Short-Term Traffic Volume Forecast Method Based On CNN-LSTM-At

Xing Xu
ZheJiang University of Science and Technology

Chengxing Liu (liuchstar@163.com)
ZheJiang University of Science and Technology

Yun Zhao
ZheJiang University of Science and Technology

Xuyang Yu
ZheJiang University of Science and Technology

Xiang Wu
ZheJiang University of Science and Technology

Research Article

Keywords: Forecast, Convolutional Neural Network, neural network, spatial matrix

Posted Date: September 16th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-870743/v1

Short-term Traffic Volume Forecast method based on CNN-LSTM-At

Xing Xu[1],Chengxing Liu[1,*],Yun Zhao[2,+],Xuyang Yu[1,+],,Xiang Wu[1,+]

[1]ZheJiang University of Science and Technology, School of Mechanical and Energy Engineering, HangZhou ,310023, ZheJiang, China
[2]ZheJiang University of Science and Technology, School of Information and Electronic Engineering, HangZhou ,310023, ZheJiang, China
*corresponding author : liuchstar@163.com
[+]these authors contributed equally to this work

ABSTRACT

In order to tackle existing traffic flow prediction problem, a Traffic Volume Forecast Model based on deep learning is designed. The model implements Convolutional Neural Network (CNN) to extract spatial matrix information, uses long and short-term neural network (LSTM) for sequence prediction, appends attention mechanism to time step on LSTM, and assigns weights to different time steps. By implementing model verification on the Chengdu taxi dataset, dividing data into various categories, cross validating different categories of data, and comparing the model with other models, it is concluded that the CNN-LSTM-At network model proposed in this article has higher accuracy compared with traditional network model.

Introduction

Traffic volume forecast has long been a question of great interest as urbanization intensifies. Obtaining urban road traffic flow in real time and predicting the future traffic flow accurately enhance the efficiency of traffic control and guidance, and improve the quality of urban road traffic[1].The Internet of Vehicle has been developing rapidly, such as vehicle to vehicle communication (V2V), vehicle to infrastructure communication (V2I)[2] vehicle to everything(V2X), etc. By integration of global positioning system (the GPS), vehicle-to-vehicle communication technology, the Internet of Vehicle is able to access to a large amount of information and data. Along with the large amounts of data, new methods of traffic volume prediction are emerging. Traffic volume prediction plays a pivotal role in alleviating urban traffic pressure and improving people's living standards. For example, accurate prediction on traffic volume at various intersections allows advanced guidance on drivers to choose a suitable road and avoid congestion during peak traffic in the city. Prediction on traffic volume at different time and location allows taxi companies allocate vehicles in advance. And by predicting pedestrian volume at different time and location, bike-sharing service providers is able to allocate bikes in advance.

There are various methods of traffic volume forecast. Traffic volume forecast is a classic problem in terms of time and space. In recent years, the development of technology has highlighted many methods, including Genetic Algorithms, Neural Networks, Data Mining, Gray theory, and Support Vector Machine, etc. For example, Li Huichao et al. proposed a short-term traffic flow forecast method based on genetic algorithm optimized wavelet neural network[3]. Luo Xianglong et al. proposed a model based on the combination of KNN and Long-Short-term Memory Network model[4]. Gao Zhongwen et al. analyzed the traffic flow based on big data[5]. Jing Huixin et al. combined the Gray Model with Neural Network to predict traffic flow[6]. Liu Kun et al. used particle swarm to optimize the Support Vector Machine and

provided new insights into traffic flow prediction[7]. Especially with the development of hardware, Neural Network play an increasingly vital role in the field of traffic flow forecasting, in which in depth mathematical model is not required. Instead, convolution is used to predict traffic volume, which is accurate and efficient. For instance, Niu K proposed a method combining CNN and LSTM to predict traffic flow[8] . Zhang Junbo proposed Spatio-Temporal Residual Network based on deep learning[9] to predict traffic volume. Huaxiu Yao proposed a new Spatio-Temporal Dynamic Network (STDN), introducing a gating mechanism to learn the dynamic similarity between locations[10].Ling Zhao combined convolution (GCN) and Gate Iteration Unit (GRU)[11]. As time goes on, the theory of artificial intelligence has become more mature, and framework tools for deep learning have emerged. Deep learning provides useful accounts of traffic flow prediction.

There are four types of traffic flow prediction models. The first is the growth rate model, simple to calculate but exposed to a large forecast error. The second is regression analysis, which can solve multiple problems, but cannot wave non-quantitative index factors into traffic generation. The third is the category generation rate model, which allows multiple traffic generations, but not suitable when influencing factors are too many or deviate dramatically. The fourth type is the time series model, which requires certain degree of mathematical and computational background. Based on the division of time series model, we propose a spatial-sequence-based CNN-LSTM-ATTENTION model (hereinafter referred to as CNN-LSTM-At).

This article mainly tackles two problems:

(1): Visualize the traffic flow in a certain area thoroughly using matrix thermodynamic diagram to facilitate transplantation to various mobile terminal devices.

