


## Research Article

# Optimization of Traffic Congestion Management in Smart Cities under Bidirectional Long and Short-Term Memory Model

Yujia Zhai, Yan Wan, and Xiaoxiao Wang 

Faculty of Humanities and Arts, Macau University of Science and Technology, Avenida Wai Long, Taipa 999078, Macau, China

Correspondence should be addressed to Xiaoxiao Wang; [120212202032@ncepu.edu.cn](mailto:120212202032@ncepu.edu.cn)

Received 4 January 2022; Accepted 9 February 2022; Published 1 April 2022

Academic Editor: Sang-Bing Tsai

Copyright © 2022 Yujia Zhai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To solve the increasingly serious traffic congestion and reduce traffic pressure, the bidirectional long and short-term memory (BiLSTM) algorithm is adopted to the traffic flow prediction. Firstly, a BiLSTM-based urban road short-term traffic state algorithm network is established based on the collected road traffic flow data, and then the internal memory unit structure of the network is optimized. After training and optimization, it becomes a high-quality prediction model. Then, the experimental simulation verification and prediction performance evaluation are performed. Finally, the data predicted by the BiLSTM algorithm model are compared with the actual data and the data predicted by the long short-term memory (LSTM) algorithm model. Simulation comparison shows that the prediction results of LSTM and BiLSTM are consistent with the actual traffic flow trend, but the data of LSTM deviate greatly from the real situation, and the error is more serious during peak periods. BiLSTM is in good agreement with the real situation during the stationary period and the low peak period, and it is slightly different from the real situation during the peak period, but it can still be used as a reference. In general, the prediction accuracy of the BiLSTM algorithm for traffic flow is relatively high. The comparison of evaluation indicators shows that the coefficient of determination value of BiLSTM is 0.795746 greater than that of LSTM (0.778742), indicating that BiLSTM shows a higher degree of fitting than the LSTM algorithm, that is, the prediction of BiLSTM is more accurate. The mean absolute percentage error (MAPE) value of BiLSTM is 9.718624%, which is less than 9.722147% of LSTM, indicating that the trend predicted by the BiLSTM is more consistent with the actual trend than that of LSTM. The mean absolute error (MAE) value of BiLSTM (105.087415) is smaller than that of LSTM (106.156847), indicating that its actual prediction error is smaller than LSTM. Generally speaking, BiLSTM shows advantages in traffic flow prediction over LSTM. Results of this study play a reliable reference role in the dynamic control, monitoring, and guidance of urban traffic, and congestion management.

## 1. Introduction

With the continuous development of society, the process of urbanization is also accelerating and the traffic congestion is getting more and more serious. According to the data released by the Traffic Management Bureau of the Ministry of Public Security, the number of motor vehicles has reached 390 million as of the end of September 2021, of which 297 million are cars; and there are 476 million motor vehicle drivers nationwide, of which 439 million are car drivers. In the first quarter alone, the number of newly registered motor vehicles nationwide reached 27.53 million, a year-on-year increase of 4.363 million, and a growth rate of 18.83% [1]. Nowadays, it is urgent to tackle the urban traffic congestion.

With the advent of the era of big data and the rise of the “Internet +” boom, smart transportation engineering has gradually been developed [2]. Smart transportation engineering integrates vehicle networking technology, artificial intelligence (AI) technology, automatic control technology, computer technology, information and communication technology, and electronic sensor technology to build a unified cross-regional transportation information resource sharing platform and comprehensively manage information resources in the transportation field, realizing intelligent management of road operations. It is a real-time, accurate, and efficient comprehensive transportation management system that is applied to the entire ground transportation management system and established for a large-scale, all-

round function. By constructing a short-term traffic state prediction mechanism for urban roads, it can grasp the traffic conditions of traffic roads in a period of time in the future, use the prediction results for traffic guidance, improve the utilization rate of urban road resources, and alleviate the traffic pressure on the urban-congested sections [3].

In the traditional traffic flow prediction method, some scholars consider multiple factors that affect the flow, so they use multiple linear regression to predict the traffic flow, but the real-time performance is low. Kumar and Vanajakshi (2015) considered the periodicity of traffic flow, fitted a model for real-time data statistical processing, and applied the seasonal autoregressive integrated moving average (ARIMA) model to short-term traffic flow prediction. However, the nonlinear and uncertain fitting of traffic flow is poor, and it is not suitable for short-term traffic flow prediction [4]. Huang et al. (2014) proposed a deep belief network to extract traffic flow features and a top-level multitask regression two-layer deep learning model [5]. Zhang et al. (2020) adopted the fast graph convolution recurrent neural network (FastGCRNN) to model the spatiotemporal dependence of traffic flow [6]. Xia et al. (2021) developed a distributed modeling framework for traffic flow prediction on MapReduce under the Hadoop distributed computing platform, which solved the storage and computing problems existing in the single-machine learning model processing large-scale traffic flow data [7]. These deep learning models only consider one-way traffic flow data for prediction, ignoring the change law of traffic flow data after the prediction time point. With increasing emphasis on the application of traffic big data and the creation of smart transportation cities, the bidirectional long short-term memory (BiLSTM) algorithm is applied to the traffic flow prediction. The innovation of the research is to model the traffic flow data through the BiLSTM model and to analyze the influence of the time series change rules of the front and rear traffic flow on the short-term prediction. Firstly, a BiLSTM-based urban road short-term traffic state algorithm network is established based on the collected road traffic flow data, and then the internal memory unit structure of the network is optimized. After training and optimization, it becomes a high-quality prediction model. Then, the experimental simulation verification and prediction performance evaluation are performed. Results of this study play a reliable reference role in the dynamic control, monitoring, and guidance of urban traffic, and congestion management.

