D-LSTM: Short-Term Road Traffic Speed Prediction Model Based on GPS Positioning Data

Xianwei Meng, Hao Fu, Liqun Peng, Guiquan Liu, Yang Yu,
Zhong Wang, and Enhong Chen, *Senior Member, IEEE*

***Abstract*— Short-term road traffic speed prediction is a long-standing topic in the area of Intelligent Transportation System. Apparently, effective prediction of the traffic speed on the road can not only provide timely details for the navigation system concerned and help the drivers to make better path selection, but also greatly improve the road supervision efficiency of the traffic department. At present, some researches on speed prediction based on GPS data, by adding weather and other auxiliary information, using graph convolutional neural network to capture the temporal and spatial characteristics, have achieved excellent results. In this paper, the problem of short-term traffic speed prediction based on GPS positioning data is further studied. For the processing of time series, we innovatively introduce Dynamic Time Warping algorithm into the problem and propose a Long Short-Term Memory with Dynamic Time Warping (D-LSTM) model. D-LSTM model, which integrates Dynamic Time Warping algorithm, can fine-tune the time feature, thus adjusting the current data distribution to be close to the historical data. More importantly, the fine-tuned data can still get a distinct improvement without special treatment of holidays. Meanwhile, considering that the data under different feature distributions have different effects on the prediction results, attention mechanism is also introduced in the model. Our experiments show that our proposed model D-LSTM performs better than other basic models in many kinds of traffic speed prediction problems with different time intervals, and especially significant in the traffic speed prediction on weekends.**

***Index Terms*— Traffic forecast, speed prediction, neural network.**

I. Background

WITH the rapid development of the national economy, the number of cars has increased dramatically, which is in sharp contradiction with the limited road resources, and leads to the increasingly frequent traffic congestion in cities. These factors make traffic prediction a sub-problem to be solved in Intelligent Transportation Systems (ITS) [1]. Accordingly, the accurate prediction of road conditions can not only effectively help drivers and navigation systems to select and plan their routes, but also help relevant regulatory authorities to conduct timely supervision [2], [3].

In the short-term traffic speed prediction methods, both the traditional non-parametric methods and the parametric methods have been constantly improved [4]–[6]. Later, with the development of deep learning, artificial neural network have gradually become a mainstream method for predicting traffic speed. Long Short-Term Memory (LSTM) neural network is a typical representative of the application of artificial neural network in traffic speed prediction because of its good processing ability for time series data [7]–[9]. Recently, some researchers have also promoted the development of short-term traffic speed prediction to a certain extent by using the auxiliary information of road periphery and weather [10].

In terms of data acquisition, most of the existing methods are based on image, video and other information obtained by infrastructural sensors that monitor the traffic condition of the whole road. But with the gradual maturity of GPS technology, we can easily obtain a large number of traffic data through satellite, which leads to more and more researchers to study road traffic on GPS traffic data [11], [12].

Although there are many effective models to predict short-term traffic speed, the structure of most of these models are complex. At present, the use of neural networks for short-term traffic speed prediction faces at least the following challenges: (1) The data of traditional algorithms are mostly obtained by road infrastructure sensors, but the researches on GPS positioning data is relatively rare. (2) Though the daily data in history are similar in the overall distribution, there are still exists slight differences in the specific time. From this point, how can we make LSTM achieve better results in the short-term traffic speed prediction problem? (3) The traffic data obtained at different time positions have different effects on the prediction results, and we can make a trade-off in these data.

In this paper, we propose a DTW-based LSTM model (D-LSTM) with a relatively simple structure for speed prediction. D-LSTM model takes the road traffic flow and time information extracted from GPS data of online car-hailing as

Manuscript received September 18, 2019; revised April 5, 2020 and June 18, 2020; accepted October 6, 2020. This work was supported by National Nature Science Foundation of China under the Grant No. 61703160, National Key Research and Development Program of China under the Grant No. 2018YFB1801105 and STCSM 18DZ2270700. The Associate Editor for this article was J. Sanchez-Medina. *(Corresponding author: Hao Fu, Guiquan Liu.)*

Xianwei Meng, Hao Fu, Guiquan Liu, Yang Yu, Zhong Wang, and Enhong Chen are with the Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China, Hefei 230052, China (e-mail: mxianwei@163.com; hfu@mail.ustc.edu.cn; gqliu@ustc.edu.cn; koinu@mail.ustc.edu.cn; gracewz@mail.ustc.edu.cn; cheneh@ustc.edu.cn).

