Received January 30, 2019, accepted February 16, 2019, date of publication February 28, 2019, date of current version March 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2902185

A Novel Spatio-Temporal Model for City-Scale Traffic Speed Prediction

KUN NIU, HUIYANG ZHANG, TONG ZHOU, CHENG CHENG, AND CHAO WANG
School of Software Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Kun Niu (niukun@bupt.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFE0204500, in part by the National Natural Science Foundation of China under Grant 61671081, in part by the Beijing Natural Science Foundation under Grant 4172042, and in part by the Fundamental Research Funds for the Central Universities.

ABSTRACT City-scale traffic speed prediction provides significant data foundation for the intelligent transportation system, which enriches commuters with up-to-date information about traffic condition. However, predicting on-road vehicle speed accurately is challenging, as the speed of the vehicle on the urban road is affected by various types of factors. These factors can be categorized into three main aspects, which are temporal, spatial, and other latent information. In this paper, we propose a novel spatio-temporal model named L-U-Net based on U-Net as well as long short-term memory architecture and develop an effective speed prediction model, which is capable of forecasting city-scale traffic conditions. It is worth noting that our model can avoid the high complexity and uncertainty of subjective features extraction and can be easily extended to solve other spatio-temporal prediction problems such as flow prediction. The experimental results demonstrate that the prediction model we proposed can forecast urban traffic speed effectively.

INDEX TERMS Convolutional neural network, long short-term memory neural network, spatio-temporal modeling, traffic speed prediction.

I. INTRODUCTION

The main goal of short-term traffic condition prediction is to obtain future traffic conditions based on historical traffic data, which can serve the operation of Intelligent Transportation System(ITS) better. In recent years, the amount of historical and real-time traffic data rises rapidly due to the increase of deployed sensors and the convergence of multi-source data, which provides great information infrastructure and data environment for traffic speed prediction.

Accurate prediction of traffic conditions plays an important role in many aspects. For traffic management departments, it helps them to develop dispatch plans in advance to manage the traffic flow scientifically, so as to maximize the utilization of urban transportation resources and reduce the occurrence of traffic congestion, even traffic accident. Meanwhile, for drivers, it can provide realistic estimation of journey time to reduce uncertainty. Moreover, traffic conditions forecasting is benefit for the route selection, alternative route recommendation, and expected delay assessment of navigation applications. Whether for the traffic management departments or drivers and navigation applications, it is obvious that making decisions according to the predicted future traffic conditions are more reasonable than only utilizing current traffic information [1]. However, it is very challenging to predict the traffic speed in a scale of entire city accurately, because the traffic conditions of urban roads could be affected by various factors, such as temporal, spatial, and other latent information. In light of this, we conduct as series of literature review about traffic speed prediction models. For instance, Jiang and Fei [2] proposed a data-driven vehicle speed prediction method in the context of vehicular networks, in which the average traffic speeds of road segments were predicted by neural network. Cheng *et al.* [3] proposed a big data based deep learning approach to predict the vehicle speed for an individual trip, which is capable of accurately predicting vehicle speed for both freeway and urban traffic networks. It can be perceived that the application of neural network models in the field of traffic and vehicle speed prediction tasks are of great importance.

Among different types of deep neural networks, Convolutional Neural Network (CNN), inspired by visual cortex in human's brain, has been extensively studied and applied in various artificial intelligence researches [4]. After Krizhevsky *et al.* [5] applied CNN to image recognition and achieved astonishment improvement in 2012, CNN gradually

The associate editor coordinating the review of this manuscript and approving it for publication was Zhanyu Ma.

shows its powerful ability in image and video recognition, recommender systems, image classification, natural language processing, as well as object detection [6]–[10]. Except for CNN, another important neural network model is Recurrent Neural Network (RNN), which is able to achieve high performance facing time series analysis as it can use their internal state (memory) to process sequences of inputs. As a result, it is applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition [11]. However, when the time intervals of input sequence of RNN is too long, RNN would face the problem of back-propagated error decay through memory blocks. As an improvement of RNN, Long Short-Term Memory (LSTM) Neural Network can overcome this problem, thus exhibits the superior capability for time series prediction with long temporal dependency. LSTM was proposed in [12] and show its strong ability in various fields such as speech recognition and machine translation [13]–[16]. Moreover, researchers have made achievements in multiple fields such as automatic image captioning and visual recognition by combining CNN with LSTM [17], [18]. For instance, Guggilla *et al.* [19] proposed a method to classify the claim made in online arguments using CNN and LSTM, and achieved a significant improvement on different datasets and tasks. Ullah *et al.* [20] proposed an action recognition framework by utilizing frame level deep features of the CNN and processing it through deep bidirectional LSTM.

