applied sciences

MDPI

Article

Short-Term Traffic Flow Prediction Based on a K-Nearest Neighbor and Bidirectional Long Short-Term Memory Model

Weiqing Zhuang [1] and Yongbo Cao [2,*]

[1] School of Internet Economics and Business, Fujian University of Technology, Fuzhou 350014, China
[2] School of Transportation, Fujian University of Technology, Fuzhou 350118, China
* Correspondence: cyb1205389533@163.com

Abstract: In the previous research on traffic flow prediction models, most of the models mainly studied the time series of traffic flow, and the spatial correlation of traffic flow was not fully considered. To solve this problem, this paper proposes a method to predict the spatio-temporal characteristics of short-term traffic flow by combining the k-nearest neighbor algorithm and bidirectional long short-term memory network model. By selecting the real-time traffic flow data observed on high-speed roads in the United Kingdom, the K-nearest neighbor algorithm is used to spatially screen the station data to determine the points with high correlation and then input the BILSTM model for prediction. The experimental results show that compared with SVR, LSTM, GRU, KNN-LSTM, and CNN-LSTM models, the model proposed in this paper has better prediction accuracy, and its performance has been improved by 77%, 19%, 18%, 22%, and 13%, respectively. The proposed K-nearest neighbor-bidirectional long short-time memory model shows better prediction performance.

Keywords: nearest neighbor algorithm; bidirectional long and short-term memory model; traffic flow

check for updates

Citation: Zhuang, W.; Cao, Y. Short-Term Traffic Flow Prediction Based on a K-Nearest Neighbor and Bidirectional Long Short-Term Memory Model. *Appl. Sci.* **2023**, *13*, 2681. https://doi.org/10.3390/app13042681

Academic Editor: Carla Raffaelli

Received: 20 December 2022
Revised: 15 February 2023
Accepted: 16 February 2023
Published: 19 February 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the development of the social economy, science and technology, and the acceleration of urbanization, the number of automobiles has increased rapidly. The resulting problems of traffic congestion and right-of-way distribution are also becoming increasingly obvious and seriously affect traffic safety and efficiency. The construction of an intelligent transportation system (ITS) can effectively alleviate road congestion, shorten travel time, reduce pollution, and improve traffic safety. Accurate prediction of short-term traffic flow is the core issue of ITS, which can provide monitoring and technical support of traffic flow in a certain period in the future. Timely and accurate prediction of traffic flow is the basis and prerequisite for traffic management and travel route planning. This can help managers to take appropriate preventive measures to make travelers choose more suitable routes, thus reducing congestion on roads and improving the efficiency of distribution between roads. Among the various applications of ITS, traffic flow prediction has attracted much attention in recent decades. However, this remains a difficult topic for transportation researchers.

Traffic flow forecasting is divided into short-term traffic flow forecasting and medium- and long-term traffic flow forecasting, depending on the time interval. The medium-term and long-term forecast units are generally based on days, weeks, months, and years. Due to the large interval, the data stability is relatively good, so it is often used for forecasting. The short-term traffic flow forecast is generally at intervals of 5–15 min. Due to the short time interval, the data stability is relatively poor, the complexity is high, and the random variation is large, which increases the difficulty of the forecasting work. Given the increasingly complex traffic situation, the development of more accurate short-term traffic flow forecasting to achieve accurate real-time traffic information determination is still an urgent problem to be solved.

Domestic and foreign scholars have carried out extensive research on short-term traffic flow prediction. According to the research content, they can be divided into three

categories: statistical theory and methods, superficial machine learning models, and deep learning models. The statistical-theory-based methods are mainly represented by the auto-regressive integrated moving average (ARIMA) model and its improved model. Han et al. [1] proposed a real-time adaptive prediction of short-term flow based on the ARIMA model. The parameters are estimated by the forgetting factor recursion, and then the linear minimum variance is used to predict the traffic flow. Williams et al. [2] and Shi, G. et al. [3] proposed a seasonal ARIMA according to the periodicity of traffic flow to improve prediction accuracy. This kind of parametric model, based on statistical methods, has a simple structure, fast working speed, and good interpretation ability and can achieve better results with fewer data. The disadvantages are that the form is relatively rigid, it is difficult to explain complex problems, and the deviation is relatively large. To compensate for the above shortcomings, shallow machine learning models have been proposed based on the analysis of randomness and nonlinear characteristics of traffic flow, such as support vector regression (SVR) and the K-nearest neighbor model (KNN) (Liu et al., 2018 [4]; Liu et al., 2021 [5]). Wu et al. [6] applied the support vector machine model to travel time prediction and proved the feasibility of its application. Habtemichael et al. [7] proposed a data-driven approach for short-term traffic forecasting. Such models are flexible and powerful and apply well to solving complex traffic time series problems. However, the models are too complex, more parameters need to be learned, and the training time is too long.

With the rapid rise in deep learning in recent years, a growing number of researchers have devoted themselves to the task of combining traffic flow prediction with deep learning to obtain a new model. In 2006, Hinton et al. [8] proposed a deep belief network (DBN), which was the first time that a deep learning concept was proposed, and Huang et al. [9] proposed the use of a multitask regression layer to supervise the DBN. For the first time, deep learning was applied to short-term traffic flow prediction, and Lv et al. [10] first proposed the use of a stacked auto-encoder (SAE) model for traffic flow prediction, which used a stacked auto-encoder to learn the traffic flow Cho, K. et al. [11] used two recurrent neural networks (RNN) as a framework and performed joint training through encoding and decoding, which improved the performance of the machine translation system. Pascanu et al. [12] pointed out that RNN had advantages in time series, but in long time series, problems of gradient disappearance and explosion problems occur, resulting in inaccurate prediction accuracy. To solve this problem, Ma et al. [13] proposed a long-short-term memory model (Long Short-Term Memory Network, LSTM), which is an improved model of RNN used to obtain the long-term sequence dependence of traffic speed and effectively avoid the problem of gradient disappearance and explosion; Wang et al. [14] proposed a deep BiLSTM model, which captures the deep features of the traffic flow through forward and backward propagation and further improves the prediction accuracy.

The models proposed above are all for forecasting time series. The complex spatial characteristics of traffic flow are also important indicators that affect the prediction of traffic flow. Therefore, to achieve an ideal condition, more researchers have devoted themselves to the research of the deep mixture model of the spatio-temporal characteristics of traffic flow. Cheng et al. [15] proposed a combination of ARIMA and wavelet neural network (WNN) to divide the traffic flow sequence into linear and nonlinear structural parts, using ARIMA to predict the linear part and WNN to predict the nonlinear residual part. To improve the accuracy of the prediction results, Luo et al. [16] used a CNN in the underlying network to learn traffic flow features for the spatio-temporal characteristics of traffic flow and used the extracted results as input to SVR for prediction. Luo, X. et al. [17] proposed combining KNN and LSTM. The spatial stations were screened by the KNN model, and the screened station data were then used as input to the LSTM model for prediction. Song, Xiang et al. [18] proposed a combination model based on a group data processing algorithm and time series forecasting. By dividing the data into three groups, each group is divided according to weekdays and weekends, holidays and non-holidays, seasons, etc. The group data were substituted into the time prediction model for prediction, and good results were obtained. However, this paper is more about the operation in the data processing part,

ignoring the spatial relationship between the detection points. Ma et al. [19], through periodic component processing, grouped data into 288 sample ranges per day. The data in each period were integrated into a matrix, which was input into the CNN model to extract spatial features, and finally passed into the LSTM model for fusion through the full connection layer. Qu et al. [20] proposed mining the potential spatial relations of context according to the supervised learning algorithm and transmitting the data to the deep neural network for training. Based on the literature [18], Ma et al. [21] used a genetic algorithm to sort the input context factors and then convert their importance into weights. A group of historical data was selected as the input prediction algorithm according to the similarity of weights for prediction.