Liqun Peng is with the School of Transportation and Logistics, East China Jiaotong University, Nanchang 330013, China (e-mail: lq.peng@ecjtu.edu.cn).

Digital Object Identifier 10.1109/TITS.2020.3030546

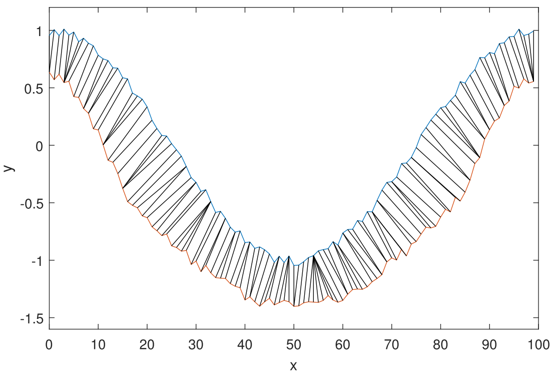


Fig. 1. Warping two curves using the DTW algorithm.

the original input, introduces dynamic time warping (DTW) algorithm to warping time feature which can fine tune the time feature of input data using historical data, and then extracts the features for prediction using LSTM network with attention mechanism. The introduction of attention mechanism can weigh the influence level of different time periods. Finally, we use full-connected neural network to predict the speed. It is precise because of the introduction of DTW algorithm that D-LSTM model can also have a better prediction effect on holidays without adding additional information.

The rest of this paper is organized as follows. Section II introduces the related work of DTW algorithm and LSTM. Section III proposes the D-LSTM model to predict short-term traffic speed. Section IV discusses the experimental design and performance of the D-LSTM model. Finally, we conclude in section V.

II. Related Work

A. Dynamic Time Warping

Dynamic Time Warping (DTW) was proposed in the 1960s. It is a method to measure the similarity between two time series of different lengths [13]. It is also widely used in template matching, such as isolated word speech recognition, gesture recognition, data mining, and information retrieval [14]–[17]. Recently, an end-to-end multi-task learning temporal convolutional neural network (MTL-TCNN) is proposed to predict short-term passenger demand in a multi-zone level, which combines the spatiotemporal dynamic time warping algorithm and can solve the multi-task prediction problem well with the consideration of spatiotemporal correlations [18]. In the research of [19], researchers propose a novel bagging tree and DTW integrated algorithm for the detection of driving events employing acceleration and orientation data from a smartphone's low cost three-axis accelerometers and gyroscopes.

The principle of DTW algorithm is as follows. For two time series $Q = \{q_1, q_2, \ldots, q_n\}$ and $P = \{p_1, p_2, \ldots p_m\}$, where n and m represent their sequence lengths, respectively. Define a warping path between two pulses as $L = \{l_1, l_2, \ldots, l_K\}$, where K is the path length. Warping path refers to the mapping between two waveforms, which is visually represented by the mapping relationship between the connecting lines of the two pulse points in Fig. 1. The kth element $l_k = (i, j)_k$ in the path represents that the ith point of Q corresponds to the jth point of P, and the path distance between them can be expressed as $d(l_k) = |q_i - p_j|$. Warping path need to satisfy three conditions: (1)boundary conditions which is described as $l_1 = (1, 1)_1$ and $l_K = (n, m)_K$. Simply stated, this requires that the start and end points of the two time series meet the mapping relationship; (2)continuity which is expressed as if $l_{k-1} = (a, b)_{k-1}$, then $l_k = (a', b')_k$ must satisfy $a' - -a \leq 1$ and $b' - b \leq 1$. That is to say, it is impossible to cross a certain point to match, only to align with the adjacent points. This ensures that every coordinate in Q and P appears in L; (3)monotony which is stated as if $l_{k-1} = (a, b)$, then for the next point $l_k = (a', b')$ of the warping path, $0 \leq a' - a$ and $0 \leq b' - b$ need to be met. This limits the point in L to be monotonous over time [20], [21]. DTW algorithm uses dynamic programming method to find an optimal warping path to minimize the cumulative distance of the optimal path, and the calculation formula is described as

