Time Series Prediction based on Improved Deep Learning

Huang Sen

Abstract—**To mine the effective information contained in massive data and improve the accuracy of short-term prediction, an improved deep learning model is proposed for time series prediction based on long short-term memory (LSTM) network. The characteristics of data timing and nonlinearity are used by the proposed method. Firstly, the long short-term memory network is introduced briefly. Secondly, LSTM is improved with attention mechanism. Two dimensions of time and sequence features are considered to design attention mechanism. The time dependence of different features is modeled. The constraint function is added to optimize LSTM. Thirdly, the time series prediction process is given, which mainly contains model training and predicting flow. The experimental results show that the proposed method has higher prediction accuracy than the traditional LSTM prediction method.**

***Index Terms*—time series; neural network; deep learning; LSTM**

I. Introduction

With the rapid development of the industrial system, time series prediction is urgently required for industrial system planning, scheduling, and operation. It plays an important role in ensuring the industrial system safe and economic operation. As a result, time series prediction has attracted great attention of engineers and scholars. It has become an important direction in scientific research. Time series prediction can be divided into long-term, medium-term, and short-term prediction. Short-term prediction plays an important role in guiding and regulating the daily operation. Therefore, most research focuses on short-term time series forecasting.

For time series prediction, the historical data is usually analyzed to explore the internal relationship of time series [1-5]. Time series usually have the characteristics of temporality and nonlinearity [6,7]. Based on time series characteristics, the prediction models are generally divided into two categories. One is the time series analysis methods, such as regression analysis [8], exponential smoothing model [9], Kalman filter [10], multiple linear regression [11], Fourier expansion model [12] and autoregressive integral moving average (ARIMA) model [13], etc. The basic idea is to predict the future value according to the past and present value. The advantage is its consideration of the temporal relationship for data. The disadvantage is that it has limited prediction ability for nonlinear relationship data. The second prediction models are machine learning methods. For example, back propagation (BP) neural network is used in literature [14,15]. Literature [16] proposed a time series neural network model, which is used to estimate the state of charge in the battery. The proposed model reduces the prediction error. The experiment result indicates the good performance of the presented method. The proposed model can overcome the error caused by RLS algorithm. Literature [17] studied the recurrent neural networks for multivariate time series problem. In time series forecasting and other related tasks, it has been noted that the missing values and their missing patterns are usually related to target labels. A novel prediction model based on the improved gated recurrent unit is proposed. Experiments of time series classification tasks on real-world clinical datasets are performed to verify the effectiveness of the proposed method. Literature [18] applied the time series prediction for financial forecasting with support vector machine (SVM). The evidence collected in this article shows that the improved support vector machine is more accurate than the traditional support vector machine. However, the experimental cases used in each study are different. It is impossible to obtain the general conclusion. Traditional machine learning algorithms are lack of information expression for big data due to their low model complexity. At the same time, the traditional machine learning algorithm can not add the expression of time series dimension.

Deep learning is a method based on data representation, which has become a research hotspot for time series prediction. The purpose of deep learning is to establish a neural network to simulate analytical learning of human brain. The mechanism of the human brain is simulated to interpret data. And the internal laws hidden in data distribution is mined. Deep learning can effectively express the massive data information. Long short-term memory (LSTM) is one of the deep learning methods, which considers the temporal and nonlinear relationship of data. LSTM has been gradually applied for time series prediction, such as language model [19], machine translation [20], speech recognition [21], etc. Literature [22] improved the LSTM to forecast the stock market in India. The LSTM performance is related to the choice of model hyperparameters. These model hyperparameters should be carefully selected to obtain superior performance. A series of experiments were conducted to analyze the LSTM in literature [23]. The Wilcoxson signed-rank test was used to compare two z-normalization techniques. Z-normalization was used for each sample independently. The experimental results show that multiple LSTM blocks should be used in combination to obtain the superior prediction result. LSTM model can fully reflect the long-term historical process of time series [24, 25].

Manuscript received December 23, 2021; revised October 2, 2022.

Huang Sen is a lecturer of Henan Vocational College of Agriculture, Zhengzhou, Henan, China (email: 84056535@qq.com)

In recent studies, it has been found that sequence prediction with attention mechanism can effectively and quickly screen out critical information. Attention mechanism can further enhance the prediction ability of time series.