Inspired by the application of the above models in predicting traffic flow and using analysis and summary, starting from the spatio-temporal characteristics of traffic flow, this study proposes a KNN-BILSTM combination model using KNN for spatial feature selection and adjusting the encoding vector by the attention mechanism. This study was divided into two parts. First, the KNN algorithm was used to screen the spatial site correlations of the selected data. By setting different thresholds, the selected data under different K values were used as input data of BILSTM for prediction, and the final prediction was produced. The result with the smallest error was considered the final result. Compared with other existing models, the model proposed in this study has better prediction accuracy and is a reasonable model for predicting traffic flow.

2. KNN-BILSTM Model

2.1. KNN Algorithm

As a very mature theoretical method, the original KNN model is used to solve classification and regression problems and has been used in many studies. In [22], the intuitive advantage of the KNN model is that there are no assumptions on data distribution, high flexibility, and easy operation. However, the original KNN model has a lag in the time series and cannot fully consider the correlation of nearby road segments, resulting in a deviation in the prediction accuracy. The current KNN considers the spatio-temporal correlation between roads and augments the original KNN model using a Gaussian weighting method, as found in Cai et al. [23]. The core idea of KNN is to calculate the distance between different eigenvalues, find the point closest to the target point, and obtain the result by the weighted average. The distance metric is used to measure the state vector and the current state vector in the historical database. Several methods are commonly used to measure the distance, including the Chebyshev distance, the Huffman distance, and the Euclidean distance formula. Luo X. et al. [17] proposed that since the Euclidean distance can be used to calculate not only arbitrary spatial distances but also small time series, this study uses the Euclidean distance formula to select the correlation of traffic flow.

$$d_i = \sqrt{\sum_k (x_o(k) - x_i(k))^2} \quad (1)$$

Among the variables, $x_o(k)$ is the traffic flow detected by the target detection section at time k and $x_i(k)$ is the traffic flow of the i-th detection station in the road network at time k.

2.2. BILSTM Algorithm

Traffic flow data were used as time-series data, and information between adjacent nodes could be transferred to each other. In recurrent neural networks (RNNs), the output of neurons in the previous moment can be used as the input of neurons in the next moment so that the RNN has a memory function in short-term time series prediction. However, RNNs cannot preserve long-term historical data and have poor performance in long-term memory, with vanishing and exploding gradients. To overcome this shortcoming, a modified model of the RNN, the LSTM, is proposed, whose purpose is to allow memory cells to determine when to forget certain information in order to determine the optimal delay for time-series

problems. A typical LSTM consists of an input layer, a recursive hidden layer with memory blocks as the basic unit, and an output layer. The memory block contains self-connected memory cells with stored temporal states and three adaptive multiplication gate cells (input, output, and forgetting gates) that control the flow of information in the block. Three additional gates provide a sequential simulation of the write, read, and reset operations on the block. The multiplication gates can learn to open and close. Therefore, [24] pointed out that LSTM memory cells can store and access information over a long period of time, mitigating the vanishing gradient problem. This was calculated as follows:

$$\text{Forgotten Gate}: \ f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{2}$$

$$\text{Input gate}: \ i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{3}$$

$$\widetilde{C} = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{4}$$

$$\text{Unit status}: \ C_t = f_t C_{t-1} + i_t \widetilde{C}_t \tag{5}$$

$$\text{Output gate}: \ o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{6}$$

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

The one-way LSTM model derives information for a given time in the future only from historical data. In traffic flow prediction, time series prediction not only refers to the historical information of the current moment but also takes into account the information of the future moment to achieve an accurate prediction of the long-term traffic flow. Therefore, based on the LSTM model, the BILSTM model was proposed to improve the prediction accuracy. The BILSTM model uses a double-layer LSTM model unit structure and simultaneously transmits information through forward and backward propagation. Forward propagation is calculated from time $1 - t$, and the information output at each time is retained. Backward propagation is calculated from time $t - 1$, and the information output at each time is retained. Finally, the output state variables of the two results are concatenated as the final result. Table 1 lists the structural parameters of the model network.

$$\overrightarrow{h}_t = LSTM(x_t, \overrightarrow{h}_{t-1}) \tag{8}$$

$$\overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t-1}) \tag{9}$$

$$yt = \left(W_{\underset{hy}{\rightarrow}} \overrightarrow{h}_t + W_{\underset{hy}{\leftarrow}} \overleftarrow{h}_t + b_y\right) g \tag{10}$$

Table 1. Network structure parameters.

Parameter	Meaning
σ	Sigmoid activation
tanh	Activation function
W	Weight matrix
b	Bias term
f_t	The forget gate outputs at time t
h_{t-1}	Input from the previous unit
$\widetilde{C}$	Memory cell output
C_{t-1}	The state of the previous memory cell

BILSTM Model Prediction Process

When the BILSTM model predicts, as shown in Figure 1, it first assumes that its input samples are x_{t-1}, x_t, x_{t+1}, and then calculates through two separate LSTM units:

(a) First, input the samples x_{t-1}, x_t, x_{t+1} into the LSTM unit in sequence according to the forward calculation, and obtain the forward state output $\left\{h_{t-1}^f, h_t^f, h_{t+1}^f\right\}$;

(b) For the LSTM unit of backward calculation, the sample order is input according to x_{t+1}, x_t, x_{t-1}, and the output $\left\{h_{t+1}^b, h_t^b, h_{t-1}^b\right\}$ of the backward state is obtained;

(c) Splice two sets of output state variables of the same latitude to obtain $\left\{\left\{h_{t-1}^f, h_{t+1}^b\right\}, \left\{h_t^f, h_t^b\right\}, \left\{h_{t+1}^f, h_{t-1}^b\right\}\right\}$.