(2): Propose the neural network model of CNN-LSTM-At. The network model can extract the temporal and spatial relationships of the matrix and make real-time predictions. If enough data is available, the model can be updated in real time to predict traffic flow with high accuracy.

Model

Problem Description

Traffic flow involves both time and location. Data on taxi suggests traffic flow of a city. This article uses OD matrix combined with time and location to carry out research on taxi volume of a city.

An area of the city is divided into a * b squares, a total of S (S = a * b) grids, S is in the range of [1,2,3,4 ..... S], the value of each square is the traffic volume in this area. The entire time period is divided into n number successive equal time intervals t. The horizon of the time period depends on the size of the data set, which can be one month or several months. The interval of t determines the size of n. For example, when the data interval is 15 minutes, and the entire time period is one month (30 days) , n = 4*24*30 = 2880. When the selected data set is not robust enough, we replace the dataset, select the most suitable time interval, or enhance the data.

The traffic flow in a time interval is defined as the number of taxi orders that arrive in or depart from the area in this time interval. The matrix of taxi orders arriving in and departing from the area is represented by O(t) and D(t) respectively. O_t (Origin) and D_t (Destination) refer to the total number of orders arriving in and departing from this area respectively. S_b^a (a >=0, b>= 0) refers to the taxi orders arriving in the area within a certain time interval of each small grid, then.

$$O(t) = \begin{pmatrix} S_0^0 & S_1^0 & S_2^0 & \cdots & S_b^0 \\ S_0^1 & S_1^1 & S_2^1 & \cdots & S_b^1 \\ S_0^2 & S_1^2 & S_2^2 & \cdots & S_b^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_0^a & S_1^a & S_2^a & \cdots & S_b^a \end{pmatrix}, a>=0, b>=0$$

In terms of location, O(t) and D(t)both belong to the a * b square. In terms of time, O (t-1), O(t), O(t+1) are continuous. In the process of feature extraction, it will affect each other's performance. This applies to D_t as well, the article predicts O(t) when making the prediction. In special cases, O(t) and D(t) are combined, which makes the depth of the matrix to 2, and connects to a matrix in the shape of [a, b, 2]. Then it only needs to change the output shape and make prediction at the same time.

Model Description

By dividing the traffic flow into a matrix, it is found that there is spatial correlation within the matrix, and at the same time, there is temporal correlation between adjacent matrices. The article is carried out by firstly using the method of sliding window to divide the data set into several continuous matrices, extracting the spatial information of each continuous matrix, and then applying LSTM to predict the time dimension. When a string of time series is input into LSTM, the output is a three-dimensional time series, which serves as the feature of each time node. It is expected that, in training the network, the network model pays attention to the significant time steps, and increases the weight of time steps in output. Therefore, this article appends attention mechanism into the output of LSTM, then connects the network into Fully Connected Layer, and finally generates a prediction result matrix through reverse network.

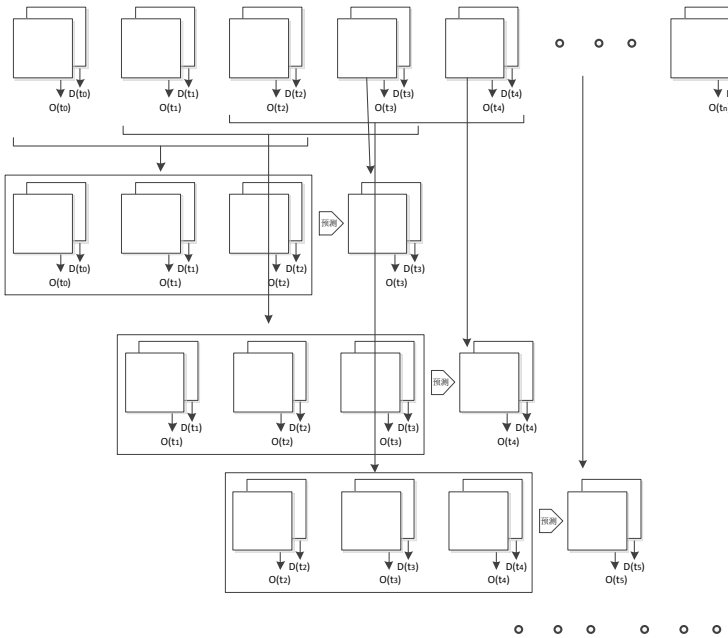


Figure 1. Data structure diagram

The division of traffic flow according to the above definition is essentially a multi-dimensional matrix. It is beneficial to use the convolutional neural network to extract feature information in the matrix. This research uses the method of sliding window to generate the data set, packs each continuous matrix, and enters the convolutional layer. The convolutional Layer consists of 64 of 2*2 two-dimensional convolution kernel, and uses relu activation function. The output of width and height after padding should be consistent with the input.

MaxPolling is utilized in pooling layers. A 2*2 two-dimensional convolution kernel is used to extract the characteristics of the data and reduce the length and width of the matrix to half of the original. Then add the appropriate Dropout to avoid overfitting. Finally, the output data of the continuous matrix is flattened to generate a series of three-dimensional data and input into the LSTM layer.