## 2. Materials and Methods

**2.1. Standard LSTM Algorithm.** Among various deep neural networks, recurrent neural network (RNN) is widely used in the prediction of time series, but long-term series data have caused gradient explosion and gradient disappearance [8]. Therefore, Hochreite and Schmidhuber proposed long short-term memory (LSTM) in 1997. LSTM is an improved RNN that has the function of long-term memory information and solves the problem of RNN gradient disappearance. Its structure includes a module chain structure,

but it only changes the hidden layer module structure [9]. A memory block is added to the hidden layer of LSTM to realize its memory function. The memory block is composed of a set of iteratively connected subnets. Each subnet has one or more storage units, which are connected to each other. The memory block contains three multiplication units consisting of gates: input gates, output gates, and forget gates. All three gates have a nonlinear summation function, which contains two activation functions to control the amount of data transfer [10]. The internal structure diagram of the storage unit is shown in Figure 1.

In Figure 1, the three modules were used as three storage units in the memory block, which were connected to each other, and each of them contains three multiplication units composed of gates: an input gate, output gate, and forget gate.  $X$  is the input,  $h$  is the weight of the neuron,  $\tanh$  is the activation function, and  $\sigma$  is the sigmoid activation function. On the basis of this kind of chain, LSTM improves the interior of the module, using 3 sigmoid neural network layers and a gate composed of point-by-point multiplication to strengthen the control ability of information [11]. The  $\tanh$  activation function mainly processes data for state and output functions [12]. It indicates the input gate,  $f_t$  represents the forget gate,  $C_t$  indicates internal memory,  $o_t$  represents the output gate, and  $h_t$  represents the output of the LSTM unit at time  $t$ . The input gate controls the input of the output information of the upper unit to the unit information of this layer and retains the previous information of the sequence. The calculation equation for each threshold layer is given as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \quad (1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \quad (2)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \quad (3)$$

where  $W$  is the weight of the threshold layer and  $b$  is the offset of the threshold layer. After each threshold layer is updated, the internal memory  $C_t$  is updated with the following equation:

$$C_t = f_t * C_{t-1} + i_t * \tanh(W_c \cdot [h_{t-1}, x_t] + b_c), \quad (4)$$

where " $W_c$ " and " $b_c$ " are the weights and offsets, respectively. The neural network output weight  $h_t$  of the internal memory is controlled by the output gate, and the activated unit state is output to the next layer of neural network and chain unit, which is specialized as shown in the following equation:

$$h_t = o_t * \tanh(C_t), \quad (5)$$

where  $\sigma$  is the sigmoid activation function. The sigmoid activation function takes the memory state of the network as the output value. When traffic flow data are input to the sigmoid activation function, the sigmoid activation function will compress it to  $[0, 1]$ : 0 means no amount is allowed to pass and 1 means any amount can pass. If the output value is within the specified range, the output value is matrix multiplied with the calculation result of the current layer, and then the result is input into the lower layer to map the

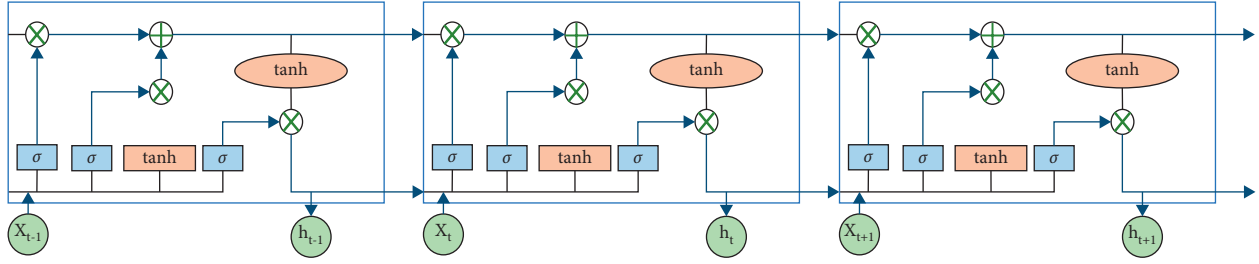


FIGURE 1: The internal structure of LSTM.