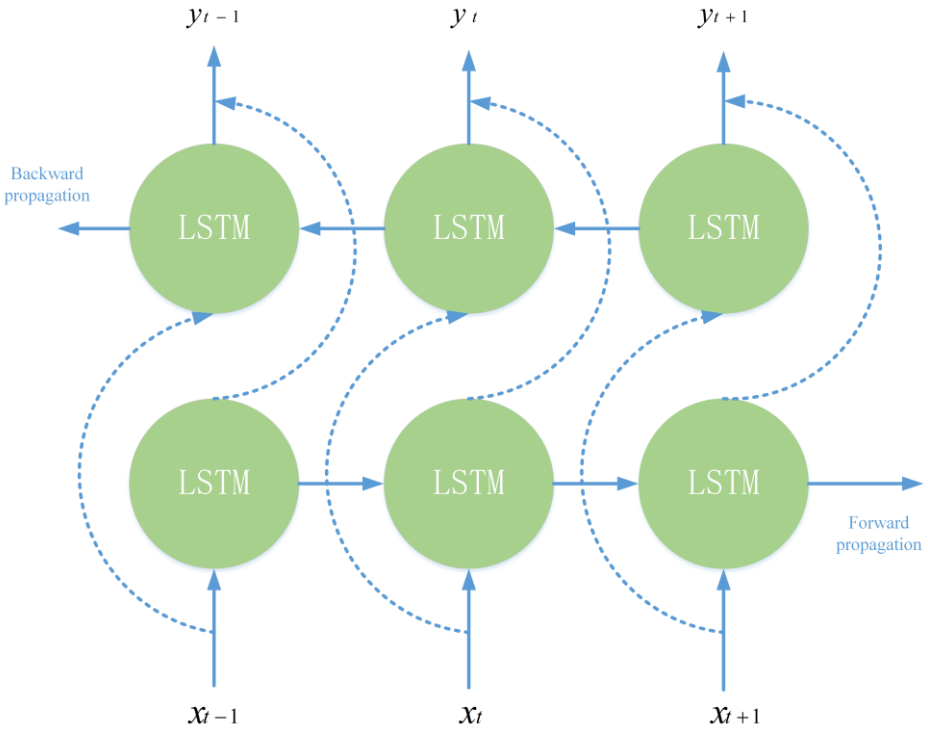


Figure 1. BILSTM structure diagram.

2.3. KNN-BILSTM Algorithm

Assuming that there are N-section detection points in the selected highway network structure, we take one of the detection points as the target detection point and form all traffic flow data into spatio-temporal correlation matrix data. The matrix of traffic flow data on the d-th day is defined as T^d:

$$T^d = \begin{bmatrix} S_1^d \\ S_2^d \\ \vdots \\ S_o^d \\ S_i^d \\ \vdots \\ S_{N-1}^d \\ S_N^d \end{bmatrix} = \begin{bmatrix} s_1^d(1) & s_1^d(2) & \cdots & s_1^d(j) & \cdots & s_1^d(k) \\ s_2^d(1) & s_2^d(2) & \cdots & s_2^d(j) & \cdots & s_2^d(k) \\ \vdots & \vdots & & \vdots & & \vdots \\ s_o^d(1) & s_o^d(2) & \cdots & s_o^d(j) & \cdots & s_o^d(k) \\ s_i^d(1) & s_i^d(2) & \cdots & s_i^d(j) & \cdots & s_i^d(k) \\ \vdots & \vdots & & \vdots & & \vdots \\ s_{N-1}^d(1) & s_{N-1}^d(2) & \cdots & s_{N-1}^d(j) & \cdots & s_{N-1}^d(k) \\ s_N^d(1) & s_N^d(2) & \cdots & s_N^d(j) & \cdots & s_N^d(k) \end{bmatrix} \quad (11)$$

The above formula, S_i^d represents the traffic flow data of the i-th day detection point on the d-th day, S_o^d is the traffic flow data of the selected target detection point on the d-th day, the corresponding $s_i^d(j)$ represents the traffic flow data detected at the i-th detection point at time j on the d-th day, $s_o^d(j)$ is the traffic flow of the target detection point at the j time of the d-th day, k is the number of traffic flow samples of the time series in one day, and N is the number of selected monitoring points.

After determining the road network data, the KNN algorithm was used for spatial correlation screening. By setting different spatial thresholds, the filtered data samples were

input to the BILSTM model for training, and the optimal result was selected as the final test result. The specific steps of the algorithm are described as follows:

(1) The traffic flow data selected in the road network structure was averaged using all the traffic flow samples for the total number of days, and the average value of the different detection points was obtained. The formula used is as follows:

$$\overline{S_i} = \frac{1}{n}\sum_{d=1}^{n} S_i^d \tag{12}$$

(2) We determined the Euclidean distance between the average sample flow of the target detection point and that of the other detection points.
(3) Multidimensional spatial data search with KDTree in the NS method was used to build a spatial matrix.
(4) The KNN model was built using the above method, and the correlation calculation was performed.
(5) We selected the MIV evaluation index to evaluate and sort the correlation size.
(6) If K = 1, 2, 3 ... n, we selected from large to small according to the correlation and input the matrix corresponding to the sample data corresponding to different K values into the BILSTM model for training.
(7) The first 80% of the selected data were training data, and the rest were prediction datasets.
(8) The result predicted by the K value corresponding to the selected error was the final result.

The prediction flow chart is shown in Figure 2:

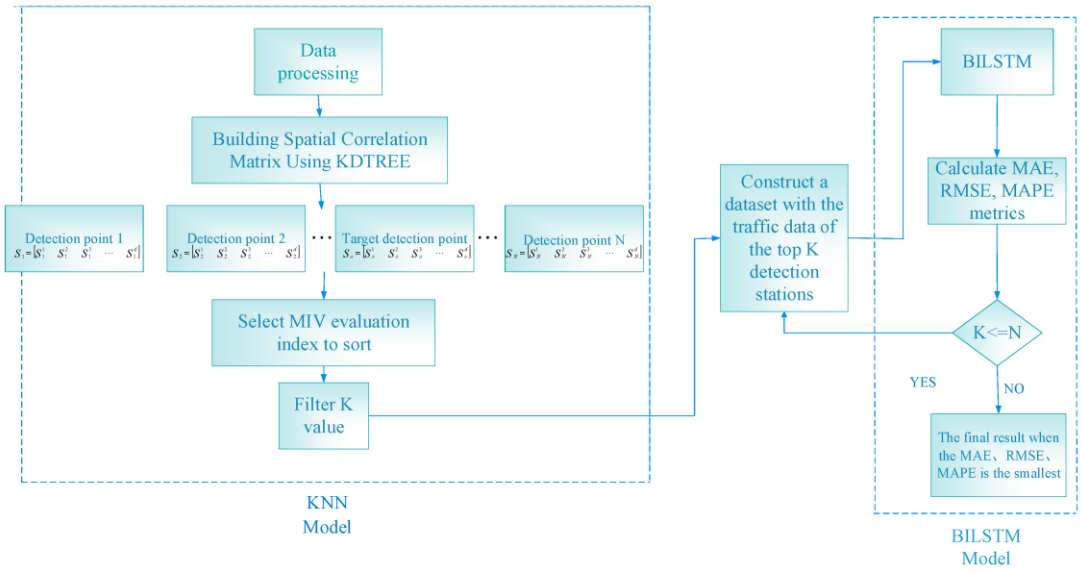


Figure 2. Traffic flow prediction flow chart.

3. Experimental Data

To verify the availability of the KNN-BILSTM model proposed in this paper, we experimentally tested the algorithm using traffic flow data from British motorways. The dataset includes real-time recordings of time, flow, and speed from all detection points on all motorways in the UK. Fifteen detection points in the interchange area of the M25 and the M23 near WARWICKWOLD in London, England, were selected as the research object. The corresponding positions of each detection point are shown in Figure 3. The selected detection point is the traffic flow in the same direction, and since the traffic flow on and off the overpass is affected, 3310 B is used as the target detection point. Due to the periodicity of the traffic flow data, the traffic flow data from 1 January 2021 to 30 April 2021

was selected as the experimental data, the time interval was recorded every 15 min, and the number of samples of the current traffic flow sequence was $4 \times 24 \times 31$. Eighty percent of all the data was used as training data, and the rest of the data was used as test data for the experiments.