Aiming at the dynamic prediction of time series, a prediction model is designed based on long short-term memory networks with attention mechanism. The characteristics of time and sequence are considered to design attention mechanism. The results are verified by experiments. The experiment results fully illustrate the effectiveness of the proposed method for time series prediction.

II. Long short-term memory networks

LSTM is an improved recurrent neural network (RNN) model. Compared with the traditional neural network model, RNN can establish weight connection among layers. The neurons of each RNN cell can also establish weight connection. In the standard RNN structure, the neurons of the hidden layer have weight. The previous hidden layer affects the next hidden layer. Therefore, RNN is better than other kinds of neural network models for time series problems.

The core idea of LSTM is to add three gate control units to each RNN cell, as shown in Figure 1. To solve the gradient disappearance problem of RNN model for long time series, remembering or forgetting the information of key nodes is chosen.

The cell of each LSTM includes an updating gate unit G_u, forgetting gate unit G_f, outputting gate unit G_o, memory unit R and information transmission unit b. Supposing the certain time instance is t, the information c^{t-1} at t-1 time instance and the input information x^t at t time instance are multiplied by the weight W. The offset b is added. The control coefficient of the gate cell G is obtained by sigmoid activation function $f(x)$. Updating the alternative information $R^{t'}$ in the memory unit or forgetting the information R^{t-1} stored at time instance t-1 is controlled. The information R^t in the memory unit is updated. The information in the updated memory cell R^t is passed through the activation function tanh(x). The result is multiplied by the control coefficient of the output gate unit G_o. Then c^t is passed into the next LSTM cell.

The key information can be memorized and updated in the process.

$$R^{t'} = \tanh(W_R[c^{t-1}, x^t] + b_R) \tag{1}$$

$$G_u = \alpha(W_u[c^{t-1}, x^t] + b_u) \tag{2}$$

$$G_f = \alpha(W_f[c^{t-1}, x^t] + b_f) \tag{3}$$

$$G_o = \alpha(W_o[c^{t-1}, x^t] + b_o) \tag{4}$$

$$R^t = G_u R^{t'} + G_f R^{t-1} \tag{5}$$

$$c^t = G_o \tanh R^t \tag{6}$$

III. LSTM with Attention Mechanism

A. Attention Mechanism

Most of the attention mechanisms are based on deep loop network coding and decoding process for timing series. The learning model is divided into two modules. Firstly, the encoder composed of a single-layer or multi-layers is used according to the time relationship. The encoder learns the dependencies of the known sequence and the representation of the current state. The hidden state is obtained at the last time, named vector $\boldsymbol{D}$. $\boldsymbol{D}$ retains the dynamic information of the historical sequence and the current sequence. A decoder is composed of neural network units with similar structure. The coding vector $\boldsymbol{F}$ is converted into timing information with prediction length L. The input of each time instance i is a vector obtained by the mapping of vector $\boldsymbol{F}$ and the target value sequence (z_1, z_2, …, z_{i-1}). The output value is predicted at time instance i.

$$\boldsymbol{F} = f(x_1, x_2, \cdots, x_L) \tag{7}$$

$$z_i = g(\boldsymbol{F}, z_1, z_2, \cdots, z_{i-1}) \tag{8}$$

In traditional coding and decoding model, the vector $\boldsymbol{F}$ used at each time of decoding is fixed. Such structure can not integrate different information into the same model. Researchers have made a more in-depth exploration. The attention mechanism is introduced from image recognition into the sequence prediction. By combining the design of attention mechanism with the structure of coding and decoding, a method of sequential attention mechanism is proposed.

$$\boldsymbol{F}_i = f(x_1, x_2, \cdots, x_L, \boldsymbol{c}, c^{i-1}) \tag{9}$$

$$z_i = g(\boldsymbol{F}_i, z_1, z_2, \cdots, z_{i-1}) \tag{10}$$

where f () represents the process of combining the attention mechanism for the encoder. c^{i-1} is the hidden state of the decoder at time instance i-1. $\boldsymbol{c}$ is the set of hidden states of the encoder. Different from traditional coding and decoding model, the encoder obtains a dynamic vector F_i with different attention information at each time instance i. Thus, the decoding process can pay more attention to the important historical information for the current time prediction content.