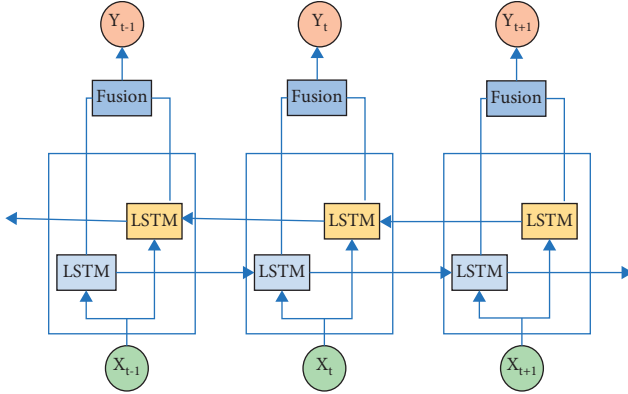


FIGURE 2: Structure of BiLSTM.

real number domain to the range of  $[0,1]$ . The function value represents the probability of belonging to the positive class [13–15]. Its expression is shown in the following equation:

$$f(z) = \frac{1}{1 + e^{-z}}. \quad (6)$$

The tanh activation function is different from the sigmoid activation function. It can map the real number domain to the range of  $[-1,1]$ . When the input is 0, the output is also 0. The expression is shown as follows:

$$f(x)_{\tanh} = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \quad (7)$$

In the neural network training stage, LSTM learns the weights and offsets of each threshold layer from the past information. In the real-time prediction stage, the trained model is used to calculate the input data to obtain the predicted value of the time series, thereby improving the efficiency of mining past information and shortening the training time [16].

BiLSTM is an improved version of LSTM, composed of forward standard LSTM and reverse standard LSTM. By adding a layer of reverse LSTM to the LSTM structure, the effect of extracting global data features is achieved [17]. BiLSTM uses the memory unit of the standard LSTM to calculate the input data in order and in reverse order to obtain two different hidden layer features. Although it is carried out simultaneously, the structures in the two directions do not share the hidden state. The hidden state data of the forward LSTM are transmitted to the forward LSTM, the hidden state data of the reverse LSTM are transmitted to

the reverse LSTM, and there is no connection between the two directions. Finally, the two hidden layer features are linearly fused, and the final hidden layer feature result is obtained. In BiLSTM, the output value at each moment is jointly determined by the LSTM in the two directions. Therefore, the obtained model takes into account the parameter factors of the past and future directions, and the accuracy of the algorithm's prediction has been greatly improved [18–20]. Its specific structure is shown in Figure 2.

The BiLSTM structure is divided into two parts: one part is the forward standard LSTM, which is calculated in the forward direction over time and outputs  $h$ . The other part is the reverse standard LSTM, which performs reverse operation over time. The essence of the reverse operation is to reverse the input traffic flow data, then output to the reverse LSTM, and finally output  $H$ . After the forward and reverse output results are fused, the final output result [21] is obtained. In this process, the state of the hidden layer at time  $t$  in the forward LSTM calculation is related to the state at time  $t-1$  and the state of the hidden layer at time  $t$  in the reverse LSTM calculation is related to the state at time  $t+1$ . The training is realized using a loss function [22]. The specific BiLSTM derivation principle is shown in Figure 3.

In BiLSTM, the calculation equations for the thresholds of the forward standard LSTM are consistent with those for the thresholds of the standard LSTM, and the calculation equations for the thresholds of the reverse standard LSTM are shown as follows:

$$\begin{aligned} i_t &= \sigma(W_i \cdot [H_{t+1}, X_t] + b_i), \\ f_t &= \sigma(W_f \cdot [H_{t+1}, X_t] + b_f), \\ o_t &= \sigma(W_o \cdot [H_{t-1}, x_t] + b_o), \\ C_t &= f_t * C_{t+1} + i_t * \tanh(W_C \cdot [H_{t+1}, x_t] + b_C). \end{aligned} \quad (8)$$

The output results of the bidirectional LSTM are linearly fused, and the fusion equation is shown as follows:

$$\begin{aligned} Y_t &= g(Uh_t + b) \\ &= g(U[h_t; H_t] + b). \end{aligned} \quad (9)$$

The forward direction extracts past information features from time 1 to  $t$ , and the reverse direction extracts future information features from time  $t$  to 1. The combined training of forward and reverse will reconsider the factors considered or discarded. Therefore, BiLSTM is more comprehensive than LSTM training and the prediction will be more accurate