The LSTM layer has three gates: input gate, output gate and forget gate. The calculation formula for each gate is:

$$f_t = \sigma\left(W_f \bullet \left[h_{t-1},\ x_t\right] + b_f\right)$$
$$i_t = \sigma\left(W_i \bullet \left[h_{t-1},\ x_t\right] + b_i\right)$$
$$o_t = \sigma\left(W_o \bullet \left[h_{t-1},\ x_t\right] + b_o\right)$$
$$h_t = o_t * tanh(C_t)$$
$$\widetilde{C_t} = tanh\left(W_c \bullet \left[h_{t-1},\ x_t\right] + b_c\right)$$
$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}$$

‘ ° ’ refers to the multiplication of the elements at the corresponding positions on the two vectors; ‘σ’ refers to the sigmoid function, which is used as the activation function; f refers to the forget gate; i refers to the input gate, o refers to the output gate; C refers Cell state; $\tilde{C}$ is the current input unit state; h is the cell output. [12]

LSTM well characterizes data with temporal and special sequence, and has a significant impact on suppressing the diffusion gradient phenomenon[13].In this paper, the previous output of CNN is regarded as the input sequence of LSTM, along with 256 neuron nodes, to make a prediction on the dimension of output space. Then wave attention mechanism into output of LSTM, find the time step that needs the most attention, and finally flatten it and generate inverse network output. Inverse network has a * b output to fully connect sample and integrated change. When making preliminary predictions, we predict O(t), and the network structure diagram is shown in Figure 2:

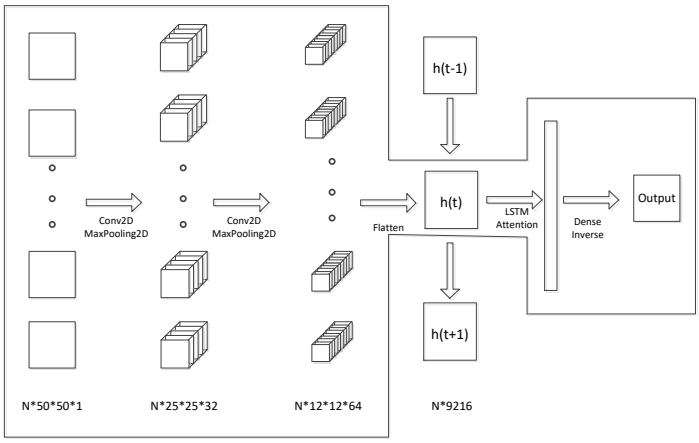


Figure 2. Overall network model

Related Work

Original Data

With the development of smart city [14], the amount of data in traffic has increased dramatically. Data of Chengdu taxi used in this paper is the product of smart city. They are retrieved from https://gaia.didichuxing.com. The main source is track and order on DiDi Express in an area of Chengdu. The collecting interval of track points is 2-4s. The track points have been tied to the road, ensuring that the data is in accordance with actual road information. Sampling Data range from 1st Nov 2016 to 30th Nov 2016. A total of 7065937 orders. The original data is as follows:

{Order number, Start billing time, End billing time, Pick up location longitude, Pick up location latitude, Drop off location longitude, Drop off location latitude}

Order number: The source data has been encrypted, desensitized and anonymized.

Start and end billing time: Unix timestamp in seconds.

Pick up location longitude/latitude: GCJ-02 coordinate system.

Drop off location longitude/latitude:

[104.043333,30.727818],[104.129076,30.726490]

[104.042102,30.652828],[104.129591,30.655191]

The approximate area of track is about 8KM*8KM, as shown in the figure 3:

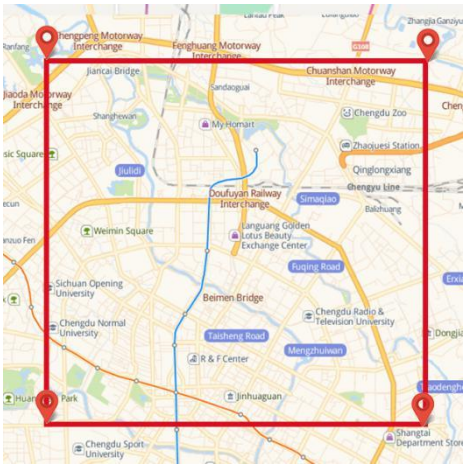


Figure 3. Range of track

Data Definition and Processing

In order to visualize data and reduce the complexity of processing large amounts of data, the 8KM*8KM track area is divided into an n * n grid map. The area of each grid refers to the latitude and longitude coordinates within a certain range[15]. The number of pick up and drop off in a grid is equivalent to the passenger flow volume within this time interval and area. Generally, there is a certain spatial correlation between an area with dense passenger flow and its neighboring area and an area with sparse passenger flow with its neighboring area. In a certain area and its neighboring areas, the probability of taxi drivers encountering passengers is high (low).