$$L_{DTW}(Q, P) = \min_{L} \sum_{k=1}^{K} d(l_k), \tag{1}$$

where $L_{DTW}(Q, P)$ is the optimal path distance of Q and P. Obviously, the shorter the path distance calculated by DTW algorithm, the higher the similarity between the two waveforms, the more likely it is a pair of matched waveforms. In this paper, we will make use of the warping path under the optimal path distance. Because the warping path contains the mapping relationship between elements in two sequences, we can scale the dimension of the time feature based on this.

B. Long Short-Term Memory Neural Network

Recursive neural networks (RNN) introduce the feedback mechanism of time series, which is of great significance in the analysis of time series signals such as voice and music. On this basis, Hochreater and Schmidhuber proposed the LSTM network structure in 1997 [22]. By introducing the time memory unit, they can learn the dependency information of different lengths in time series and overcome the problems of gradient vanishment and gradient explosion in traditional RNN [23], [24]. Therefore, it has a better effect on dealing with and predicting interval and delay events in time series. LSTM consists of an input layer, a hidden layer and an output layer, in which the hidden layer is different from the original neural network, and its basic unit is memory block [25]. The ingenuity of LSTM lies in that the weight of self-circulation is changed by adding input gate, forgetting gate and output gate, so that the integral scale can be changed dynamically at different time under the condition of fixed model parameters, thus avoiding the problem of gradient disappearance or gradient expansion.

The network structure of LSTM is shown in Fig. 2. For the tth neuron of LSTM, the input is: input value x_t at time t, output value h_{t-1} at time $t-1$ and state c_{t-1} of gate control unit at time $t-1$. The output of LSTM is: the output value h_t of LSTM at time t and the state c_t of gate unit at time t. In LSTM, the forgetting gate determines the impact of c_{t-1} on c_t, the input gate determines the impact of x_t on c_t, and

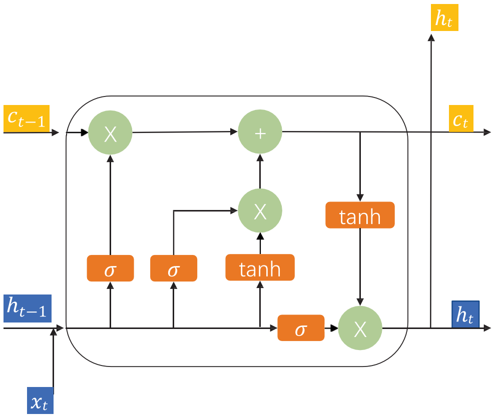


Fig. 2. LSTM neuron structure.

the output gate controls the impact of c_t on h_t. The formulas of forgetting gate, input gate and output gate are shown in formula (2), (3) and (4), respectively.

$$f_t = \sigma(W_f * h_{t-1} + W_f * x_t + b_f), \tag{2}$$
$$i_t = \sigma(W_i * h_{t-1} + W_i * x_t + b_i), \tag{3}$$
$$o_t = \sigma(W_o * h_{t-1} + W_o * x_t + b_o). \tag{4}$$

Among them, $\sigma(.)$ represent rectified linear unit (relu) activation functions, f_t, i_t and o_t are the calculation results of forgetting gate, input gate and output gate states respectively; W_f, W_i and W_o are the weight matrices of forgetting gate, input gate and output gate respectively; b_f, b_i and b_o are the bias terms of forgetting gate, input gate and output gate respectively; and the final output of LSTM is determined by output gate and unit state.