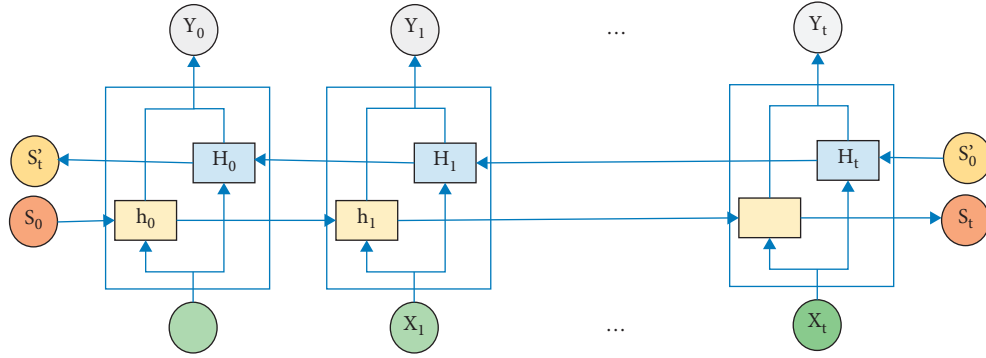


FIGURE 3: Specific derivation principle of BiLSTM.

[23, 24]. The final calculation results are obtained through training in BiLSTM, and the calculation steps are as follows.

*Step 1.* Defining the initial value. When  $t=1$  is set, the weight derivative value is calculated as shown in the following equations:

$$\varepsilon_C^t = \frac{\partial l}{\partial I_j^t}, \tag{10}$$

$$\varepsilon_C^t = \frac{\partial l}{\partial S_C^t}, \tag{11}$$

where  $l$  represents the loss function used for training,  $S_C^t$  means that neuron  $C$  is at time  $t$ , and  $I_j^t$  represents the data value input to neuron  $j$  at time  $t$ .

*Step 2.* Calculating the weight of the output gate. Since the output gate does not involve the time dimension, the weight of the output gate is shown in equation (12). In the equation, “ $O_o^t$ ” represents the data value output from the output gate at time  $t$  and  $h$  represents the output activation function of the internal memory  $c$ :

$$W_o = W_o - \eta f'(O_o^t) \sum_{c=1}^C h(S_C^t) \varepsilon_C^t. \tag{12}$$

*Step 3.* Calculating the weight of the forget gate. The weight of the forget gate is shown in the following equation:

$$W_f = W_f - \eta f'(F_f^t) \sum_{c=1}^C S_C^t \varepsilon_S^t, \tag{13}$$

where “ $F_f^t$ ” represents the output data value of the forget gate at time  $t$  and “ $\varepsilon_S^t$ ” represents the state of the output gate at time  $t$ .

*Step 4.* Calculating the weight of the input gate. The weight of the input gate is shown in the following equation:

$$W_i = W_i - \eta f'(F_i^t) \sum_{c=1}^C g(I_C^t) \varepsilon_S^t, \tag{14}$$

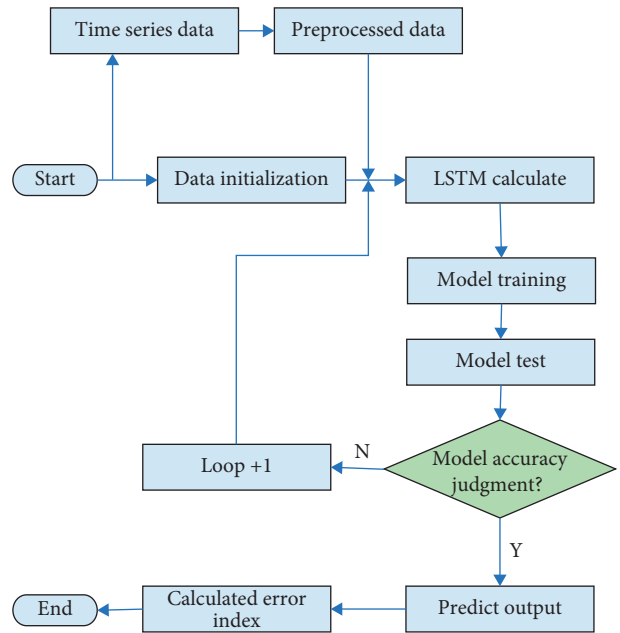


FIGURE 4: Flow of LSTM.

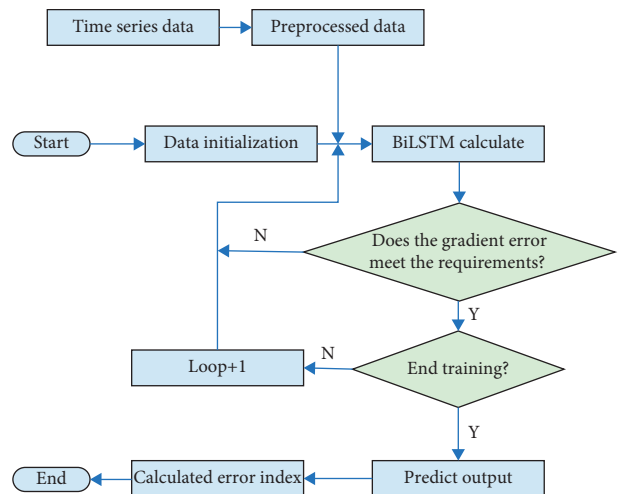


FIGURE 5: Flow of BiLSTM.