B. LSTM with Attention Mechanism

After LSTM model, the dynamic state c^t and memory state R^t can be obtained. Important information should be extracted from the previous time series. The model needs to learn the key role of information at different times in long-term prediction.

Firstly, the design idea of codec model is considered. Attention degree to the hidden state at different times is not the same. An attention mechanism based on the historical state is constructed in the time dimension. The output vector $\boldsymbol{c}$ = [c^1, c^2, …, c^T] with hidden layer states can be obtained. The model further takes $\boldsymbol{c}$ as the output of the encoder. Each c^t in the vector $\boldsymbol{c}$ represents the state at time instance t, which is treated as the input of the attention mechanism. Then, the importance of the prediction state z_i is calculated through the following equations.

$$\beta_i^t = P^{\mathrm{T}} \tanh(Qc^{i-1} + Xc^t) \tag{11}$$

$$b_i = \mathrm{soft}\max(\beta_i^t) \tag{12}$$

where c^{i-1} is the hidden state of the decoder. c^t is the state of the encoder at time instance t. P, Q and X are decoder hidden state, encoder state and the parameter matrices of attention mechanism, respectively. β_i^t represents the influence degree of encoder state at time instance t to the output state at time instance i.

Finally, using the softmax function, β_i^t is normalized to obtain the weight factor b_i. Total weight factor w_i at time instance i is calculated as following.

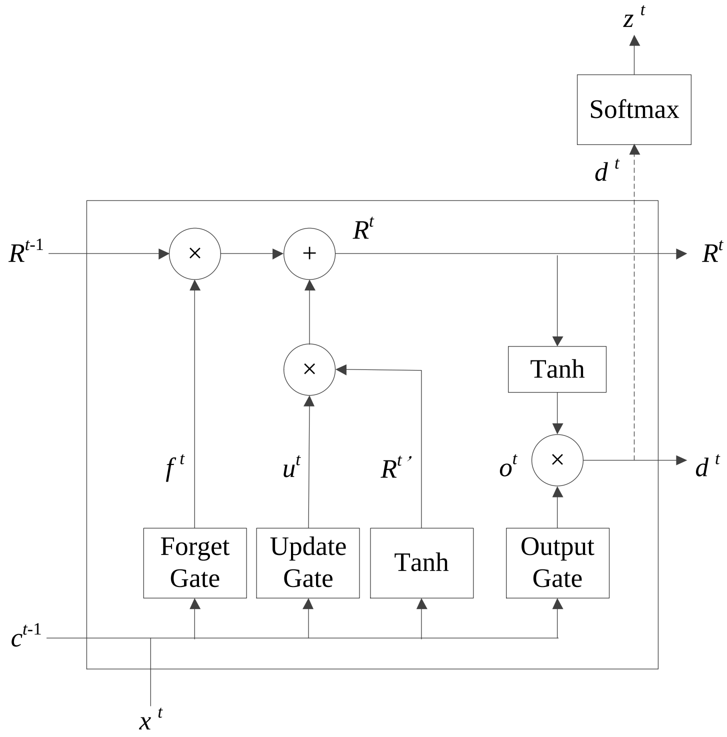


Fig.1 Long Short-Term Memory Block Diagram

$$w_i = \sum_{t=1}^{T} \beta_i^t bc^t \tag{13}$$

Thus, the predicted value z_i at time instance i can be gradually obtained.

$$z_i = LSTM(w_i, c^{i-1}, R^{i-1}) \tag{14}$$

where w_i is the total weight factor. c^{i-1} is the hidden state of the decoder at time instance i-1. R^{i-1} is the memory cell at time instance i-1.

C. Optimization Function Design

Time series are usually characterized by stable and gentle changes. Therefore, a linear evolutionary constraint process is proposed to optimize the function. The conditional distribution is used to meet the linear stationary constraint for time series coding.

$$\boldsymbol{h}_t \mid \boldsymbol{h}_{t-1} \sim N(\boldsymbol{M}\boldsymbol{h}_{t-1}, \Sigma) \tag{15}$$

where M is the state transfer matrix. The value of M is optimized during model training. And Σ is the covariance matrix.