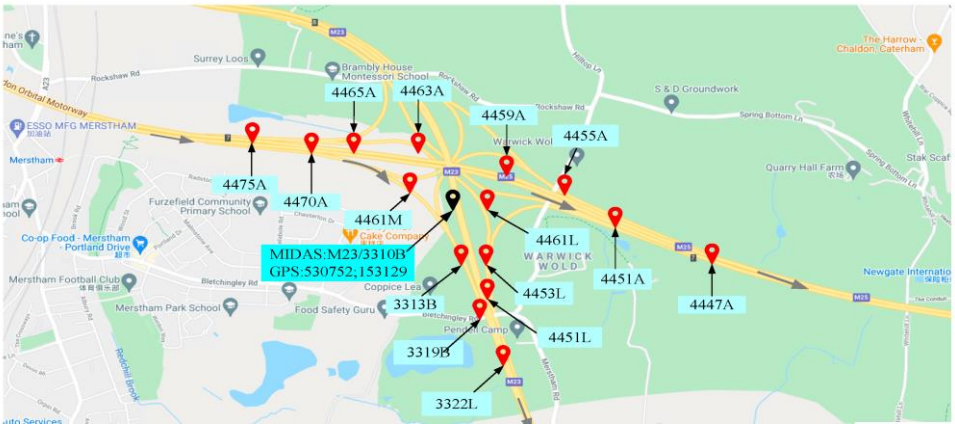


Figure 3. Distribution and ID of test points in the experimental road network.

To evaluate the performance of the model in predicting traffic flow, the mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE) were used as prediction evaluation indicators, and the definition formula was as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{13}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{14}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - y_i}{y_i}\right| \tag{15}$$

where n is the number of samples, y_i is the real value of the data, and $\hat{y}_i$ is the predicted value of the data sample.

4. Analysis of Experimental Results

4.1. Experimental Platform

The computer configuration used in this study was 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30 GHz, GPU: NVIDIA GeForce RTX 3060 Laptop GPU, Compute Capability: ′8.6′, RAM: 16 G. The running environment in this model was: MATLAB R2021b.

4.2. Experimental Environment Parameter Settings

In several experiments, we set the BILSTM model to three layers: the input layer, the hidden layer, and the output layer, where the fully connected layer was used to adjust the dimension of the output layer. Parameter setting: The Adam function was optimized, the number of iterations of model training was 400, the batch size was 24, the initial learning rate was 0.005, the period in which the learning rate decreased was set to 100, and the learning rate was adjusted by setting the weight factor to 0.8. After adjustment, the learning rate was equal to the current learning rate $\times$ weight factor, and the dropout value was set to 0.2.

4.3. Experimental Results

Using the prediction of the experimental process in Figure 2, the predicted and actual values of the obtained traffic flow are shown in Figure 4, where the predicted data flow is

represented by a solid red line, and the actual flow is represented by a dotted black line. It can be seen in (a) in Figure 4 that the predicted traffic flow is in good agreement with the actual traffic flow. It can be seen in the two sub-graphs (b) and (c) that the errors in training and prediction are relatively small. The training error is basically kept between -5 and 5 vehicles, and the test error is basically kept between -4 and 4, which shows that the KNN-BILSTM model proposed in this study is suitable for traffic flow forecasting.

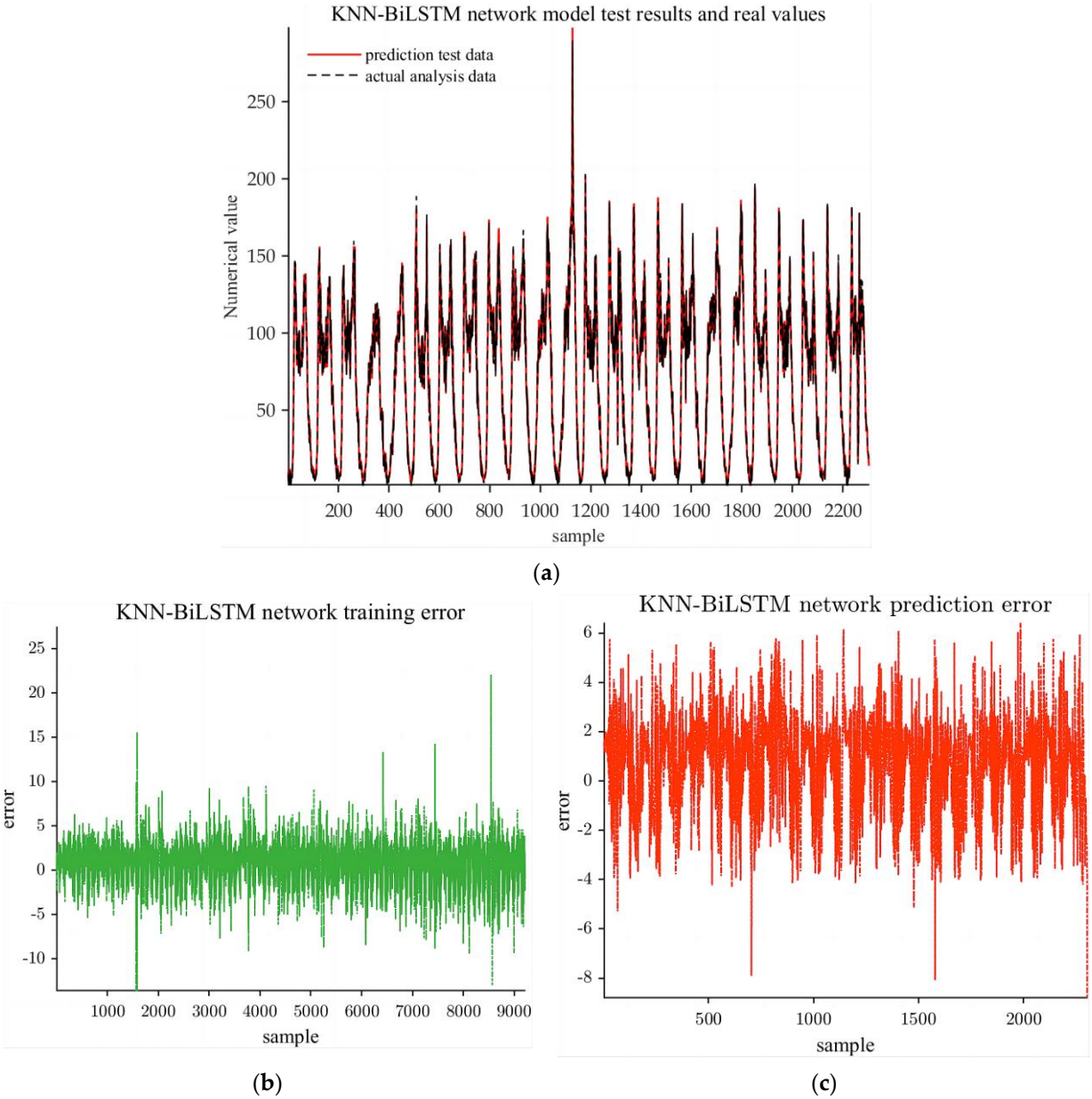


Figure 4. Prediction results and true values. (**a**) The picture shows the comparison between the predicted flow and the real flow. (**b**,**c**) are two subgraphs: (**b**) is the training error map, and (**c**) is the prediction error map.