Then, formula (5), (6) and (7) is defined to calculate the values of c_t, and h_t.

$$\widetilde{c_t} = \tanh(W_c * h_{t-1} + W_c * x_t + b_c), \tag{5}$$
$$c_t = f_t * c_{t-1} + i_t * \widetilde{c_t}, \tag{6}$$
$$h_t = o_t * \tanh(c_t). \tag{7}$$

Among them, $\widetilde{c_t}$ is the state of the input unit at t-time; W_c is the weight matrix of the input unit state; b_c is the state bias term of the input unit; tanh is the activation function; the input threshold and the forgetting threshold update $\widetilde{c_t}$ and c_{t-1} synthetically into c_t.

III. D-LSTM Model

To solve the problem of short-term traffic speed prediction based on GPS data, we propose the D-LSTM model. Fig. 3 shows the structure of the D-LSTM model. The input of Long Short-Term Memory with Dynamic Time Warping (D-LSTM) is the road characteristics of several time intervals extracted from GPS data. In the model structure diagram, we can see that D-LSTM consists of three modules, the first one is DTW-layer, which uses DTW algorithm to fine-tune the features of input data to make it more close to historical data; the second one is attention-based LSTM layer, which uses attention mechanism to weigh the comprehensive features of different data distributions. The third module is a fully connected network for traffic speed prediction.

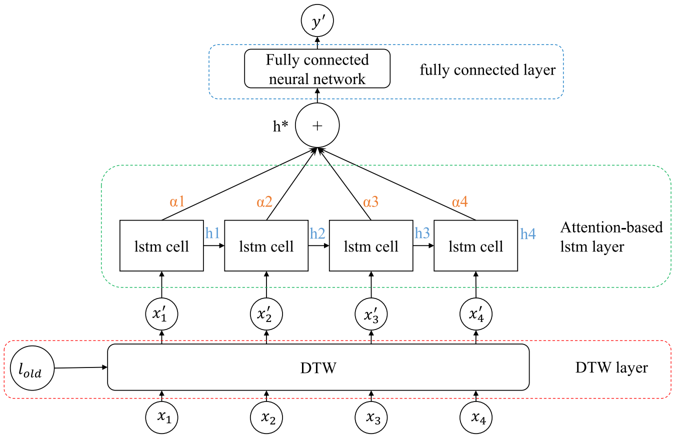


Fig. 3. Architecture diagram of the D-LSTM model. The D-LSTM model is composed of three modules: DTW layer, attentiion-based LSTM layer and fully connected layer. DTW layer preprocesses the input data, attentiion-based LSTM layer extracts the deep features that can be predicted from the adjusted data, and finally forecasts through fully connected layer.

A. Input Data Structure

Assuming that the input features of LSTM are extracted from q consecutive time intervals and the length of each time interval is Δt, we can construct a two-dimensional matrix $X = (x_1, x_2, \ldots, x_q)^T$ as the original input of D-LSTM, and $x_i = (v_i, t_i, car\_n_i, p\_c_i, s\_p\_c_i)$ represents the road traffic characteristics in the ith time interval, where v_i, t_i, car_n_i, p_c_i and $s\_p\_c_i$ represent the average road speed, time, the number of cars in the road, the number of GPS positioning records and the number of records with zero speed, respectively. It is noteworthy that we convert the value of time t_i into a minute representation, i.e. its value ranges from 0 to 1440.

Since the speed in the GPS positioning data is the instantaneous speed of the car, which shows poor representation ability for the road speed in the current time interval, so we use the velocity-time integration model to calculate the v value in each interval. For the GPS positioning sequence of a car $((v_1, t_1), \ldots, (v_k, t_k), \ldots (v_n, t_n))$, we use (v_k, t_k) to express the instantaneous velocity and the time of the kth GPS point. Then the formula for calculating v_{car} is shown in formula (8).

$$v_{car} = \frac{s}{t} \approx \frac{1}{t_n - t_0}[v_0(\frac{t_1 - t_0}{2}) + v_n(\frac{t_n - t_{n-1}}{2}) + \sum_{k=1}^{n-1} v_k(\frac{t_{k+1} - t_{k-1}}{2})]. \tag{8}$$