For traffic speed prediction problems, on the one hand, future speed conditions are related to the previous speed obviously, which is a problem that LSTM is good to solve [21]; on the other hand, the speed of vehicles on each road is affected by those on surrounding roads, however, if match the speed conditions with the actual map and form a traffic speed distribution map, then we can extract the spatial features between them by automatic image feature extraction, which can be solved by CNN [22]. Therefore, combining CNN with LSTM can be used to predict city-scale traffic speed effectively.

In this paper, we propose a novel spatio-temporal model, named L-U-Net, to conduct city-scale traffic speed prediction. The model can not only capture features both on the temporal and spatial dimension without extensive features engineering, but also be extended to solve other spatio-temporal prediction problems such as flow prediction. The L-U-Net model consists of two parts, U-Net neural network [23] and LSTM neural network, and is trained with historical traffic speed data. The U-Net part extracts the spatial features related to vehicle speed from road conditions, whereas the relationship between temporal feature and vehicle speed is established by the LSTM part. After comprehensive consideration of spatio-temporal features, L-U-Net could output the prediction results of traffic speed on urban roads.

The major contributions of this paper are summarized as follows:

1. We consider and utilize the temporal and spatial features simultaneously in traffic speed prediction problems by exploiting the deep learning architecture of U-Net and LSTM without extensive features engineering.

2. We propose a novel end-to-end spatio-temporal model named L-U-Net for city-scale traffic speed prediction. The model applies Encoder-decoder architecture, and is capable of extracting advanced semantic information effectively. Moreover, the model can be easily extended to solve other spatio-temporal prediction problems such as flow prediction.

The rest of this paper is organized as follows: Section II presents an overview of the related work. Section III presents an introduction of preliminaries. The proposed model is detailed explained in Section IV. The experimental process and parameter setting are introduced in Section V. Experimental results and contrastive analysis with other prediction models are discussed in Section VI. Section VII concludes the paper with future research directions.

II. RELATED WORK

Over the last decade, the classic topic of traffic and vehicle speed prediction has drawn widespread concerns, existing approaches can be mainly categorized into two types: parametric approach and non-parametric approach [2]. Generally, parametric approaches rely on predetermined models based on priori theoretical assumptions, where model parameters are calculated and calibrated with empirical data. As a result, the accuracy of parametric approach heavily relies on massive yet representative training data. The typical models applied parametric approach include Constant Speed (CS), Constant Acceleration (CA), SUMO simulator model (SUMO), and Intelligent Driver Model (IDM) [24]–[26]. Comparing with parametric approach, non-parametric approach does not require expertise and does not fix the model structures in advance, as they rely much on historical traffic speed data so that they can derive parameters from data instead. Non-parametric approaches are more effective for large-scale transportation systems or long-term vehicle speed prediction systems [24]. Common methods adopting non-parametric approach include Gaussian Mixture Regression (GMR), and Artificial Neural Network (ANN) [27], [28]. As the research object of this paper is city-scale traffic speed, the non-parametric approaches are suitable.

In terms of the traffic speed prediction, more advanced and powerful deep learning models have been applied recently [29]. Ma *el al.* [21] introduced a long short-term memory neural network, which named LSTM, into traffic prediction and demonstrated both stability and accuracy of the LSTM in terms of traffic speed prediction by using remote microwave sensor data. Park *et al.* [30] proposed a speed prediction model named NNTM-SP, which can be trained with historical traffic data and is capable of detecting traffic dynamic changes and predict speed profile precisely in the future up to 30 minutes. Recently, derived from image feature modeling [31], Ma *et al.* [22] proposed a CNN-based method that learns traffic as images and predicts large-scale traffic speed.

However, previous works mainly focused on the prediction on a road section or a small part of road network. Few works consider the transportation network as a whole and predict the traffic conditions in city scale. Moreover, the majority of these methods merely considered the temporal features, but did not consider the spatial correlations, let alone consider it from the perspective of the network. In this paper, we propose a novel model with spatio-temporal correlations to forecast the future city-scale traffic conditions for citizens.

III. PRELIMINARY

A. DATA DESCRIPTION

As the target of our prediction is a specific city, we need to select some data which can reflect the speed of vehicles on all roads across the city. Through a lot of screening, we choose the Global Positioning System(GPS) trajectory data as the original data, which were collected from Chengdu in 2014. The format of each data record is a quintuple as follows:

$$\{Id, Latitude, Longitude, Flag, Timestamp\},$$

where Id is the taxicab's identifier, Flag is the mark whether the taxicab is carrying passengers, the others represent the relevant location and time information of the GPS record. Hence, each data record shows the location and passenger state of a taxicab at a specific time.