To see the details of the prediction fit plot in Figure 4 more clearly, we screened the sample size of the model to 96 samples for prediction. The results are shown in Figure 5. In the figure, the *y*-axis represents the number of vehicles in each timestamp, and the *x*-axis represents the data output sample of the test set (the number of vehicles detected at each timestamp as a sample).

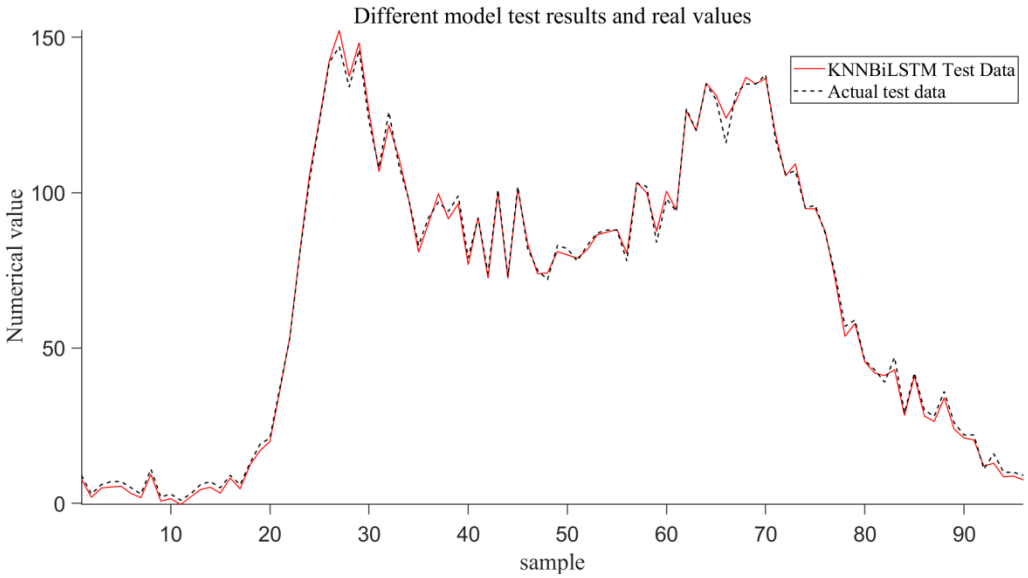


Figure 5. Screening one-day sample size prediction result.

In order to verify the influence of different K values in the KNN algorithm on traffic flow prediction, this paper selects the spatial correlation; that is, different values of K are selected for prediction. The results show that different K values have a significant impact on prediction performance. As can be seen in Figure 6, when k = 8, that is, the number of relevant detection points is 8, the effect is the best, and the loss value is the lowest.

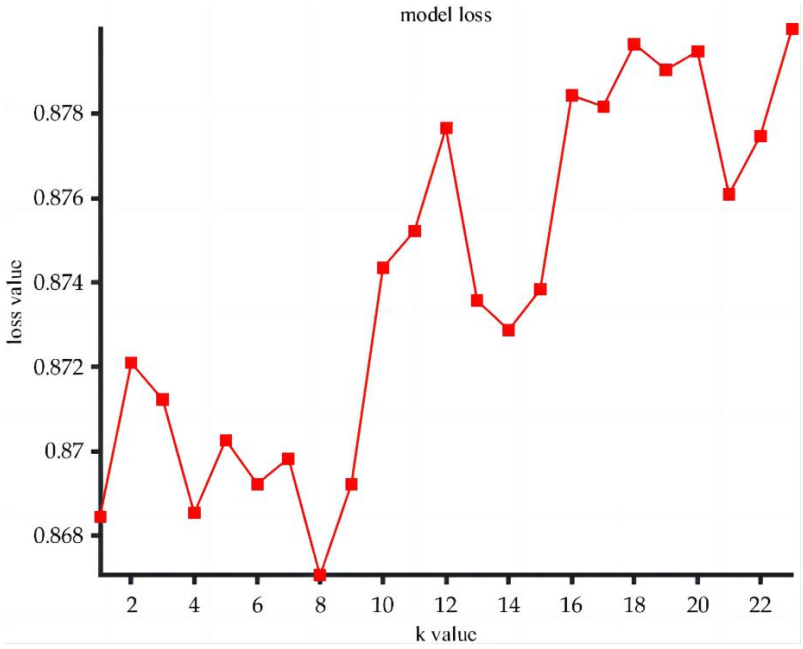


Figure 6. Loss values corresponding to different k values.

In order to more intuitively observe the data correlation between all detection points, we selected a total of 192 sample data from the first two days of all testing points for observation. It can be observed from bottom to top in Figure 7 that the amplitude of eight line segments, including target points, is small, indicating that the data structure correlation is strong. The amplitude of the above seven line segments is increasing, which can further prove the conclusion obtained when k = 8. At this time, the corresponding detection point numbers are 4465A, 4455A, 4459A, 4451L, 4470A, 4461L, 4453L, and 4475A, and they are all

located in the upstream and downstream sections of the target detection point. The spatial correlation between the selected detection points and the target detection points at different K values is shown in Figures 6 and 8 and Table 2.

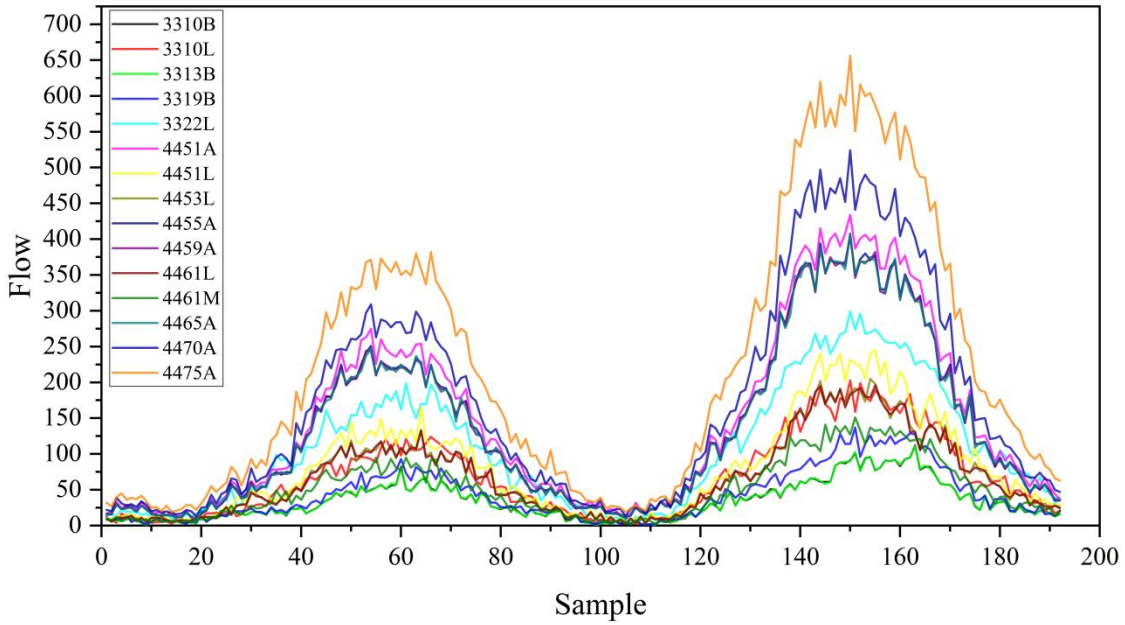


Figure 7. Detection point data distribution.