Assuming that there are p cars on the road in the current interval, the speed values of each car calculated by formula (8) are v_1, v_2, …and v_p, then the average speed of the road can be calculated directly by arithmetic average method. The calculation formula is shown in formula (9).

$$v_{road} = \frac{\sum_{i=1}^{p} v_i}{p}. \tag{9}$$

By combining the road information of different intervals in time sequence, we can get a two-dimensional data matrix with size $(q, 5)$. Thus, the structure of the input data is shown in

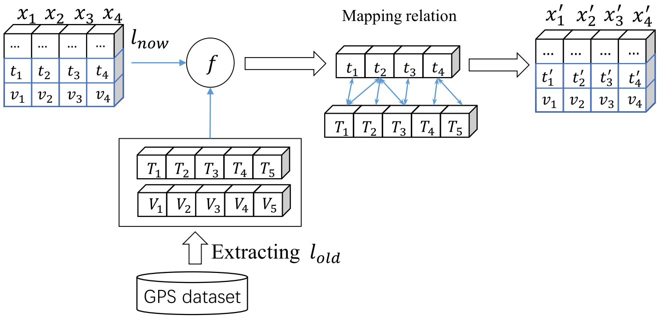


Fig. 4. Processing flow chart of DTW layer. First, extract l_{now} and l_{old} from current data and historical data respectively, where t_i, v_i represent the time and speed of lnow, and T_i, V_i represent the time and speed of lold, respectively. Then calculate the mapping relationship of time feature by function f, where f is DTW algorithm. Finally, update l_{now} by mapping relationship.

formula (10).

$$X = \left[x_1 \; x_2 \; \ldots \; x_q\right]^T = \begin{bmatrix} v_1 & t_1 & car\_n_1 & p\_c_1 & s\_p\_c_1 \\ v_2 & t_2 & car\_n_2 & p\_c_2 & s\_p\_c_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_q & t_q & car\_n_q & p\_c_q & s\_p\_c_q \end{bmatrix}. \tag{10}$$

B. DTW Layer

In the road network, the traffic conditions of different roads are different. However, for the same road, although there are some differences in traffic conditions in recent days, the overall distribution is similar. Therefore, if the current data can be fine-tuned to be as similar to the historical data, the accuracy of the model can be effectively improved. Furthermore, the variation of road traffic speed with time of the adjacent days are very similar, while there are some differences in the time dimension. Based on this, we introduce the DTW algorithm to warping the time dimension of the input data x_q, so that the current data features are closer to the historical data features.

Fig. 4 shows the processing of DTW layer. DTW algorithm requires the current data sequence $l_{now} = (l_{n1}, l_{n2}, \ldots, l_{nq})$ and a template data sequence $l_{old} = (l_{o1}, l_{o2}, \ldots, l_{op})$ for warping. Each element in the sequence is a binary (v, t). The binary is composed of the average speed v and the time t at the end of each interval. Among them, we set the time of l_{o1} earlier than l_{n1} and l_{op} later than l_{nq}. This is because the time dimension of two sequences may be contracted or extended when warping, which may make the time range wider. Then, in order to ensure that the selected template sequence has less impact on warping, we add the first and last data (l_{o1} and l_{op}) in l_{old} to l_{now}, to ensure that the starting and ending point of DTW is the same, resulting in a new current sequence $l'_{now} = (l_{o1}, l_{n1}, l_{n2}, \ldots, l_{nq}, l_{op})$. From formula (1), we can get the optimal matching between elements in l'_{now} and l_{old}, then we can get the mapping from each time point in l_{now} to one or more time points in l_{old}. Assuming that the time of the jth data l_{nj} in l_{now} has a mapping relationship with k time points $(l'_{o1}, l'_{o2}, \ldots, l'_{ok})$ in l_{old}, after l_{nj} passes through DTW-layer, its time is adjusted according to the following formula.