The original data set records the time billing starts and ends and the corresponding latitude and longitude coordinates for each order. Data are autocorrelated in time thus time variables are discrete in this paper. In order to separate weekends from working days, all working days (Monday to Friday) are divided into consecutive minutes.

Time flow appears to be consecutively related, take the five working days in the second full week of November as example. The five days are divided into 15- minute continuous time intervals, the flow is shown in the figure 4. Evidence revealed that the daily traffic flows are consistent within the same time frame. The traffic flow changes only before holidays and when unexpected events occur.

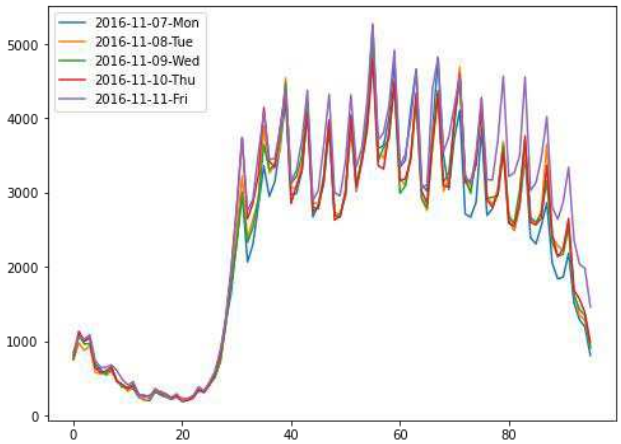
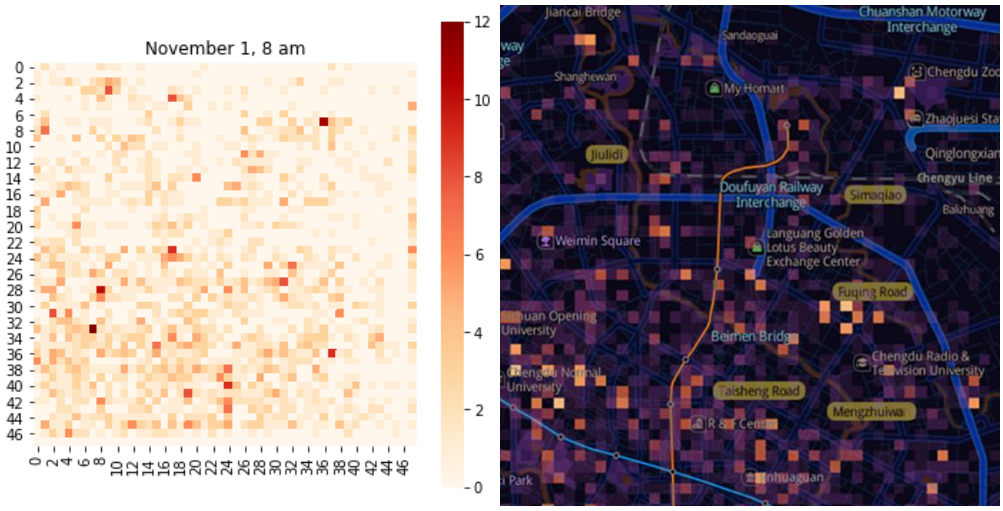


Figure 4. Graph of total traffic flow

The passenger flow volume within several minutes in the entire area forms a matrix of n*n. Then visualize the matrix in a thermo dynamic diagram[16]. E.g., The matrix thermodynamic diagram and urban visualization thermodynamic of the matrix of passenger flow volume d from 08：00 to 08：15 on 1st Nov 2016 is shown in figure 5. Since this paper predicts short-term passenger flow, the time interval is either 15 minutes or 30 minutes. The grids are divided into either 32*32 or 48*48. At last, input all the data in the model and start training.



(1) (2)

Figure 5. Flow matrix heat map and city heat map

Experimental and Predicted Results Analysis

Experimental Conditions and Parameters

The language used in this paper is python 3.7. The hardware platform is Intel (R) Core (TM) i9-10900X CPU@3.70GHZ, 64GB RAM, 1TB solid state hard drive, NVIDIA GeForce RTX 2080Ti graphics card. And Tensor Flow 2.0.0 is used to train and test the model.

The main parameters of the model are shown in the table 1.

Training Parameters	Corresponding Method and Value

Time Interval	15 Min		30 Min	
Training Set(group)	1804		892	
Enhanced Training Set(group)	4872		2408	
Validation Set(group)	540		268	
Test Set(group)	849		417	
Mesh Size	32	48	32	48
Epoch	250			
Parameter Optimization Method	Adaptive Moment Estimation			
Loss	Root-Mean-Square Error			
Prevent Overfitting	Dropout			