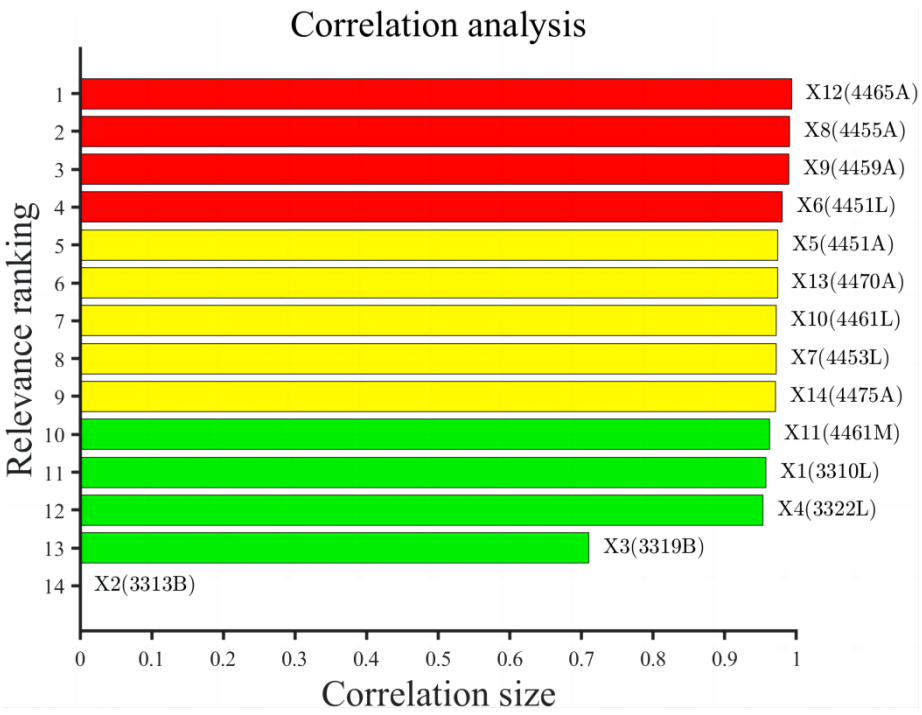


Figure 8. Correlation analysis between detection points and target points. Red indicates the four points that are highly correlated and similar to the target detection point; yellow is next; green is the last.

Table 2. Detection point numbers under different combinations.

K Value	Detection Point Number
Target point	3310B
K = 1	3310B, 4465A
K = 2	3310B, 4465A, 4455A
K = 3	3310B, 4465A, 4455A, 4459A
K = 4	3310B, 4465A, 4455A, 4459A, 4451L
⋮	
K = 14	3310B, 4465A, 4455A, 4459A, 4451L, 4451A, 4470A, 4461L, 4453L, 4475A, 4461M, 3310L, 3322L, 3319B, 3313B

To further evaluate the effectiveness of the KNN-BILSTM model, this paper selects MAE, RMSE, and MAPE as the evaluation indicators and selects the SVR, LSTM, GRU, KNN-LSTM, CNN-LSTM, and AT-convLSTM models as the comparison models for verification. By conducting 30 separate experiments on the 5 groups of models and encapsulating the experimental results, the average value of the evaluation indicators corresponding to the 30 groups of models is used for verification.

The current literature on traffic flow can be roughly divided into two categories. The first is traffic flow prediction based on Euclidean space data, and the second is traffic flow prediction based on non-European space, such as convolutional neural networks and variants. In traffic flow prediction based on Euclidean space, KNN, and CNN, there are many studies on classic models, such as LSTM. With the popularization of attention mechanisms, more researchers have added attention mechanisms based on CNN, KNN, LSTM, and other models to improve and achieve better results. Therefore, we first selected the representative model of AT-ConvLSTM for comparison. Among them, for the AT-convLSTM model, two groups of experiments were carried out in this paper. One group was used to add a new dataset. The new dataset had the same structure as [25]'s dataset, and both belong to the public data of PEMS Expressway District 10. The new dataset was substituted into the KNN-BILSTM model for training without considering the periodic influence of multiple components; the MAE and RMSE index values were 4.0442 and 4.6293, respectively, which were better than 7.14 and 9.69 found in the literature (Zheng et al., 2020). The error diagram is shown in Figure 9 shown. Another set of experiments was performed by putting the dataset in this paper into the AT-convLSTM model. The training results are shown in Figure 10. It can be seen that the training effect of the AT-convLSTM model for the dataset in this paper is relatively general.

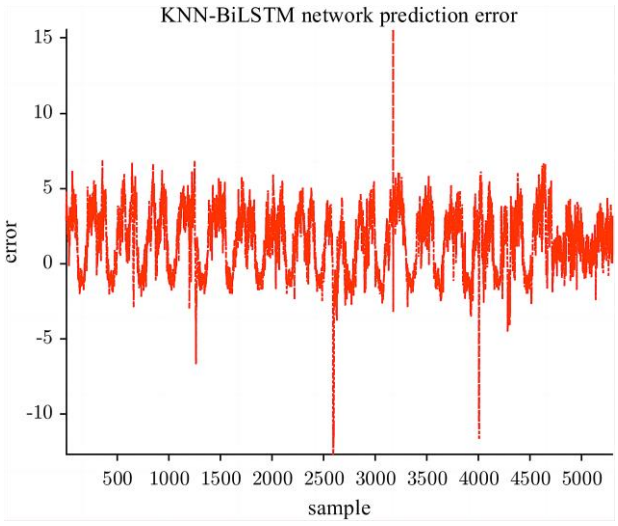


Figure 9. New data prediction error.