$$l_{nj}.t' = \frac{l'_{o1}.t + l'_{o2}.t + \ldots + l'_{ok}.t}{k}, \tag{11}$$

where $l.t$ represents the time corresponding to the sequence element l. In other words, formula (11) updates the time attribute of each element in sequence l'_{now} with the time attribute of the mapped sequence in l_{now}. It is worth noting that due to the continuity and monotony of the DTW algorithm, DTW-layer not only ensures that there is no temporal overlap in the warping process, but also ensures that the mapping time of l_{now} is increasing accordingly. After the above processing, the fine-tuned value of time for each element in l_{now} can be obtained, we update the original time with this value to obtain the input matrix x'_q of the attention-based LSTM layer. The fine-tuned matrix x'_q is shown in formula (12).

$$X' = \left[x'_1 \quad x'_2 \quad \ldots \quad x'_q\right]^T = \begin{bmatrix} v_1 & t'_1 & car\_n_1 & p\_c_1 & s\_p\_c_1 \\ v_2 & t'_2 & car\_n_2 & p\_c_2 & s\_p\_c_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_q & t'_q & car\_n_q & p\_c_q & s\_p\_c_q \end{bmatrix}. \tag{12}$$

C. Attention-Based LSTM Layer

LSTM can deal with time series problem well. Assuming that the number of hidden layer neurons of LSTM is set to r, in the traditional LSTM network structure, we can finally get a deep feature of $r * 1$ dimension for prediction.

With the significant progress of LSTM model based on attention mechanism in natural language processing and computer vision [26], [27], some researchers also use similar structures to weigh the correlation between different days in traffic flow prediction, and achieve better prediction results [28], [29]. In this paper, we use a similar attention mechanism to construct a attention-based LSTM layer in D-LSTM, which enables D-LSTM model to assign different weights to different road features, thus achieving better prediction result. The inspiration of attention mechanism comes from human beings themselves: when our vision perceives the scene in front of us, it does not always look at everything in a scene, but only at what we want to see. Its core operation is to learn a set of weight parameters, and then merge the elements according to their importance. Weight parameter is a coefficient of attention allocation, which indicates how much attention is allocated to the corresponding elements.

Attention mechanism can be divided into three steps: first, information input; second, calculation of attention distribution α; third, calculate the weighted average of input information based on α. In the attention-based LSTM layer, we use a fully connected neural network to generate attention distribution α. For the input value x'_q, we design a fully connected neural network with the number of hidden layer units s and output layer neurons 1, then we can get an initial weight matrix M

of size $q * 1$:

$$M = [u_1, u_2, \ldots, u_q]^T, \tag{13}$$

$$u_i = \tanh(W_h * h\_a_i + b_2) \quad i = 1, 2, \ldots, s, \tag{14}$$

$$h\_a_i = \tanh(W_x * x'_i + b_1) \quad i = 1, 2, \ldots, q. \tag{15}$$

The matrix M can then be converted into matrix M' by the softmax normalization formula shown in formula (17).

$$M' = [\alpha_1, \alpha_2, \ldots, \alpha_q]^T. \tag{16}$$

$$\alpha_i = \frac{\exp(u_i)}{\sum_{j=1}^{q} \exp(u_j)} \quad i = 1, 2, \ldots, q. \tag{17}$$

where q is the number of time intervals selected. After obtaining the weight matrix M', the deep features of D-LSTM model with attention mechanism are obtained by formula (18).

$$h* = \sum_{i=1}^{q} \alpha_i * h_i. \tag{18}$$

The above formula uses the weighted average method to obtain the weighted features, in which h_i is the hidden state output value of LSTM.

D. Speed Prediction and Model Training

Feedforward neural network is one of the simplest neural networks, with layered arrangement of neurons. Each neuron only receives the output from the former layer and outputs the processed value to the next layer without feedback between the layers. At present, it is one of the most widely used and rapidly developed artificial neural networks. In the D-LSTM model, the fully connected neural network we use contains only a single hidden layer. The input of the network is $h*$. we set the number of neurons in the hidden layer and the output layer to g and 1 respectively. So the calculation formula of of the predicted speed value y' is as follows.