The hidden state of time series is no longer directly generated by LSTM, which is evolved from the final state. The evolution mode is as following.

$$\boldsymbol{h}_h = \boldsymbol{M}\boldsymbol{h}_f \tag{16}$$

where $\boldsymbol{h}_h$ denotes the hidden state of the time series. $\boldsymbol{h}_f$ denotes the latest time state of the historical information. To further meet the dynamic stationary characteristics, an optimization objective is designed based on the linear evolutionary process of the history hidden state. The optimization objective is used to constrain the representation of model learning, which meets the linear stationary characteristics. The specific optimization objective is to minimize L_p.

$$L_p = \sum_{i=1}^{N-1} \left\| \boldsymbol{h}_{i+1} - \boldsymbol{M}\boldsymbol{h}_i \right\|_2^2 + \left\| \boldsymbol{h}_q - \boldsymbol{M}\boldsymbol{h}_f \right\|_2^2 \tag{17}$$

The model is to predict the state of future N time instances by learning the information from the historical sequence. The prediction accuracy is the most important optimization objective. Therefore, the root mean square error is chosen as the optimization function, as shown in equation (18).

$$L_q = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(t_i' - t_i)^2} \tag{18}$$

where t_i is the actual value at time instance i. And t_i' is the predicted value at time instance i.

To prevent overfitting of model training in equation (18), an adjustable parameter α is introduced. α can balance the optimization objectives of model stability and accuracy. The final optimization function is as follows.

$$L' = \min \sum_{j=1}^{N} (L_q + \alpha L_p) \tag{19}$$

IV. Time Series Prediction Process

Considering the data characteristics of the limited time series sample and simplified design for cyclic neural network, the overall framework of ImLSTM is shown in Figure 2. ImLSTM includes three functional modules: input layer, hidden layer, and output layer. The input layer is responsible for the preliminary processing of the original time series, which meets the input requirements. The hidden layer uses ImLSTM cells to build a single-layer cyclic neural network. The output layer provides prediction results. Network training adopts the optimization algorithm mentioned in the previous section. Network prediction adopts an iterative method to predict point by point.

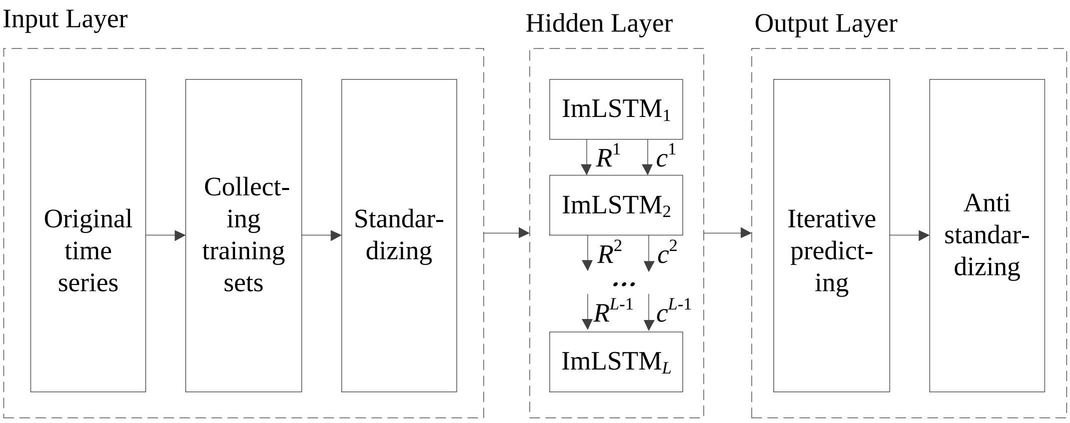


Fig. 2 ImLSTM Based Framework for Time Series Prediction

(1) Model Training

Network training mainly takes the hidden layer as the optimization object. Firstly, the original time series is defined in the input layer. The training and testing set are divided. Then the training set is standardized to X, such as using the classical z-score standardization formula. Secondly, X is taken into the hidden layer. Seen from Figure 2, the hidden layer contains L ImLSTM cells connected in time order.