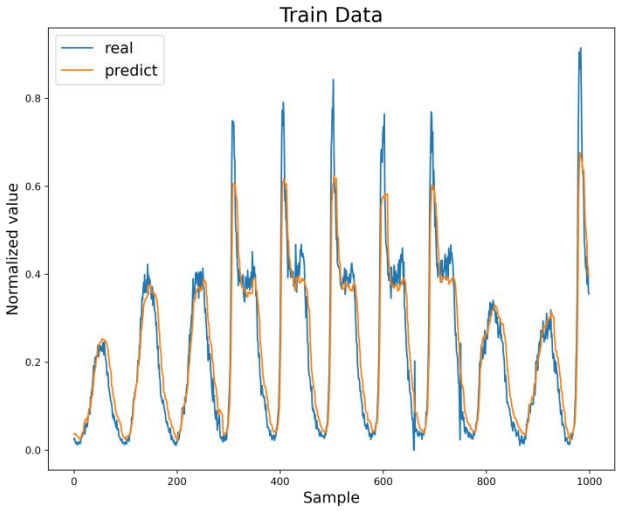
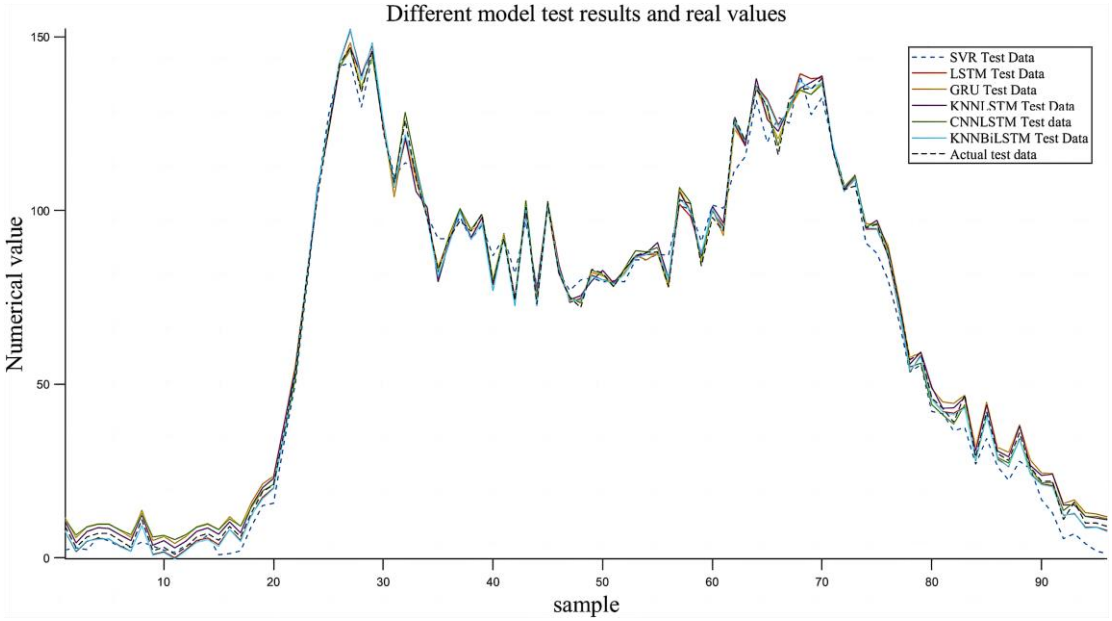


Figure 10. AT-ConvLSTM training results.

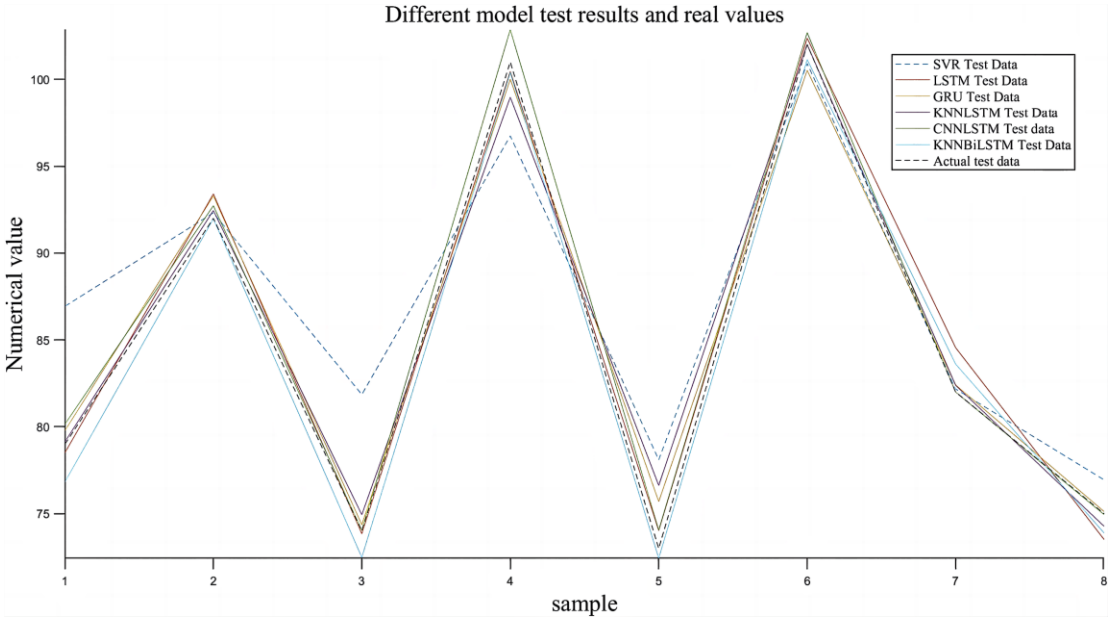
The results of other comparable models are shown in Figure 11 and Table 3. It can be seen that the model used in this study has the best predictive effect, and the SVR model has the worst fitting effect. The MAE index was compared with CNN-LSTM, KNN-LSTM, LSTM, GRU, SVR, and other models with improved performance of 13%, 22%, 19%, 18%, and 77%, respectively. In terms of running time, the SVR model takes the least amount of time, but its effect is the worst, indicating the shortcomings of SVR in predicting and processing high-dimensional data. Comparing CNN-LSTM, KNN-LSTM, and the model in this paper, although KNN-LSTM is superior to the model in this paper in running time and has fast convergence speed because it can only carry out one-way propagation in LSTM model training, the result is 22% different from the BILSTM model with 2-way propagation. Among them, the prediction effect of the LSTM model and the GRU model is close, and the GRU model is superior to the LSTM model in convergence speed and has fast convergence speed; however, the prediction performance of this model is still 18%, so the superior training speed can not make up for the performance gap, so the overall performance of this model is better than the listed comparison model.

Table 3. Error table of prediction results of the different models.

	KNN-BILSTM	**CNN-LSTM**	**LSTM**	**GRU**	**KNN-LSTM**	**SVR**
MAE	1.4677	1.6911	1.8035	1.8005	1.8916	6.4724
RMSE	1.7996	2.2335	2.3088	2.2129	2.3853	11.9097
MAPE	0.0343	0.0467	0.0795	0.0823	0.0992	0.1737



(**a**)



(**b**)

Figure 11. Comparison of errors in prediction results of the different models. (**a**) Used a day's data sample for prediction. (**b**) Used a total of 8 samples of 40–47 from Figure (**a**) for amplification.

5. Conclusions

In this study, the KNN-BILSTM combined model is used to predict the traffic flow of expressway sections. Considering the spatio-temporal characteristics of traffic flow data, the KNN model is selected to check the spatial correlation of the detection points of the road section. By selecting different K values, the correlation between the detection points of the target road section is sorted, and the two-way propagation characteristics of the BILSTM model are used to fully consider the change law of forward and backward traffic flow. When analyzing the time series of road section detection points, the effect is better than the LSTM model that only considers forward propagation. Sequence training and obtained prediction results are compared with SVR, LSTM, GRU, KNN-LSTM, and CNN-LSTM models. The experimental results prove that the model proposed in this study

has good prediction accuracy. In this study, by predicting the traffic flow of expressways, managers can obtain effective information in a short time and control and guide traffic conditions promptly, thereby effectively alleviating traffic congestion. As a time series prediction model, the model is also applicable to other time series traffic forecasts, such as urban taxi traffic forecasts and subway crowd traffic forecasts.