B. DATA PROCESSING

1) MAP MESHING

We applied the map meshing method to process city map and location information (represented by latitude and longitude) for reducing complexity of trajectory tracking data, and established a city-scale road network represented by grids. Meanwhile, because similar changes of traffic condition are usually observed in adjacent areas, we choose grid structure as model input to show these existing strong spatial dependences more specific. In this process, two details should be noted. First, we need to define the boundary of the grid-based road network. In terms of the dataset generated in Chengdu, through the evaluation and calculation of the urban areas from the map, we select a rectangular region with longitude from $103.98°E$ to $104.17°E$ and latitude from $30.59°N$ to $30.73°N$ for subsequent research. Second, the choice of grid size is critical to the accuracy of prediction and data processing time. Considering the actual width of road and distance between two continuous points that taxi report, we divide the region into 200*200 grids, and the length and width of each grid correspond to 91.04 meters and 77.92 meters on the actual map approximately. Through such a setting scheme, these grids can reflect the traffic conditions of urban roads effectively.

2) VEHICLE SPEED CALCULATION

In order to predict the traffic speed, we need to examine the speed condition of each vehicle in each grid first. According to dataset descripted above, we can extract each taxi's trajectory tracking data, represented by timestamp and corresponding location. In Chengdu taxi trajectory dataset, taxis report their position every 10 seconds. For a designated taxicab ξ, we assume it has three continuously timestamp $\{t_a, t_b, t_c\}$ and corresponds to the position $\{a, b, c\}$, the latitude and longitude coordinates are $\{(lat_a, lon_a), (lat_b, lon_b), (lat_c, lon_c)\}$. Then, the approximate distance Δd_{ab} between a and b on the surface of the earth can be calculated by following formula:

$$\Delta d_{ab} = \frac{R * Pi * \arccos\theta}{180}, \tag{1}$$

where $\theta = \sin(lat_a) * \sin(lat_b) + \cos(lat_a) * \cos(lat_b) * \cos(log_a - log_b)$, R represents the average radius of the earth.

The time interval Δt_{ab} between a and b also can be easily obtained:

$$\Delta t_{ab} = t_b - t_a. \tag{2}$$

Similarly, we can get distance Δd_{bc} and time interval Δt_{bc} between b and c.

Since the time interval between two adjacent GPS data record are pretty short and the size of the entire dataset is really large, the distance between two locations can be substituted approximately by the linear distance. So, we can obtain the approximate speed when the taxicab ξ passed the location b from the following formula:

$$v_b = \frac{\Delta d_{ab} + \Delta d_{bc}}{\Delta t_{ab} + \Delta t_{bc}}. \tag{3}$$

3) LOCATION DISCRETIZATION

After map meshing, we can obtain a series of grids represented by a square matrix, and are able to know the corresponding actual region on the surface of earth respectively. In order to calculate the average vehicle speed of each grid, we need to discretize tracking data in spatial dimension. According to the latitude and longitude coordinate of each record, we can map them into a grid $p(r, c)$ and represent them in the corresponding matrix coordinates.

4) TIME DISCRETIZATION

Generally, the traffic speed characteristics inside a region have similarities over periods of time. Since time is a continuous variable, we need to discretize it. Considering that the vehicle speed characteristics have a strong randomness in a short period of time, it is meaningless to predict vehicle speed for citizens and traffic management departments in such a short time period. Therefore, we set the time interval as 1 hour to discretize data in time dimension. That is, we only predict the average speed of target hour, and map data within the time period into corresponding time lag, so that we can effectively observe the overall traffic conditions within one hour.

5) SPEED MATRIX CALCULATION

After meshing the map and discretizing the data in temporal and spatial dimension, we can generate a 200*200 matrix $M(t)$ for each time lag and obtain the average vehicle

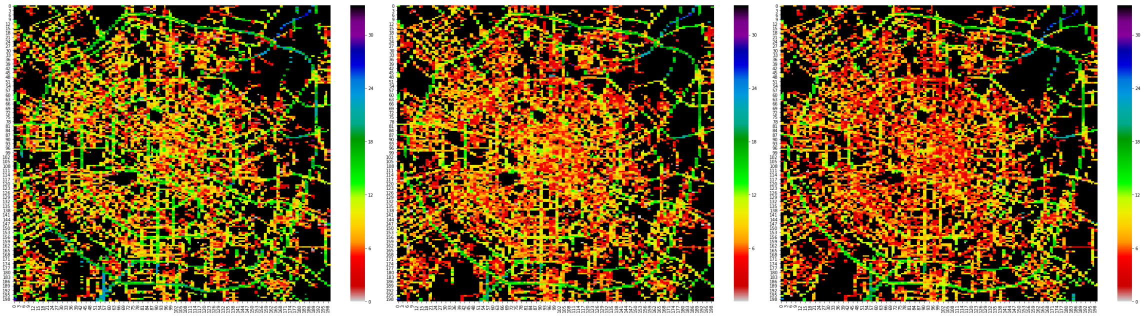


FIGURE 1. Examples of speed distribution map (on Aug 23rd, 2014, Chengdu). Black indicates non-road area, green indicates a road with a faster speed. The redder the color, the slower the speed, that is, the more crowded the road.