The adaptive moment estimation (AME) algorithm is an extension of stochastic gradient descent algorithm, which is widely used in deep learning. AME is a first-order optimization algorithm that can replace the traditional stochastic gradient descent process. AME algorithm can iteratively update the neural network weight based on the training data.

The AME algorithm is different from the traditional stochastic gradient descent (SGD) algorithm. SGD algorithm maintains the same single learning rate to update all weights. The learning rate does not change during the whole training process. The AME algorithm takes advantage of both AdaGrad and RMSProp. AME can calculate the adaptive learning rate based on the first-order moment-mean. AME can also make full use of the second-order moment-mean of the gradient.

For AME algorithm, the learning rate of each iteration has a definite range after bias correction, which makes the parameters relatively smooth. The formulas are as follows.

$$n_k = \alpha_1 n_{k-1} + (1-\alpha_1) l_k \tag{20}$$

$$w_k = \alpha_1 w_{k-1} + (1-\alpha_1) l_k^2 \tag{21}$$

$$\overline{n}_k = \frac{n_k}{1-\alpha_1^k} \tag{22}$$

$$\overline{w}_k = \frac{w_k}{1-\alpha_2^k} \tag{23}$$

$$p_{k+1} = p_k - \frac{\beta}{\sqrt{\overline{w}_k + \theta}} \overline{n}_k \tag{24}$$

where n_k and w_k are the first-order moment estimate and second-order moment estimate of the gradient, respectively. $\overline{n}_k$ and $\overline{w}_k$ are the corrected values. In this paper, the learning rate β is 0.005. And the correction coefficients α_1 and α_2 are 0.95 and 0.99, respectively. And θ is 10^{-6}.

(2) Model Predicting

The trained ImLSTM network is used for prediction. The prediction process adopts an iterative method. Then the predicted values are de-normalized to obtain the final prediction sequence.

V. Experimental Verification

A. Experimental Data and Evaluation Index

1. Experimental Data

The experimental dataset is the county-level varicella cases in Hungary from 2004 to 2014. The dataset includes 498 data, of which the first 80% is used to train the time series prediction model. The remaining 20% is used as the testing dataset to verify the prediction performance of the proposed method. The time series of the first 200 data is shown in Figure 3.

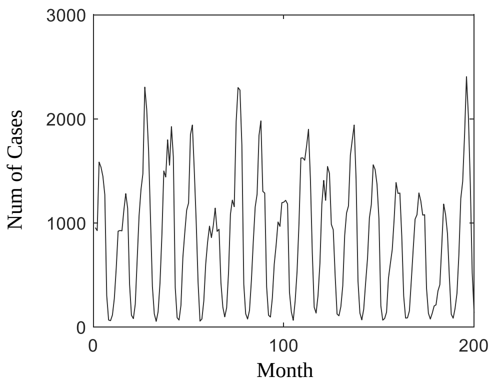


Fig. 3 Time Series Dataset of Varicella Cases

2. Evaluation Index

All prediction models are evaluated according to model accuracy. Root mean square error (RMSE) is selected as the measurement standard. The computation equation of RMSE is as follows.

$$RMSE = \sqrt{\sum_{i=1}^{T} (o_i - y_i)^2 / T} \tag{25}$$

where o_i and y_i are the observation value and prediction value of the time series at time instance i. T is the number of data points. The RMSE values of the training and testing set are calculated to quantitatively evaluate the model fitting and prediction accuracy.

B. Influence of Different Parameters on Prediction

The model parameters of LSTM are mainly determined by experience. Two key parameters of LSTM model are optimized in this section, namely the number of epochs and learning rate.

1. The Number of Epochs

In this experiment, the number of epochs is changed while the other parameters are fixed. The learning rate is fixed to 0.001, while the number of epochs is set to be 100, 200, 300 and 500, respectively. Table 1 shows the predicted RMSE under different number of epochs. Seen from Table 1, the prediction RMSE gradually decreases as the number of epochs increases. When the number of epochs is equal to 200, the prediction accuracy is the highest. When the number increases, the prediction RMSE no longer increases. The experimental result indicates that the large number of epochs is not relevant to high prediction accuracy. When the epoch number is small, the model training is insufficient while the prediction accuracy is reduced. When the epoch number is large, the model is fully trained. But the prediction accuracy does not continue to increase.