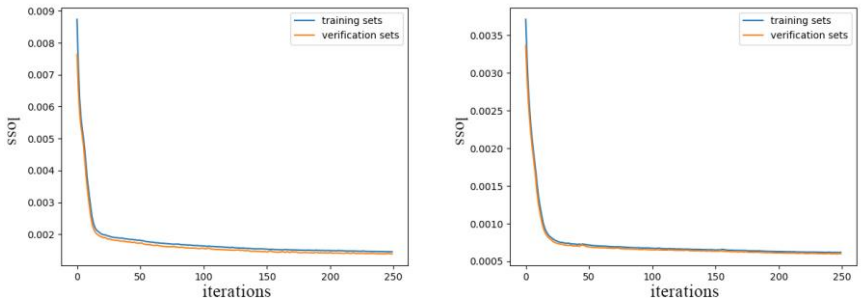
Table 1. Parameter training graph

The data are expended to 3 times as the original data set by method of mirror and reverse the original data set. The time interval is either 15 minutes or 30 minutes, and 5 time-steps are used to predict the next time step. The training period is Mondays to Fridays of the previous 4 weeks in Nov 2016 and the predicting period is Monday to Wednesday of the fifth week. Add attention mechanism in the last output direction, thus the dimension weights are not the same for time steps in different significance level. Find the attention probability imported in each time step then calculate the average.

In order to analyze the impact of different time intervals and the number of grid divisions on the prediction accuracy, the following experiments are carried out: ①The time interval is 15 minutes, and the number of grid divisions is 32*32 ; ②The time interval is 15 minutes, and the number of grids is 48* 48 ; ③The time interval is 30 minutes, and the number of grids is 32*32 ; ④The time interval is 30 minutes, and the number of grids is 48*48 .

Model Training Convergence Speed

1) The time interval is 15 minutes, and the number of grids is 32*32 and 48*48 respectively. Input the model to make predictions.The model performance is shown in the figure.



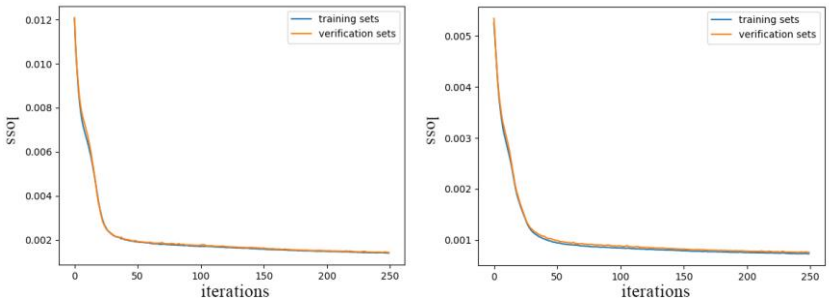
(1) (2)

Figure 6. Model training performance graph with an interval of 15 minutes

Figure(1) is the performance graph with 15 minutes time interval and 32 * 32 grid division. Figure(2) is the performance graph with 15 minutes time interval and 48 * 48 grids. The network is trained 250 times each time. The figures reveal that overfitting occurred in both training experiments, and the model has been trained to its optimal performance.The model begins to converge when it is trained for about 40

times. After that, the training set loss value and the lost value of validation set begin to converge smoothly,and tend to the final converge after the training reaches 200 times.

2) The time interval is 30 minutes, and the number of grids is 32*32 and 48*48 respectively. Input the model to make predictions. The model performance is shown in the figure 7.



(1) (2)

Figure 7. Model training performance graph with an interval of 30 minutes

The above figure(1)(2) is the model training diagram with time interval of 30 minutes and grids of 32*32 and 48*48.Unlike figure 6, the network convergence speed is significantly delayed when the time interval is 30 minutes, it just starts to converge after 50 iterations, and finally stabilizes around 230 iterations.

3)The loss value comparison between different time intervals and between different grid divisions

Table 2 shows the prediction error of different grid divisions at different time intervals after 250 iterations. It can be seen from the table that the time interval is 15 minutes, the grid is divided into 48*48, the root mean square error of the next time step is the smallest, the root mean square error of the training set is 0.000625, and the root mean square error of the verification set is 0.000606.

Then the interval is 30 minutes and the grids are divided into 48 * 48, the root mean square error increase by 0.000109 and 0.000156 respectively. But the data training volume of the former is twice that of the latter.

Number of Iterations	Time Intervals/Min	Meshing	Root Mean Square Error of Training Set	Root Mean Square Error of Validation set
250	15	32*32	0.001500	0.001400
250	15	48*48	0.000625	0.000606
250	30	32*32	0.001400	0.001400
250	30	48*48	0.000734	0.000762

Table 2. Comparison of Root Mean Square Error of Training Set and Validation Set with Different Time Intervals and Meshing Methods

Attention Weight

In terms of assigning attention weight to each time step, firstly obtain the attention probability of each time step with imported dimension, then calculate the average.

For a model with time interval of 15 minutes and grid of 48*48, the average attention distribution weight for five time-steps is shown in the figure 8 below. The attention weight is higher on the first time-step. The distribution of weights mainly depends on the extraction of data set features and model training.