speed of each grid $p(r, c)$ by following formula:

$$v_p = \frac{\sum_{i=1}^{SUM\_P} w_{p_i} v_{p_i}}{\sum_{i=1}^{SUM\_P} w_{p_i}}, \tag{4}$$

where p_i indicates the i-th data record divided into grid p, w_{p_i} indicates the weight of p_i, SUM_P indicates the total number of data falling in grid $p(r, c)$. It is worth noting that we considered the practical application scenario when apply to the actual dataset. We filter out records whose flag is 0 and the vehicle speed detected equals 0 for a long time, because we regard the taxicab as being waiting for passengers instead of driving on the road.

After data pre-processing, for each target hour, we obtain a related matrix whose every element represents the average speed of a grid, which will be used as input to the subsequent speed prediction model.

IV. METHODOLOGY

In this section, we present the details of method we applied to forecast future traffic speed in a city. By analyzing the taxicabs' GPS trajectory data in Chengdu, we found that though traffic and vehicle speed conditions have some randomness in a short time period, regularities in a long time period can be mined and utilized for prediction. Moreover, traffic conditions between neighbor districts also share similarities. However, it is difficult for traditional prediction methods to consider characteristics both in spatial and temporal dimensions. Hence, we proposed a novel spatio-temporal neural network named L-U-Net, which can capture features both in the temporal and spatial dimensions for vehicle speed prediction.

A. MATRIX DEFINITION

$$M(t) = \begin{pmatrix} v_{11}^t & \cdots & v_{1n}^t \\ \vdots & \ddots & \vdots \\ v_{n1}^t & \cdots & v_{nn}^t \end{pmatrix}, \quad t \in T. \tag{5}$$

$M(t)$ represents a $n * n$ square matrix, which is the data structure of input and output of our model. It worth noting that each $M(t)$ actually corresponds to a traffic speed distribution map, whereas each element in the matrix is equivalent to one pixel in the figure. Figure 1 are examples of matrix $M(t)$, reflecting vehicle speed distribution of each grid in some specific time lag, and we can observe the traffic conditions of different roads intuitively.

Through the statistical analysis of matrix $M(t)$ with consequent multiple moments, we find that $M(t)$ has both spatial and temporal dependencies. Hence, we decide to apply U-Net architecture to extract spatial features and apply LSTM to extract temporal features. Finally, we construct an integrated model named L-U-Net.

B. U-NET

Through comparative analysis and extensive experiments, we found that by encoding the speed distribution map represented by matrixes, spatial features can be effectively extracted. Moreover, it is possible to restore original traffic conditions according to the extracted features by decoding compressed features. Therefore, we decided to apply U-Net network to build an encoding and decoding algorithm architecture for vehicle speed prediction.

U-Net architecture consists of a contracting path, following the typical architecture of a convolutional network, and an expansive path. As the expansive path is more or less symmetric to the contracting path, and yields a u-shaped architecture. Especially, every step in the expansive path consists of an up-sampling of the feature map and a concatenation with the correspondingly cropped feature map from the contracting path [23].

By applying the U-Net architecture, we can obtain useful spatial information without extensive features engineering and other external data. More importantly, we can reduce uncertainties caused by various dynamic contribution factors.

C. LSTM

We add LSTM to the model for enhancing the ability of extracting spatial features. Based on previous work of U-Net, we can simplify the prediction task as a time series forecast problem, regardless of spatial features. Through comparative analysis and extensive experiments, we found that the traffic condition, reflected by average traffic speed, of the previous period has a great impact on the subsequent traffic condition. Hence, to enhance the accuracy of the prediction model,