The model training loss and RMSE are shown in Figure 4 and Figure 5, respectively. When the epoch number is larger than 200, the model training loss and RMSE are nearly unchanged. The experiment result indicates that an appropriate value of the epoch number should be chosen, rather than the large value.

TABLE I
RMSE ACCORDING TO DIFFERENT EPOCHS

	100	200	300	500
RMSE	122.8352	119.9246	160.9794	131.4583

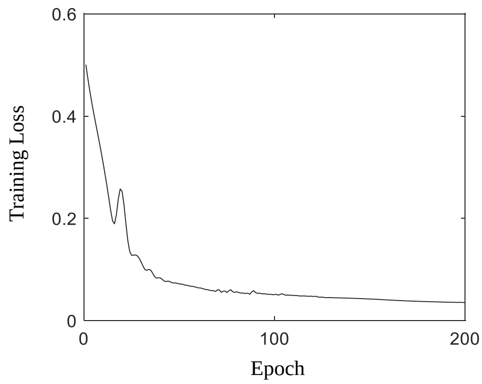


Fig. 4 Training Loss Cure

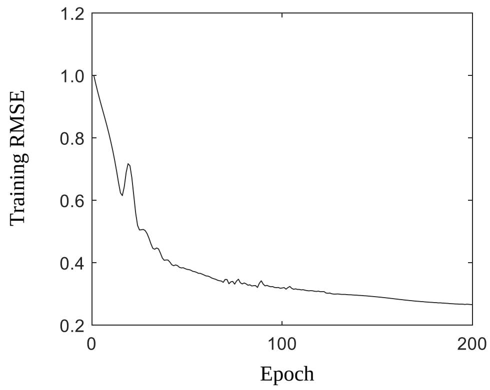


Fig. 5 Training RMSE Cure

2. Learning Rate

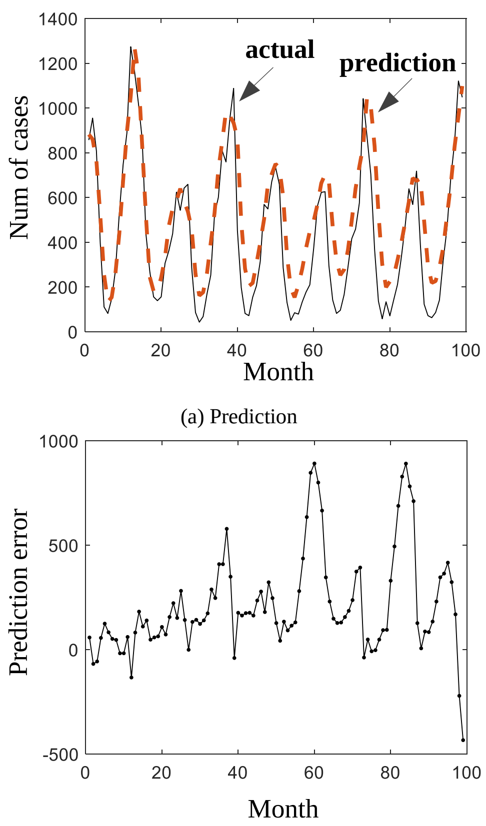
In this experiment, the value of learning rate is set while the other parameters are fixed. Make the epoch num to be 200. And set the learning rate as 0.01, 0.001, 0.0001 and 0.00001, respectively. Table 2 shows the prediction RMSE under different learning rates. Seen from Table 2, with the decrease of the learning rate, the prediction RMSE gradually decreases. When the learning rate is 0.001, the prediction accuracy is the highest. When the learning rate continues to decrease, the prediction RMSE increases. When the number of learning rate is large, the training interval of the prediction model is large. And the prediction accuracy is reduced. When the learning rate is low, the model granularity is good. But the prediction accuracy will not continue to increase. As can be seen from Table 2, an appropriate value of the learning rate should be selected, rather than the large value.