Although the prediction effect of the model proposed in this paper is good, it also has some limitations. First of all, the prediction target of this paper is only single-feature prediction, and the impact analysis of weather factors and traffic accidents is lacking. Second, the model in this paper still predicts using Euclidean spatial data, and in the actual traffic scene, non-European spatial data prediction may be closer to life. Therefore, in future research work, we will add other influencing factors of traffic flow to predict the first problem to make the model better. Weather factors, traffic speed, road blockage, and other factors will make the prediction problem more complex and accurate. For the second problem, we will consider using a new model to predict the traffic flow of non-European spatial data, such as a graph convolution neural network. Therefore, we plan to propose a model that can accurately deal with data lost due to influencing factors.

Author Contributions: W.Z. designed the ideas, frameworks, and approach and edited the draft of the manuscript. Y.C. edited the draft of the manuscript, and W.Z. contributed to its modification. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Social Science Foundation of China (22BGL007), Fujian Zhi-lian-yun Supply Chain Technology and Economy Integration Service Platform from the Fujian Association for Science and Technology, the Fujian–Kenya Silk Road Cloud Joint R&D Center (2021D021) from the Fujian Provincial Department of Science and Technology, the Fujian Social Sciences Federation Planning Project (FJ2021Z006), the General Program of the Fujian Natural Science Foundation (2022J01941), and the 2022 Scientific and Technological Innovation Think Tank Project of the Fujian Association for Science and Technology (FJKX-2022XKB023).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The traffic flow data used to support the findings of this study are available from the corresponding author upon request.

Acknowledgments: The authors would like to thank the reviewers and editors for improving this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Han, C.; Song, S.; Wang, C.H. Real-time adaptive prediction of short-term traffic flow based on ARIMA model. *J. Syst. Simul.* **2004**, *7*, 1530–1532+1535.
2. Williams, B.M.; Hoel, L.A. Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results. *J. Transp. Eng.* **2003**, *129*, 664–672. [CrossRef]
3. Shi, G.; Guo, J.; Huang, W.; Williams, B.M. Modeling Seasonal Heteroscedasticity in Vehicular Traffic Condition Series Using a Seasonal Adjustment Approach. *J. Transp. Eng.* **2014**, *140*, 04014012. [CrossRef]
4. Liu, S.; Lin, Y.; Luo, C. A novel learning method for traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm. In Proceedings of the 2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS), Chengdu, China, 23–26 April 2021; pp. 205–210.
5. Liu, Z.; Du, W.; Yan, D.M.; Chai, G.; Guo, J.H. Short-term traffic flow forecasting based on combination of k-nearest neighbor and support vector regression. *J. Highw. Transp. Res. Dev.* **2018**, *12*, 89–96. [CrossRef]
6. Wu, C.-H.; Ho, J.-M.; Lee, D. Travel-Time Prediction With Support Vector Regression. *IEEE Trans. Intell. Transp. Syst.* **2004**, *5*, 276–281. [CrossRef]
7. Habtemichael, F.G.; Cetin, M. Short-term traffic flow rate forecasting based on identifying similar traffic patterns. *Transp. Res. Part C Emerg. Technol.* **2016**, *66*, 61–78. [CrossRef]
8. Hinton, G.E.; Osindero, S.; Teh, Y.-W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* **2006**, *18*, 1527–1554. [CrossRef] [PubMed]
9. Huang, W.; Song, G.; Hong, H.; Xie, K. Deep Architecture for Traffic Flow Prediction: Deep Belief Networks With Multitask Learning. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 2191–2201. [CrossRef]

10. Lv, Y.; Duan, Y.; Kang, W.; Li, Z.; Wang, F.-Y. Traffic Flow Prediction With Big Data: A Deep Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 865–873. [CrossRef]
11. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
12. Pascanu, R.; Mikolov, T.; Bengio, Y. On the difficulty of training recurrent neural networks. In Proceedings of the International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 1310–1318.
13. Ma, X.; Tao, Z.; Wang, Y.; Yu, H.; Wang, Y. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transp. Res. Part C Emerg. Technol.* **2015**, *54*, 187–197. [CrossRef]
14. Wang, J.; Hu, F.; Li, L. Deep bi-directional long short-term memory model for short-term traffic flow prediction. In *International Conference on Neural Information Processing*; Springer: Cham, Switzerland, 2017; pp. 306–316.
15. Cheng, Y.; Cheng, X.; Tan, M. Traffic Flow Prediction Based on Combination Model of ARIMA and Wavelet Neural Network. *Comput. Technol. Dev.* **2017**, *27*, 169–172.
16. Luo, W.; Dong, B.; Wang, Z. Short-term traffic flow prediction based on CNN-SVR hybrid deep learning model. *J. Transp. Syst. Eng. Inf. Technol.* **2017**, *17*, 68–74.
17. Luo, X.; Li, D.; Yang, Y.; Zhang, S. Spatio-temporal traffic flow prediction with KNN and LSTM. *J. Adv. Transp.* **2019**, 2019. [CrossRef]
18. Song, X.; Li, W.; Ma, D.; Wang, D.; Qu, L.; Wang, Y. A Match-Then-Predict Method for Daily Traffic Flow Forecasting Based on Group Method of Data Handling. *Comput. Civ. Infrastruct. Eng.* **2018**, *33*, 982–998. [CrossRef]
19. Ma, D.; Song, X.; Li, P. Daily Traffic Flow Forecasting Through a Contextual Convolutional Recurrent Neural Network Modeling Inter- and Intra-Day Traffic Patterns. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 2627–2636. [CrossRef]
20. Qu, L.; Li, W.; Li, W.; Ma, D.; Wang, Y. Daily long-term traffic flow forecasting based on a deep neural network. *Expert Syst. Appl.* **2018**, *121*, 304–312. [CrossRef]
21. Ma, D.; Ben Song, X.; Zhu, J.; Ma, W. Input data selection for daily traffic flow forecasting through contextual mining and intra-day pattern recognition. *Expert Syst. Appl.* **2021**, *176*, 114902. [CrossRef]
22. Clark, S. Traffic prediction using multivariate non-parametric regression. *J. Transp. Eng.* **2003**, *129*, 161–168. [CrossRef]
23. Cai, P.; Wang, Y.; Lu, G.; Chen, P.; Ding, C.; Sun, J. A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. *Transp. Res. Part C Emerg. Technol.* **2016**, *62*, 21–34. [CrossRef]
24. Kang, D.; Lv, Y.; Chen, Y. Short-term traffic flow prediction with LSTM recurrent neural network. In Proceedings of the 2017 IEEE 20th international conference on intelligent transportation systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 1–6.
25. Zheng, H.; Lin, F.; Feng, X.; Chen, Y. A Hybrid Deep Learning Model With Attention-Based Conv-LSTM Networks for Short-Term Traffic Flow Prediction. *IEEE Trans. Intell. Transp. Syst.* **2020**, *22*, 6910–6920. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.