TABLE 1: Experimental procedure and experimental steps.

Experimental procedure	Experimental steps
Input	Traffic flow raw data, including time, direction, lanes, and numbers
Firstly, the traffic flow data are preprocessed	Step 1: the original traffic flow data is cleaned and filtered
	Step 2: a direction is determined as the research data
	Step 3: the traffic flows of all lanes in the selected direction are collected and summarized
	Step 4: the generated data set is outputted
Secondly, the training set is trained based on the BiLSTM algorithm	Step 1: the BiLSTM algorithm is initialized
	Step 2: the training data is inputted
	Step 3: the data enter the forward LSTM and are processed
	Step 4: the data enter the reverse LSTM and are processed
	Step 5: linear fusion is performed for the forward processing result and the reverse processing result
Thirdly, 20% of traffic flow data is selected to make predictions	Step 1: the forward and reverse BiLSTM algorithms are loaded
	Step 2: the predicted result values are loaded
	Step 3: the final result is obtained through linear fusion processing
Output	Prediction results and prediction performance evaluation values

where “ $F_i^t$ ” represents the output data value of the input gate at time  $t$  and “ $I_C^t$ ” represents the state of the unit at time  $t$ .

It has to calculate the weight of LSTM twice in the calculation process, and finally, 8 parameter values are obtained.

**2.2. Count Affects.** The data set is divided into a training set and a prediction set, which is allocated according to the ratio of 8:2 between the training set and the prediction set. By learning the training set, the algorithm finds the most suitable weights and confirms the values of related factors. The specific LSTM algorithm and BiLSTM algorithm execution flow are shown in Figures 4 and 5, respectively.

As shown in Figure 5, the standard LSTM algorithm can only perform single-item training of data, which means that it can perform feature extraction on existing data, but cannot perform data prediction. The use of BiLSTM for two-way training can process both past data and future data, so that the overall accuracy of the model will be higher [25]. In BiLSTM, the most important step is to train the data. The main purpose of training is to find suitable weights and related factor values through the training set [26].

Features with larger data levels have a greater impact on predictions, and it will cause the algorithm to converge slowly, so the data need to be preprocessed. If the degree of fitting in the linear form is too low (i.e., underfitting), it will not be fully suitable for the training set; if the degree of fitting is overfitted in the high power form, it will affect the prediction results although it is very suitable as a training set. Therefore, when the fitting is not suitable, some unimportant features can be directly discarded or normalization can be performed to reduce the number of parameters [27]. Firstly, the data are cleaned and filtered to find and correct the wrong and invalid values in the data. In this data processing, vehicles in multiple directions are firstly screened, in which vehicles in one direction are screened out and the vehicles in the other directions are discarded. Then, the traffic flows of multiple lanes are summed as data for one direction. Next, the

traffic flow data is serialized and mapped to  $[0, 1]$ . The original data  $X = \{x_1, x_2, \dots, x_n\}$  are converted, and the conversion function is shown in the following equation:

$$y_i = \frac{x_i - \min\{x_j\}}{\max\{x_j\} - \min\{x_j\}}, \quad (1 \leq j \leq n). \quad (15)$$

In the above equation, max is the maximum value of the data and min is the minimum value of the data. The new traffic sequence obtained by conversion is  $Y = \{y_1, y_2, \dots, y_n\}$ . Finally, 80% of the data is selected as the training set and 20% as the prediction set.

This experiment is mainly divided into three steps as follows: Step 1: the traffic flow data are preprocessed, filtered, cleaned, and standardized to obtain the time series of traffic flow data. Step 2: the training set is trained based on the BiLSTM algorithm to get a suitable model. Step 3: about 20% of the traffic flow data is adopted to make predictions. The specific steps are listed in Table 1.

In traffic flow data prediction, root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), mean square error (MSE), and coefficient of determination  $R^2$  (R-square) are adopted to evaluate the algorithm in order to reflect the degree of fitting of the algorithm and the accuracy of prediction. The MAPE value reflects the degree of deviation between the predicted result and the actual result. It is suitable for different algorithms of the same set of data. The smaller the MAPE value, the smaller the degree of deviation, indicating the better the prediction effect. MSE reflects a measure of the degree of difference between the predicted result and the actual result. Its value can reflect the degree of change and the distribution of errors. The smaller the MSE, the more concentrated the error distribution and the better the prediction effect. RMSE is used to evaluate the applicability of prediction algorithms to actual data. MAE reflects the average value of the absolute value of the deviation between the prediction result and the arithmetic average and can accurately reflect the magnitude of the prediction error.  $R^2$

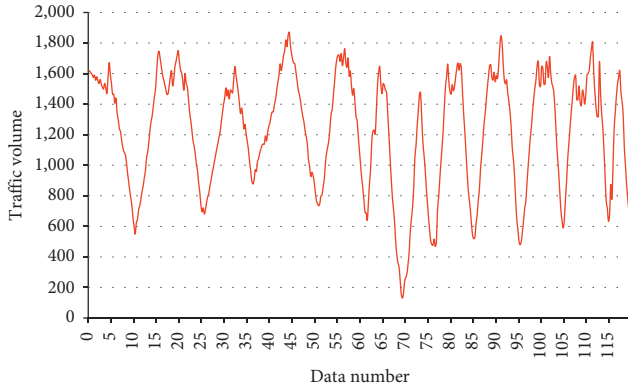


FIGURE 6: Test results of actual traffic flow.