TABLE II
RMSE ACCORDING TO DIFFERENT LERANING RATE

	0.01	0.001	0.0001	0.00001
RMSE	171.8312	119.9246	306.5187	509.3959

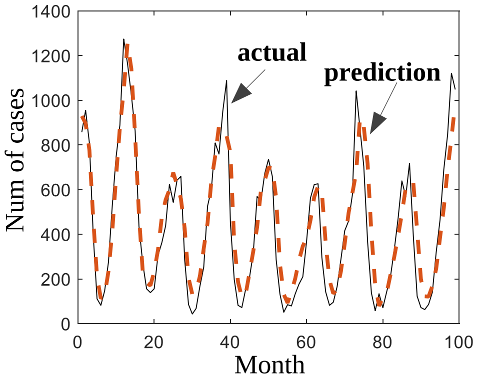
C. Comparison of Prediction Results under Different Methods

After the model parameters are determined, the ImLSTM model is trained and verified. The test dataset is predicted. The prediction RMSE is calculated. The prediction result is compared with the traditional LSTM. The prediction result and RMSE are shown in Figure 6 and Figure 7, respectively. The results show that the prediction accuracy of the proposed ImLSTM is higher than that of the traditional LSTM method, which are 119.9246 and 160.9794 respectively.

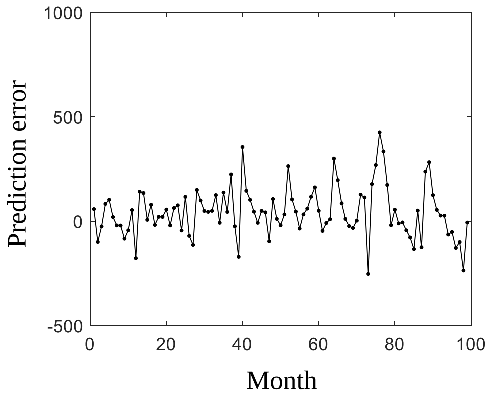


(b) Prediction Error

Fig. 6 Prediction Result of LSTM



(a) Prediction



(b) Prediction error

Fig. 7 Prediction Result of ImLSTM

VI. Conclusion

Aiming at the increasing requirements of time series prediction accuracy, a time series prediction model based on the improved LSTM is proposed. The proposed method improves the prediction accuracy by introducing attention mechanism into LSTM training process. The experimental results show that the prediction RMSE of the proposed method is less than that of the traditional LSTM. At present, the proposed method mainly considers the prediction of time series with one single type. The multi-dimensional correlation time series prediction will be studied in future.

References

[1] Xue Y, Zhang Q, and Neri F, "Self-Adaptive Particle Swarm Optimization-Based Echo State Network for Time Series Prediction," *International Journal of Neural Systems*, vol. 31, no. 12, pp1-3, 2021.

[2] Xiong L, Liu J, and Song B, "Deep learning compound trend prediction model for hydraulic turbine time series," *International Journal of Low-Carbon Technologies*, vol. 3, no. 1, pp3, 2021.

[3] Inyurt S, and Razin M, "Regional application of ANFIS in ionosphere time series prediction at severe solar activity period," *Acta Astronautica*, vol. 179, no. 1, pp450-461, 2021.

[4] Gu Q, Dai Q, and Yu H, "Integrating Multi-Source Transfer Learning, Active Learning and Metric Learning paradigms for Time Series Prediction," *Applied Soft Computing*, vol. 109, no. 3, pp107583, 2021.

[5] Zhang J, Dai Q, and Yao C, "DEP-TSPmeta: a multiple criteria Dynamic Ensemble Pruning technique ad-hoc for time series prediction," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 8, pp2213-2236, 2021.

[6] Gonggui Chen, Pan Qiu, and Xiaorui Hu, Fangjia Long, and Hongyu Long, "Research of Short-Term Wind Speed Forecasting Based on the Hybrid Model of Optimized Quadratic Decomposition and Improved Monarch Butterfly," *Engineering Letters*, vol. 30, no.1, pp73-90, 2022.