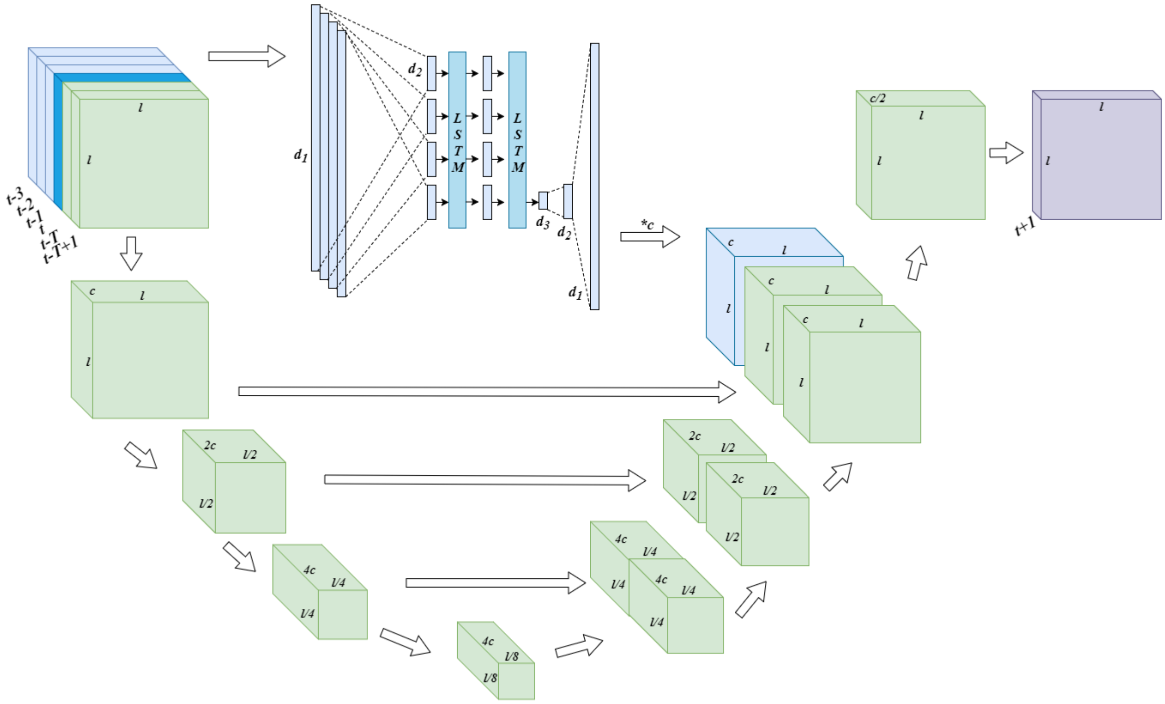


FIGURE 2. **L-U-Net architecture. The cuboids and rectangles represent multi-dimension tensors, and the lowercase letters l, c, d_1, d_2, and d_3 represent corresponding dimension respectively. Blue part depicts the time series model (LSTM), green part depicts the spatial convolution model (U-Net), and purple part is the final output of the model.**

we apply LSTM architecture to capture features on the temporal dimension.

A common LSTM architecture is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. It worth noting that LSTM architecture are well-suit to making prediction based on time series data, since there can be dependencies between important events in a time series.

By applying an LSTM architecture, we can obtain useful temporal information.

D. L-U-NET

Based on the above definition and analysis, we construct a novel model named L-U-Net for spatio-temporal prediction. The L-U-Net model is shown in Figure 2.

As it can be seen, the architecture of L-U-Net can be depicted from two aspects. Firstly, the U-Net architecture consists of a contracting path and an approximate symmetry expansive path. The contracting path follows the typical architecture of a convolutional network, and each arrow represents a compound operation that combining convolution and maxpooling. Correspondently, the expansive path is the reverse of the previous step, which are up-sampling the features map and then conduct convolution on them. Before convolution, the pre-processed speed distribution maps are need to be concatenate with the correspondingly cropped speed distribution map from the contracting path to generate a new map. In addition, unlike the original expansive path of U-Net, we joined the outputs of the LSTM model to the top layer of expansive path. The reason for choosing the top layer instead of the other layer is that the temporal features will be smoothed after convolution and cannot reflect the features of each grid accurately. Hence, this design method can reflect the local changes in urban roads maximumly.

Secondly, the LSTM model need to be used to generate partial input of the expansive path. As the dimension of the predicted data would be too high, we need to compress the vector through a fully connected network first. And then capture temporal features by the LSTM model, making up for the shortcomings of U-Net in acquisition of spatio-temporal features. After applying the LSTM model, we would decompress the output to facilitate merging with features generated by U-Net.

For predicting the average vehicle speed of time $t + 1$, the input of our model can be defined as follows: $\{M(t-3), M(t-2), M(t-1), M(t), M(t-T), M(t-T+1)\}$, where the first four items are the inputs of LSTM, the last three items are the inputs of U-Net ($M(t)$ is both used in two models), and each matrix actually corresponds to a speed distribution map. Here, time t represents a certain hour in a specific day, T represents a look-back period. Besides, what needs illustration is that T should be set as one week in theory, but we still choose it as one day considering the temporal data scarcity of our dataset.