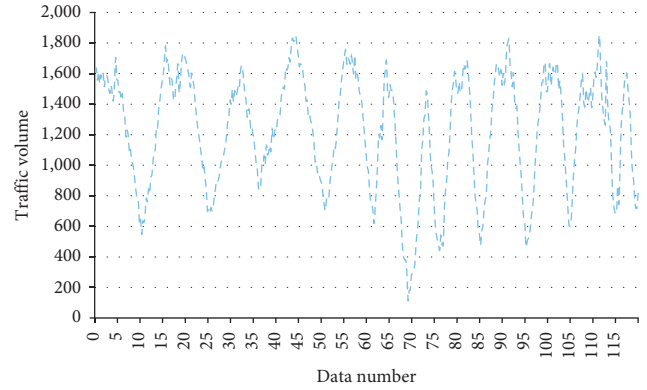


FIGURE 8: Predicted results of BiLSTM.

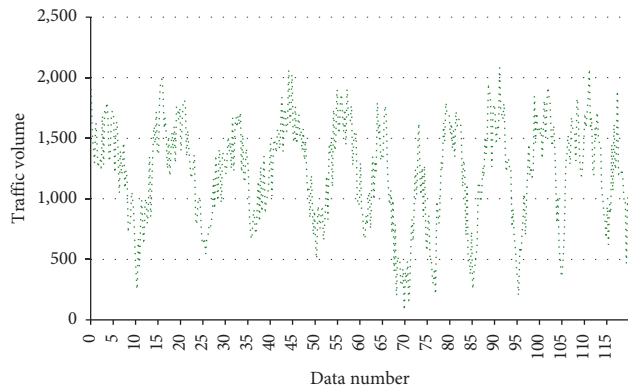


FIGURE 7: Predicted results of LSTM.

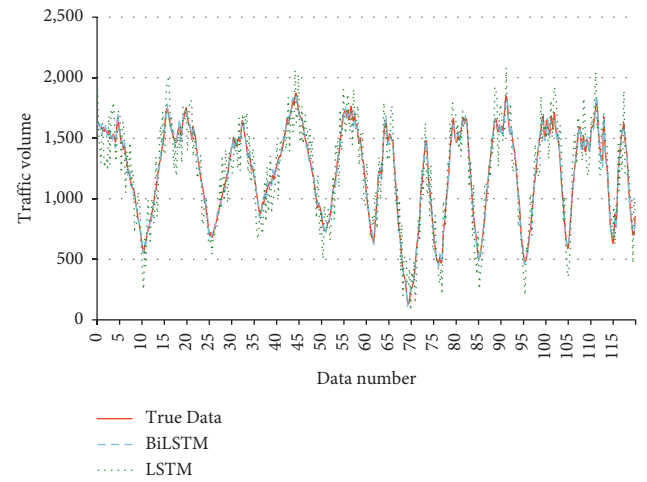


FIGURE 9: Comparison on prediction results of BiLSTM and LSTM.

reflects the degree of fitting between the predicted value and the actual result. The value range is [0, 1]. The closer to 1, the higher the degree of fitting, and the closer to 0, the lower the degree of fitting. These evaluation indicators can evaluate not only the forecast data and actual data but also the suitability of the forecast algorithm. Comprehensive consideration of trend graphs and evaluation indicators can further reflect the applicability of the algorithm in the field of traffic flow prediction. Their expressions are shown in the following equations:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|, \quad (16)$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|}{y_i}, \quad (17)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (|y_i - \hat{y}_i|)^2, \quad (18)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (|y_i - \hat{y}_i|)^2}, \quad (19)$$

TABLE 2: Indicators for prediction performance of BiLSTM.

Item	LSTM	BiLSTM
MAE	106.156847	105.087415
MAPE	9.722147%	9.718624%
MSE	31354.984156	32846.946157
RMSE	176.184617	180.145275
R <sup>2</sup>	0.778742	0.795746

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y}_i)^2}. \quad (20)$$

Here,  $y$  represents the actual traffic flow observed,  $\hat{y}$  represents the corresponding time prediction value,  $\bar{y}$  represents the average value,  $i$  refers to the amount of change in the traffic flow, and  $N$  represents the data volume of the traffic flow prediction experiment.