[7] Ying Wang, "New Results on a Single-Species Model with Nonlinear Harvesting and Feedback Control on Time Scales," *IAENG International Journal of Applied Mathematics*, vol. 51, no.4, pp886-891, 2021.

[8] Hosseini S M, and Saifoddin A, "Forecasting of CO2 emissions in Iran based on time series and regression analysis," *Energy Reports*, vol. 5, no. 1, pp619-631, 2019.

[9] Liu J L, Chen X, and Zhang T J, "Application of Time Series-Exponential Smoothing Model on Urban Water Demand Forecasting," *Advanced Materials Research*, vol. 183, pp1158-1162, 2011.

[10] Nobrega J P, and Oliveira A, "A Sequential Learning Method with Kalman Filter and Extreme Learning Machine for Regression and Time Series Forecasting," *Neurocomputing*, vol. 337, pp235-250, 2019.

[11] Alaeddini A, Alemzadeh S, and Mesbahi A, "Linear Model Regression on Time-series Data: Non-asymptotic Error Bounds and Applications," *2018 IEEE Conference on Decision and Control (CDC)*, 2019.

[12] Wang Y, Zhang Z, and Yu W, "Pricing equity-linked death benefits by complex Fourier series expansion in a regime-switching jump diffusion model," *Applied Mathematics and Computation*, vol. 399, no. 1, pp126031, 2021.

[13] Lu C, Gu Z J, and Y Yan, "RUL Prediction of Lithium-Ion Battery Based on ARIMA Time Series Algorithm," *Materials Science Forum*, vol. 999, pp117-128, 2020.

[14] Wu C, Yao S, and Du Y, "Production systems optimization of a CBM well based on a time series BP neural network," *Journal of China University of Mining and Technology*, vol. 44, no. 1, pp64-69, 2015.

[15] Liu Y K, Xie F, and Xie C L, "Prediction of time series of NPP operating parameters using dynamic model based on BP neural network," *Annals of Nuclear Energy*, vol. 85, pp566-575, 2015.

[16] Li J, and Liu M, "State-of-charge Estimation of Batteries Based on Open-circuit Voltage and Time Series Neural Network," *6th International Conference on Systems and Informatics*, 2020.

[17] Che Z, Purus Hotham S, and Cho K, "Recurrent Neural Networks for Multivariate Time Series with Missing Values," *Scientific Reports*, vol. 8, no. 1, pp6085, 2018.

[18] Jaramillo J, Velasquez J D, and Franco C J, "Research in Financial Time Series Forecasting with SVM: Contributions from Literature," *IEEE Latin America Transactions*, vol. 15, no. 1, pp145-153, 2017.

[19] Khan A, and Sarfaraz A, "RNN-LSTM-GRU based language transformation," *Soft Computing*, vol. 23, no. 3, 2019.

[20] Su C, Huang H, and Shi S, "Neural machine translation with Gumbel Tree-LSTM based encoder," *Journal of Visual Communication and Image Representation*, vol. 71, pp102811, 2020.

[21] Sabab M N, Chowdhury M, and Nirjhor S, "Bangla Speech Recognition Using 1D-CNN and LSTM with Different Dimension Reduction Techniques," *International Conference for Emerging Technologies in Computing*, 2020.

[22] Yadav A, Jha C K, and Sharan A, "Optimizing LSTM for time series prediction in Indian stock market," *Procedia Computer Science*, vol. 167, pp2091-2100, 2020.

[23] Karim F, Majumdar S, and Darabi H, "Insights into LSTM Fully Convolutional Networks for Time Series Classification," *IEEE Access*, vol. 7, pp67718-67725, 2019.

[24] Trung C V, Dinh M N, and Dimla E, "Predicting Phase-field Behavior of Brittle Fracture Model based on LSTM Time Series Forecasting Model," *2020 RIVF International Conference on Computing and Communication Technologies (RIVF)*, 2020.

[25] Yadav A, Jha C K, and Sharan A, "Optimizing LSTM for time series prediction in Indian stock market," *Procedia Computer Science*, vol. 167, pp2091-2100, 2020.

Huang Sen (1982.10-), male, acts as a lecturer in the Henan Vocational College of Agriculture. His interest is computer related research.