The reason why selecting these data as input is that, in spatial, the traffic condition in a city could be extracted by inputting holistic matrix including information about each grid and its neighbor's. And in temporal, data in time slot $\{t, t\text{-}T, t\text{-}T+1\}$ could reflect long-term temporal features and data in $\{t\text{-}3, t\text{-}2, t\text{-}1, t\}$ containing short-term features. U-Net would first build basic map by utilizing data which has strong long-term connections with the target hour. And LSTM would learn short-term traffic conditions to adjust forecasting results. Therefore, there are two decoding methods, LSTM and CNN, for these two types of data respectively. In addition, we use the *Mean Square Error(MSE)* as the evaluation function, the goal of our model is to reduce *MSE* between the output matrix and the actual matrix.

V. EXPERIMENT

In this section, the proposed approach is experimentally evaluated and the results are discussed in detail. Experiments are conducted based on a series of spatio-temporal feature map represented by corresponding matrix $M(t)$.

TABLE 1. Partial raw data of Chengdu taxi trajectory dataset.

Id	Latitude	Longitude	Flag	Timestamp
1	30.667188	104.092136	0	2014/8/8 11:23:11
1	30.650990	104.000789	1	2014/8/8 07:42:33
2	30.665850	104.040357	1	2014/8/8 08:24:29
2	30.664839	104.079992	1	2014/8/8 11:52:25
...	...	...	...	...

A. DATASET

Chengdu Taxi Trajectory Dataset: It consists of three weeks' GPS trajectory data generated by 14,000 taxicabs in Chengdu province, in which the data of first two weeks are used as a training set and the last one week are used as a testing set. Moreover, about 50 million data records are generated per day, whose structure are shown in Table 1. As it can be seen, each record contains 5 categories, which are taxi id, latitude coordinate, longitude coordinate, flag (1 means there are passengers in the taxicab, 0 means no passengers), and timestamp. It is worth noting that there are nearly 14,000 taxicabs, whose speed and location distributions could reflect the overall traffic conditions of urban traffic speed sufficiently. Moreover, for each taxicab, the data are collected every 10 seconds in most cases, which are accurate enough for us to conduct experiments. The dataset can be easily access at *http://www.pkbigdata.com.*

B. ENVIRONMENT

To complete the process of data preprocessing, model construction, training, and testing, we implemented our model using *Python Pytorch 0.4.0* and experimented with the real dataset. All experiments were run on a computer with *ubuntu 18.04* operating system, *Intel i7-8700k* processor, and 64GB RAM. Besides, the GPU used to accelerate the training of neural network is the *NVIDIA GTX 1080ti* with 11GB graphic memory.

C. DATA PRE-PROCESSING

Since the GPS trajectory data uploaded by different taxicabs are almost in a disordered state, we firstly sorted the data by the taxi id and timestamp. Then we can obtain the speed of each record by using (3). Refer to (4), we can calculate the average speed of grids presented by a matrix, and obtain a visualized map of the vehicle speed distribution at specific time in Chengdu, which are shown in Figure 1. From these figures, we can observe the general conditions of speed distribution in main region and roads. After finishing data preprocessing, we use *Pytorch* framework to build the L-U-Net model for training and testing.

D. TRAINING

After building the network architecture and performing data preprocessing, we trained model with practical data. Here is the parameter setting of our model: $l = 200, c = 8, d_1 = 40000, d_2 = 2000, d_3 = 256$.

To predict the average vehicle speed of each grid on target hour, we selected the speed matrixes in corresponding time periods illustrated previously in *Section IV* as the input of L-U-Net. Hence, the training process is defined as follows:

$$M(t+1) = L - U - Net(M(t-T), M(t-T+1), M(t), \\ M(t-3), M(t-2), M(t-1)).$$

Moreover, in order to complete the training well, we choose *MSE* as the evaluation function, Adam as the optimizer (learning rate equals 0.003, *L2* regularization coefficient equals 0.001, other parameters are default parameters).

VI. RESULT ANALYSIS

We performed comparative experiments with the following models:

1. ES: Exponential Smoothing is a rule of thumb technique for smoothing time series data using the exponential window function.

2. ARIMA: ARIMA model is fitted to time series data either to better understand the data or to predict future points in the series.