### 3. Results and Discussion

**3.1. Effect of BiLSTM.** In this study, the traffic flow measured by a vehicle detector at an intersection in Beilin District, Xi'an is selected. The duration lasts 10 days from November

1 to 10, 2021, and the traffic flow statistics interval is 2 hours. A total of 120 sets of data are measured. The actual trend, the BiLSTM result trend, and the LSTM trend predicted by the construction training and prediction are shown in Figures 6–8.

As illustrated in Figures 6 to 8, both the LSTM algorithm and the BiLSTM algorithm could roughly predict the real traffic flow data, but the fitting degree of the LSTM algorithm at peak and low peaks was a little bit worse, and the calculation results were more chaotic after simulation. The above three figures are fused, and it can show the difference between the prediction result and the prediction result, as shown in Figure 9.

In the figure, the abscissa is the data number and the interval is 2 hours. A total of 120 sets of data are measured. The ordinate is the traffic flow on the road, and the interval is 500. The figure reveals that the prediction results of LSTM and BiLSTM are consistent with the actual traffic flow trend overall, but the data of LSTM deviate greatly from the real situation, and the error is more serious during peak periods. BiLSTM is in good agreement with the real situation in the stationary period and low peak period and is slightly different from the real situation in the peak period but still has a better deviation than LSTM. In general, the prediction accuracy of the BiLSTM algorithm for the traffic flow is higher than that of the LSTM algorithm, which is in good agreement with the real situation, so it is feasible in the actual traffic flow prediction.

**3.2. Algorithm Evaluation.** The MAE, MAPE, MSE, RMSE, and  $R^2$  values of the prediction results were calculated, as listed in Table 2.

As presented in Table 2, the  $R^2$  value of BiLSTM is larger than that of LSTM, indicating that BiLSTM shows a higher degree of fitting than LSTM algorithm, and BiLSTM predicts more accurately and is more suitable for predicting traffic flow than LSTM. The MAPE value of BiLSTM is smaller than that of LSTM, indicating that the trend of BiLSTM's prediction results is more consistent than that of LSTM. The MAE value of BiLSTM is smaller than that of LSTM, indicating that the actual prediction error of BiLSTM is smaller than that of LSTM, and the deviation of each data compared with the real data is smaller. The MSE and RMSE values of the BiLSTM algorithm are larger than those of the LSTM, indicating that the LSTM is more concentrated and the effect is better than the BiLSTM. Generally speaking, the BiLSTM algorithm shows more advantages in traffic flow prediction than the LSTM algorithm.

## 4. Conclusions

This study mainly applies the traffic flow algorithm based on the BiLSTM model. By predicting the traffic flow, the traffic congestion can be managed and optimized. A BiLSTM-based urban road short-term traffic state algorithm network is established based on the collected traffic flow data, and then its internal memory unit structure is optimized. In addition, it is trained to be a high-quality prediction model,

and experimental simulation verification and predictive performance evaluation are performed. Experiments show that BiLSTM is in good agreement with the real situation in the traffic stationary period and low peak period, and there is a slight gap between the peak period and the real situation, but it can still be used as a reference. In general, the prediction accuracy of the BiLSTM algorithm for traffic flow is relatively high, so it is feasible in actual traffic flow prediction. Due to the limited capabilities, a slight flaw can be found in the design of the fusion function of the BiLSTM algorithm, which leads to a decrease in the accuracy of the prediction result. In future, it will conduct in-depth exploration in this aspect to find a more suitable fusion function to reduce the influence of human factors. All in all, this study can play a certain reference role in the dynamic control, monitoring, and guidance of urban traffic, and congestion management.

## Data Availability

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This paper was supported by Macau University of Science and Technology Foundation (FRG-21-016-FA).

## References

- [1] A. Khanna, R. Goyal, M. Verma, and D. Joshi, "Intelligent traffic management system for smart cities," *Communications in Computer and Information Science*, vol. 958, pp. 152–164, 2019.
- [2] P. Yuan and X. Lin, "How long will the traffic flow time series keep efficacious to forecast the future?" *Physica A: Statistical Mechanics and Its Applications*, vol. 467, pp. 419–431, 2017.
- [3] T. Li, A. Ni, C. Zhang, G. Xiao, and L. Gao, "Short-term traffic congestion prediction with Conv-BiLSTM considering spatio-temporal features," *IET Intelligent Transport Systems*, vol. 14, no. 14, pp. 1978–1986, 2020.
- [4] S. V. Kumar and L. Vanajakshi, "Short-term traffic flow prediction using seasonal ARIMA model with limited input data," *European Transport Research Review*, vol. 7, no. 3, p. 21, 2015.
- [5] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.
- [6] Y. Zhang, M. Lu, and H. Li, "Urban traffic flow forecast based on FastGCRNN," *Journal of Advanced Transportation*, vol. 2020, Article ID 8859538, 9 pages, 2020.
- [7] D. Xia, M. Zhang, X. Yan et al., "A distributed WND-LSTM model on MapReduce for short-term traffic flow prediction," *Neural Computing & Applications*, vol. 33, no. 7, pp. 2393–2410, 2021.