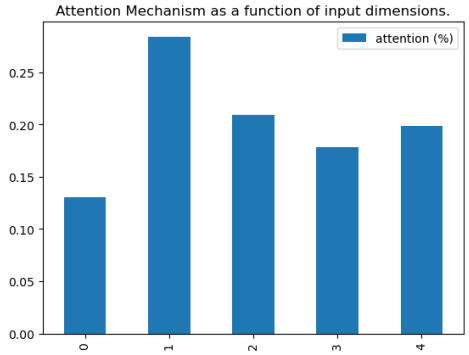
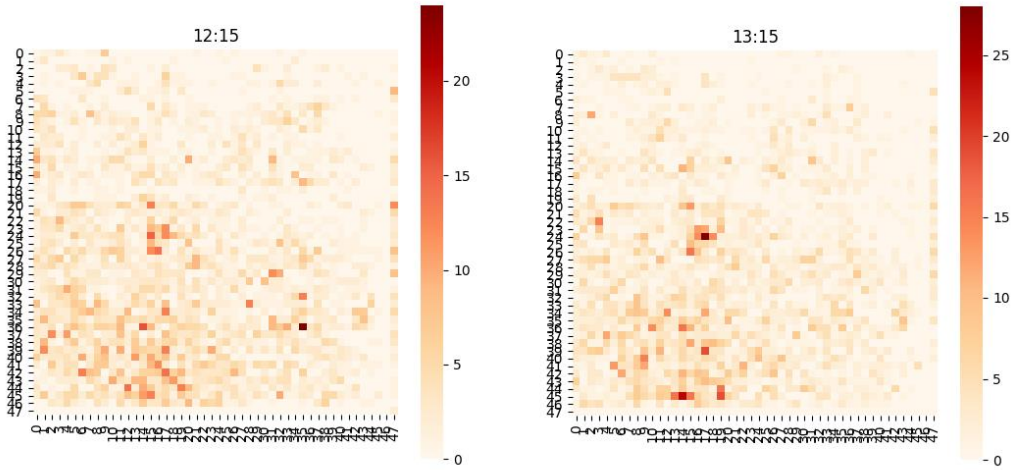


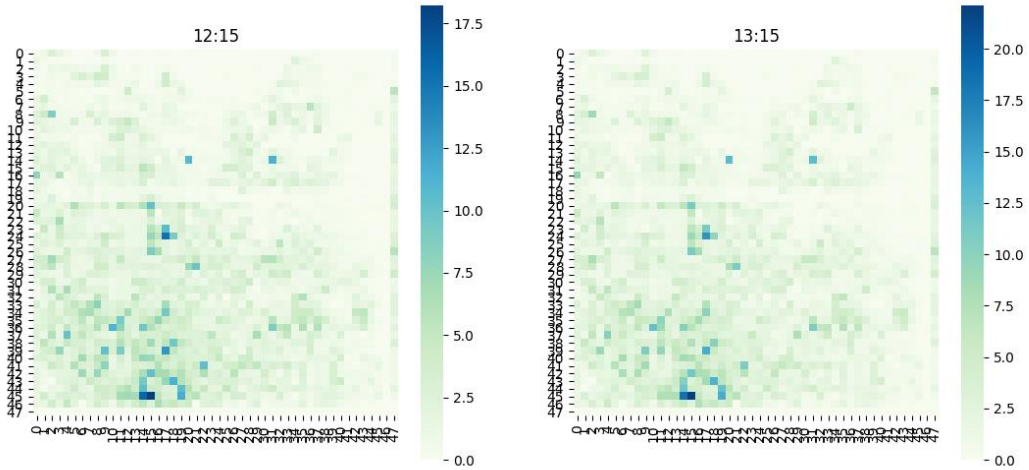
Figure 8. Attention weights at different time steps

Data Visualization

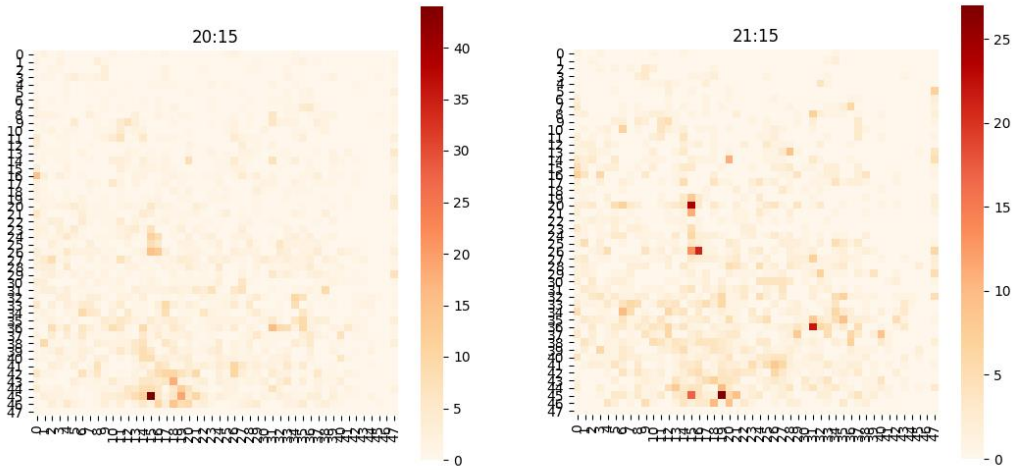
This paper randomly selects four time points on the first day of the last week of 2016 in Chengdu for testing, and then visualizes them. The interval between two time points is one hour. The model selected in the visualization process of this article is a model with time interval of 15 minutes and a grid of 48*48. As the figure 9 suggests, the prediction result of this model is highly consistent with the actual situation on the bottom of the edge. At the same time, this model also maintains good stability for sudden increases of passenger flow. When the flow of people decreases, such as at night, the model can predict the approximate results. But the model's ability to extract detailed features needs to be improved.