3. U-Net: U-Net model only considers the feature of spatial dimension.

4. LSTM: LSTM model only considers the feature of temporal dimension.

5. L-U-Net: L-U-Net is based on LSTM and U-Net, which is able to extract both spatial and temporal features from history data to speculate situation in the next an hour.

We use the *Mean Square Error (MSE)* as the evaluation function, and evaluate the accuracy of our model and above models over different time periods. As shown in Table 2, our model has a lower *MSE* on each time period. At the same time, our model does not require a large number of feature engineering and any other external data sets, which is suitable for online learning and real-time systems.

It is worth pointing out that the dataset only contains daily GPS trajectory data from 6:00 to 23:00, and our prediction model requires the previous 4 hours' data, therefore, our model's predicting range is from 10:00 to 23:00. To compare the experimental results better, we only list the prediction

TABLE 2. MSE of prediction results under different time periods.

Model	10:00-11:00	12:00-13:00	14:00-15:00	16:00-17:00	18:00-19:00	20:00-21:00	22:00-23:00
ES 0.9	0.00570	0.00568	0.00612	0.00594	0.00608	0.00602	0.00715
ES 0.8	0.00529	0.00524	0.00565	0.00545	0.00563	0.00554	0.00663
ES 0.7	0.00494	0.00486	0.00525	0.00504	0.00526	0.00514	0.00619
ARIMA	0.00375	0.00345	0.00378	0.00354	0.00385	0.00359	0.00480
U-Net	0.00418	0.00397	0.00425	0.00400	0.00424	0.00405	0.00506
LSTM	0.00373	0.00344	0.00373	0.00346	0.00383	0.00363	0.00500
L-U-Net	**0.00365**	**0.00336**	**0.00365**	**0.00343**	**0.00380**	**0.00343**	**0.00461**

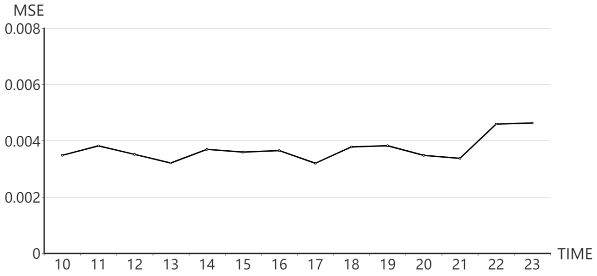


FIGURE 3. Prediction results of L-U-Net under different time periods.

results of other models in this range, although some of which can predict the traffic conditions from 6:00 to 10:00. Though the prediction results of our model are more accurate in the specified range, the limitations of prediction range are a shortcoming of our model compared to other models.

To analyze the prediction results of our model more intuitively, we draw the relevant line chart. As it can be seen in Figure 3, the forecast results are relatively worse at night because the total number of taxicabs is reduced, and the impact of accidental factors increases. However, the relative values of our predicted results are basically consistent with the original data.

VII. CONCLUSION

In this paper, we propose a novel spatio-temporal prediction model named L-U-Net by utilizing LSTM neural network combined with U-Net architecture. The model can not only capture features both in temporal and spatial dimension for traffic speed prediction, but also extract features without extensive features engineering. Our method can reduce the workload of feature engineering effectively, and we have demonstrated that it can predict traffic conditions in future well across the real dataset. It is noteworthy that our model can be easily extended to solve other spatio-temporal prediction problems. It is an attempt in the field of traffic condition prediction by combinational model inspired by method applied in image and video frame prediction task. In the future, we will continue to use L-U-Net to conduct other spatio-temporal prediction tasks. We are sure that this kind of spatio-temporal prediction will have a significant guiding role of urban transportation resources utilization in the development of smart city.

ACKNOWLEDGMENT

The authors thank all reviewers for their valuable comments and helpful suggestions.

REFERENCES

[1] Z. Ma *et al.*, "The role of data analysis in the development of intelligent energy networks," *IEEE Netw.*, vol. 31, no. 5, pp. 88–95, Sep. 2017. doi: 10.1109/MNET.2017.1600319.

[2] B. Jiang and Y. Fei, "Vehicle speed prediction by two-level data driven models in vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1793–1801, Jul. 2017.

[3] Z. Cheng, M.-Y. Chow, D. Jung, and J. Jeon, "A big data based deep learning approach for vehicle speed prediction," in *Proc. IEEE 26th Int. Symp. Ind. Electron. (ISIE)*, Jun. 2017, pp. 389–394.

[4] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[6] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, "Dynamic image networks for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3034–3042.

[7] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing [review article]," *IEEE Comput. Intell. Mag.*, vol. 13, no. 3, pp. 55–75, Aug. 2018.

[8] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Comput.*, vol. 29, no. 9, pp. 2352–2449, Sep. 2017.

