, W. Li, and J. Wang, Unit interval vertex deletion: Fewer vertices are relevant, *Journal of Computer and System Sciences,* vol. 95, pp. 109–121, 2018.

[17] F. Shi, J. Chen, Q. Feng, and J. Wang, A parameterized algorithm for the Maximum Agreement Forest problem on multiple rooted multifurcating trees, *Journal of Computer and System Sciences,* vol. 97, pp. 28–44, 2018.

[18] H. Luo, M. Li, S. Wang, Q. Liu, Y. Li, and J. Wang, Computational drug repositioning using low-rank matrix approximation and randomized algorithms, *Bioinformatics,* vol. 34, no. 11, pp. 1904–1912, 2018.

[19] C. Lu, M. Yang, F. Luo, F.-X. Wu, M. Li, Y. Pan, Y. Li, and J. Wang, Prediction of lncRNA-disease associations based on inductive matrix completion, *Bioinformatics,* doi: 10.1093/bioinformatics/bty327.

Yu Liu is a master student in computer science at Central South University. She received the bachelor degree from Central South University, China, in 2014. Her main research interests include big data progressing.

Jianxin Wang received the BS and MS degrees from Central South University in 1992 and 1996, respectively, and received the PhD degree from Central South University in 2001. He is a vice dean and a professor in School of Information Science and Engineering at Central South University, China. His current research interests include algorithm analysis and optimization, parameterized algorithm, bioinformatics, and computer network. He has published more than 150 papers in various international journals and refereed conferences. He is a senior member of IEEE.

Shuting Dong is a master student in computer science at Central South University. Her main research interests include deep learning and big data analysis. She received the bachelor degree from Jiangxi University of Finance and Economics, China, in 2016.

Mingming Lu received the PhD degree in computer science from Florida Atlantic University, US, in 2008. He is currently an associate professor at the School of Information Science and Engineering, Central South University, China. His main research interests include deep learning and big data analysis.