- [8] H. Hansika, B. Christoph, and B. Kasun, "Recurrent neural networks for time series forecasting: current status and future directions," *International Journal of Forecasting*, vol. 37, no. 1, pp. 388–427, 2021.
- [9] M. Abdel-Nasser and K. Mahmoud, "Accurate photovoltaic power forecasting models using deep LSTM-RNN," *Neural Computing & Applications*, vol. 31, no. 7, pp. 2727–2740, 2019.
- [10] B. B. Sahoo, R. Jha, A. Singh, and D. Kumar, "Long Short-term Memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting," *Acta Geophysica*, vol. 67, no. 5, pp. 1471–1481, 2019.
- [11] F. Affonso, T. M. R. Dias, and A. L. Pinto, "Financial times series forecasting of clustered stocks," *Mobile Networks and Applications*, vol. 26, no. 1, pp. 256–265, 2021.
- [12] M. Rhif, A. B. Abbes, B. Martinez, and I. R. Farah, "A deep learning approach for forecasting non-stationary big remote sensing time series," *Arabian Journal of Geosciences*, vol. 13, no. 22, pp. 1–11, 2020.
- [13] J. Kumar, R. Goomer, and A. K. Singh, "Long short term memory recurrent neural network (lstm-rnn) based workload forecasting model for cloud datacenters," *Procedia Computer Science*, vol. 125, pp. 676–682, 2018.
- [14] S. Bouktif, A. Fiaz, A. Ouni, and M. ASerhani, "Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting," *Energies*, vol. 13, no. 2, pp. 1–21, 2020.
- [15] S. V. Belavadi, S. Rajagopal, R. R. and R. Mohan, "Air quality forecasting using LSTM RNN and wireless sensor networks," *Procedia Computer Science*, vol. 170, pp. 241–248, 2020.
- [16] V. Athira, P. Geetha, R. Vinayakumar, and K. P. Soman, "Deep air net: applying recurrent networks for air quality prediction," *Procedia Computer Science*, vol. 132, pp. 1394–1403, 2018.
- [17] T. Chen, R. Xu, Y. He, and X. Wang, "Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN," *Expert Systems with Applications*, vol. 72, pp. 221–230, 2017.
- [18] L. Luo, Z. Yang, P. Yang et al., "An attention-based BiLSTM-CRF approach to document-level chemical named entity recognition," *Bioinformatics*, vol. 34, no. 8, pp. 1381–1388, 2018.
- [19] M. Liu, Y. H. Lu, S. Long, J. Bai, and W. Lian, "An attention-based CNN-BiLSTM hybrid neural network enhanced with features of discrete wavelet transformation for fetal acidosis classification," *Expert Systems with Applications*, vol. 186, 2021.
- [20] J. Zheng, L. Zhang, J. Chen et al., "Multiple-load forecasting for integrated energy system based on Copula-DBiLSTM," *Energies*, vol. 14, pp. 1–14, 2021.
- [21] X. Y. Yan, T. Guan, K. X. Fan, and Q. Sun, "Novel double layer BiLSTM minor soft fault detection for sensors in air-conditioning system with KPCA reducing dimensions," *Journal of Building Engineering*, vol. 44, Article ID 102950, 2021.
- [22] C. Gan, Q. Feng, and Z. Zhang, "Scalable multi-channel dilated CNN-BiLSTM model with attention mechanism for Chinese textual sentiment analysis," *Future Generation Computer Systems*, vol. 118, pp. 1–2, 2021.
- [23] J. Deng, L. Cheng, and Z. Wang, "Attention-based BiLSTM fused cnn with gating mechanism model for Chinese long text classification," *Computer Speech & Language*, vol. 68, no. 6, 2021.
- [24] L. Wang, X. Bai, R. Xue, and F. Zhou, "Few-shot SAR automatic target recognition based on Conv-BiLSTM prototypical network," *Neurocomputing*, vol. 443, no. 12, pp. 235–246, 2021.
- [25] T. Hu, K. Li, H. Ma, H. Sun, and K. Liu, "Quantile forecast of renewable energy generation based on indicator gradient descent and deep residual BiLSTM," *Control Engineering Practice*, vol. 114, no. 99, Article ID 104863, 2021.
- [26] H. Wei, A. Zhou, Y. Zhang, F. Chen, W. Qu, and M. Lu, "Biomedical event trigger extraction based on multi-layer residual BiLSTM and contextualized word representations," *International Journal of Machine Learning and Cybernetics*, vol. 18, pp. 1–13, 2021.
- [27] R. Janning, T. Horváth, A. Busche, and L. Schmidt-Thieme, "GamRec: a clustering method using geometrical background knowledge for GPR data preprocessing," *IFIP Advances in Information and Communication Technology*, vol. 381, pp. 347–356, 2017.