[9] D. Zhang, J. Han, C. Li, J. Wang, and X. Li, "Detection of co-salient objects by looking deep and wide," *Int. J. Comput. Vis.*, vol. 120, no. 2, pp. 215–232, 2016.

[10] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.

[11] R. Bertolami, H. Bunke, S. Fernández, A. Graves, M. Liwicki, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] H. Soltau, H. Liao, and H. Sak. (2016). "Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition." [Online]. Available: https://arxiv.org/abs/1610.09975

[14] X. Li and X. Wu, "Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4520–4524.

[15] Z. Ma, H. Yu, W. Chen, and J. Guo, "Short utterance based speech language identification in intelligent vehicles with time-scale modifications and deep bottleneck features," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 121–128, Jan. 2018.

[16] Y. Wu *et al.* (2016). "Google's neural machine translation system: Bridging the gap between human and machine translation." [Online]. Available: https://arxiv.org/abs/1609.08144

[17] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3156–3164.

[18] J. Donahue *et al.*, " Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2625–2634.

[19] C. Guggilla, T. Miller, and I. Gurevych, "CNN- and LSTM-based claim classification in online user comments," in *Proc. 26th Int. Conf. Comput. Linguistics (COLING) Tech. Papers*, 2016, pp. 2740–2751.

[20] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action recognition in video sequences using deep bi-directional LSTM with CNN features," *IEEE Access*, vol. 6, pp. 1155–1166, 2017.

[21] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transp. Res. C, Emerg. Technol.*, vol. 54, pp. 187–197, May 2015.

[22] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, vol. 17, no. 4, p. 818, 2017.

[23] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241. doi: 10.1007/978-3-319-24574-4_28.

[24] S. Lefèvre, C. Sun, R. Bajcsy, and C. Laugier, "Comparison of parametric and non-parametric approaches for vehicle speed prediction," in *Proc. Amer. Control Conf. (ACC)*, Jun. 2014, pp. 3494–3499.

[25] D. Krajzewicz, "Traffic simulation with SUMO—Simulation of urban mobility," in *Fundamentals of Traffic Simulation*. New York, NY, USA: Springer, 2010, pp. 269–293. doi: 10.1007/978-1-4419-6142-6_7.

[26] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 62, no. 2, p. 1805, 2000.

[27] Z. Ghahramani and M. I. Jordan, "Supervised learning from incomplete data via an EM approach," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 120–127.

[28] L. Mozaffari, A. Mozaffari, and N. L. Azad, "Vehicle speed prediction via a sliding-window time series analysis and an evolutionary least learning machine: A case study on San Francisco urban roads," *Int. J. Eng. Sci. Technol.*, vol. 18, no. 2, pp. 150–162, 2015.

[29] Z. Ma, J.-H. Xue, A. Leijon, Z.-H. Tan, Z. Yang, and J. Guo, "Decorrelation of neutral vector variables: Theory and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 129–143, Jan. 2018.

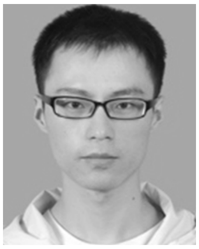
[30] J. Park *et al.*, "Real time vehicle speed prediction using a neural network traffic model," in *Proc. 2011 Int. Joint Conf. Neural Netw. (IJCNN)*, Jul./Aug. 2011, pp. 2991–2996.

[31] Z. Ma, Y. Lai, W. B. Kleijn, Y.-Z. Song, L. Wang, and J. Guo, "Variational Bayesian learning for Dirichlet process mixture of inverted Dirichlet distributions in non-Gaussian image feature modeling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 449–463, Feb. 2018.

KUN NIU is currently an Associate Professor with the School of Software Engineering, Beijing University of Posts and Telecommunications. Her research interests include big data analysis, data mining, intelligent information process, and industry application.

HUIYANG ZHANG received the B.E. degree from the Beijing University of Posts and Telecommunications, in 2017, where he is currently a Grade 2 Master Student with the School of Software Engineering. His research interests include data mining and knowledge discovery.

TONG ZHOU received the B.E. degree from the Beijing University of Chemical Technology, in 2018, where he is currently a Grade 1 Master Student with the School of Software Engineering. His research interests include machine learning and data mining.



CHENG CHENG graduated from the Beijing University of Posts and Telecommunications, in 2017, where he is currently pursuing the master's degree in software engineering. His research interests include big data mining and machine learning.

CHAO WANG received the B.E. degree from the Beijing University of Posts and Telecommunications, in 2018. He was admitted to the School of Software Engineering, Beijing University of Posts and Telecommunications. His research interests include data mining and knowledge discovery.