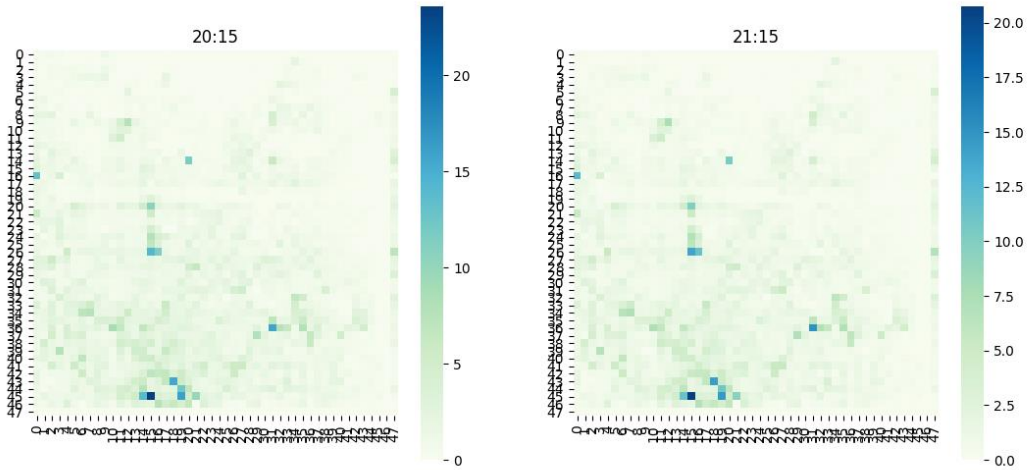
(1) Actual value



(2) Predicted value



(3) Actual value



(4) Predicted value

Figure 9. Predicted and actual values at different time periods

Image information is evaluated by information entropy in this paper.Image entropy refers to the average

number of bits of the image gray level set, unit bit/pixel, and it also describes the average amount of information of the image source. For discrete two-dimensional images, the formula for calculating information entropy is:[17]

$$m_k = \frac{1}{N_k \times M_k} \sum_{j=1}^{N_k} \sum_{i=1}^{M_k} \log_2 |C_k(i,j)|$$

$$e_k = \sum_{j=1}^{N_k} \sum_{i=1}^{M_k} p\,[C_k(i,j)] \ln p\,[C_k(i,j)]$$

Among them, p_i is the probability of each gray level. Entropy refers to the degree of chaos in the system. The entropy of a well-focused image is greater than that of an unfocused image, so entropy can be used as a focus evaluation criterion. The greater the entropy is, the clearer the image is.

The entropy of an image is a statistical form of features. It reflects the average amount of information in the image and represents the aggregation characteristics of the image grayscale distribution, but it cannot reflect the spatial characteristics of the image grayscale distribution. In order to characterize this spatial feature, the two-dimensional entropy of the image can be composed of a feature quantity that can reflect the spatial feature of the grayscale distribution based on the one-dimensional entropy.

Image entropy is used to measure the similarity between the predicted value image and the actual value image in different time periods. The closer the entropy of the two images is, the more similar they are. In the process of generating images, in order to make the graphics of predicted results look neat, the local maximum value was suppressed but not at the cost of affecting the experimental results. The maximum value of the two different images in the same time period after processing will be suppressed to be same, that is, the color depth standards of the two images are the same. The image information entropy after processing is as table 3:

Periods of Time	Original Image Entropy	Predict Image Entropy	Difference Value
12：15	3.3341	4.7933	1.4592
13：15	3.2989	4.6085	1.3096
20：15	2.6348	4.0010	1.3662
21：15	2.5661	3.8870	1.3209

Table 3. Comparison of original image entropy value and predicted image entropy value in different time periods

It can be seen from the table that the entropy difference between the predicted image and the circle image is between 1.3 and 1.5, indicating that for each time point, the predicted result is stable, the model is relatively robust, and the robustness is strong.

Error Evaluation

In order to evaluate the performance of the test results, this paper uses MAE to evaluate four time periods. The data are retrieved from four time points on the first day of the last week in 2016 of Chengdu. The data interval is 15 minutes, and the grid is divided into 48*48 squares. MAE is defined as follows:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|(y_i - \hat{y}_i)|$$

y_i is the actual value of the local area at the time point. $\hat{y}_i$ is the predicted result. And m refers to the number of local areas, which is 48 * 48 = 2304.

This article uses Multi-CNN, that is, multi-step CNN, ConvLSTM, and Multi-LSTM as the comparison of this model. It can be seen that this model performs well in most time points and evaluation indicators.The model comparison results are shown in table 4.

Models	12:15	13:15	20:15	21:15
Multi-CNN	0.0216	0.0201	0.0162	0.0149
ConvLSTM	0.0246	0.0218	0.0170	0.0152
Multi-LSTM	0.0343	0.0335	0.0331	0.0294
CNN-LSTM-At	**0.0201**	**0.0195**	**0.0158**	**0.0140**

Table 4. Comparison of MAE values of different models

Conclusions

This paper proposes a neural network model of CNN-LSTM-At on the base of the prediction of the passenger flow of taxis in Chengdu. CNN is used to extract local features of the traffic matrix, LSTM is used in the time dimension, and the attention mechanism is used to control the weight of the time dimension. Due to the limited training data, the related network of this model needs to be improved, but the corresponding results can be predicted based on the existing data. The method proposed in this paper can be deployed in the cloud, and the model can be trained in real time to ensure the timeliness of the model when the data is updated timely. The method in this paper aims to explore the relationship between time and space of traffic flow and provide insights to the field of intelligent transportation.

The method in this paper can be enhanced, such as changing the attention mechanism to artificially control the attention weight based on the data set in the case of multitime step input. The network structure can also be improved to better extract the edge features of the space and improve accuracy.

References

1. Mei, D. et al. A short-term traffic flow prediction model based on EMD and GPSO-SVM. *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference*.(2017) .
2. Zhang, D.G,. Yang, P. & Zhang, J. A traffic flow prediction method for Internet of Vehicles based on quantum particle swarm optimization strategy. *Computer Science*. **47(S2),** 327-333 (2020).
3. Li, H.C., Li, H & Zhang, B. Short-term traffic flow forecast based on genetic algorithm optimized wavelet neural network. *Computer Applications and Software.* **35(07),** 148-152 (2018).
4. Luo, X.L., Li, D.Y. & Yang, Short-term traffic flow prediction based on KNN-LSTM. *Journal of Beijing University of Technology*. **44(12),** 1521-1527 (2018).
5. Gao, Z.W., Niu, Z. Analysis of traffic flow pattern based on big data. *Journal of Harbin University of Science and Technology.* **23(06),** 124-127 (2018).
6. Jing, H.X., Qian, W. & Che, K. Short-term traffic flow forecast based on gray ELM neural network. *Journal of Henan University of Technology (Natural Science Edition).* **38(02),** 97-102 (2019).
7. Liu, K. et al. Particle swarm optimization optimization support vector machine network traffic

chaos prediction. *Modern Electronic Technology*. **42(02),** 120-123 (2019).
8. Niu, K., Cheng, C. & Chang, J. Real-Time Taxi-Passenger prediction with L-CNN. *IEEE Transactions on Vehicular Technology.* **68(5),** 4122-4129 (2019).
9. Zhang, J., Zheng, Y. & Qi, D. Predicting citywide crowd flows using deep Spatio-Temporal Residual Networks. *Artificial Intelligence.* (2017).
10. Yao, H., Tang, X. Revisiting Spatial-Temporal similarity: A deep learning framework for traffic prediction. *Proceedings of the AAAI Conference on Artificial Intelligence.* **33,** 5668-5675 (2019).
11. Zhao, L., Song, Y. & Zhang, C. T-GCN: a temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems.* **PP(99),** 1-11 (2019).
12. Yan, Z., Yu, C.Z. Short-term traffic flow prediction method based on CNN+LSTM. *Computer Engineering and Design.* **40(09),** 2620-2624+2659 (2019).
13. Pavlos, T., Liu, X.L. & Lin, Z. Fault detection by an ensemble framework of Extreme Gradient Boosting (XGBoost) in the operation of offshore wind turbines. *Renewable Energy.* **179,** 945-962(2021).
14. Wang, J.Y., Li, C. & X, Z. A survey of data-centric smart city research. *Computer Research and Development.* **1(02),** 239-259 (2014).
15. Zhao, H., Zhang, Y.Q. & Fan, Q. Geographic information data coordinate conversion based on grid model. *Surveying and Spatial Geographic Information.* **3(10),** 201-203+207+211 (2020).
16. Yan, W.D., Liu, P. & He, X.Z. Research on calibration of traffic generation model based on thermodynamic diagram. *Jiangsu Urban Planning.* **08,** 24-26+47 (2019).
17. Ma, Y.D., Dai R.L. & Li, L. An automatic image segmentation method based on pulse coupled neural network and image entropy. *Journal of Communications.* **01,** 46-5 (2002).

Acknowledgements

This work was supported by the National Key Research and Development Program of China (Grant No. 2019YFE0126100), the Key Research and Development Program of Zhejiang Province of China (Grant No. 2019C54005).

Author contributions

Xing. X,. Liu, C.X. wrote the paper, and The rest process the data.

Additional information

This paper is completed by the cooperation of the authors, and there is no conflict of interest between them.