$$y' = f(h'), \tag{19}$$

$$h' = \sum_{i=1}^{g} v_i * h_i + b_2, \tag{20}$$

$$h_j = \sigma(\sum_{i=1}^{h} W_{ij} * h * + b_1) \quad j = 1, 2, \ldots, g, \tag{21}$$

where $\sigma(.)$ and $f(.)$ represent relu and sigmoid activation functions respectively. For the training of the model, we use the mean square error (MSE) and gradient descent method as the loss function and training method. Assuming that the training data set is $\{sample_1, sample_2, \ldots, sample_m\}$, where $sample_i = [x_i, y_i]$, y_i represents the ground truth. We use y'_i as the predicted value of the D-LSTM model, and the formula for calculating the loss function is shown in formula (22).

$$loss = \frac{1}{m} \sum_{i=1}^{m} (y_i - y'_i)^2. \tag{22}$$

The following algorithm 1 gives the training process of the model, in which the training process does not pass through the DTW layer.

Algorithm 1 Training Algorithm of D-LSTM Model

```
Require: $x$, number of training rounds $epochs$, $batch\_size$
Ensure: The parameter $p$ required by the D-LSTM model
Standardize the data to get $x$.
for $i = 0$ to $epochs$ do
    Randomly extract training data of size $batch\_size$ from $x$.
    Calculate $h*$ by formulas (17) and (18).
    Calculate the predicted speed by equations (19), (20) and (21).
    Optimizing loss function in formula (22) by Gradient Descent Method.
    if D-LSTM has reached the convergence criterion then
        berak
```

IV. Experiments

A. Dataset

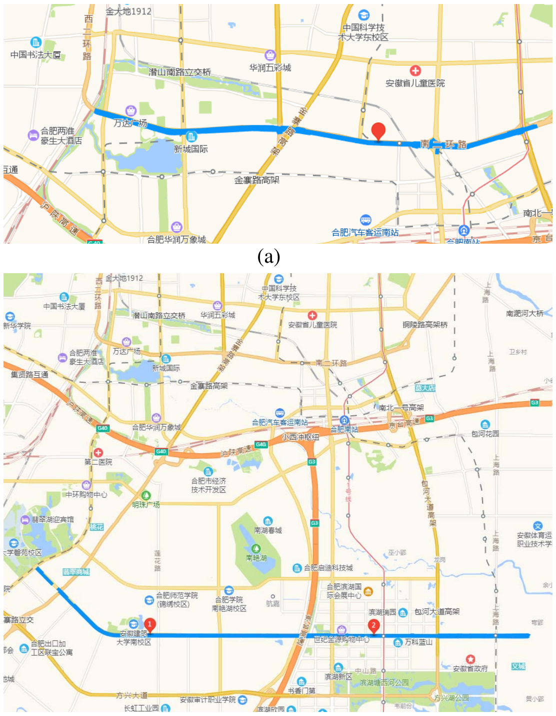
All the experimental data are collected from GPS positioning data of online car-hailing in Hefei City, Anhui Province. For each positioning data is uploaded in real time on the way by Didi,[1] which is the largest online car-hailing company in China. Our experiment is carried out on the dataset collected from the South Second Ring Road and Ziyun Road. The location of this road is shown in Fig. 5. The collection time of experimental data is from January 5, 2019 to January 30, 2019. These data are sampled from cars traveling from west to east, with a total of 641244 GPS positioning data, and the number of cars passing through each day is about 2815. It is worth noting that no traffic accident occurred in the experimental period, and the speed value of the positioning data is in km/h. Since the model introduces the DTW algorithm, the training set and test set of the model are not directly divided. We first use the data of the last five days to train the model. Then, we use the data of the previous day to train the model again, and extract the reference sequence of DTW algorithm from previous day's data.

B. Data Processing

1) Exception Handling:

- There are some abnormal cars in the road, whose location longitude and latitude are invariable for a long time and the speed is 0. These abnormal data may be caused by illegal driving or abnormal equipment. We delete these abnormal data directly.
- When analyzing the data in different time periods, we find that from 0:00 to 6:00, the GPS positioning records of online car-hailing in this period will be much less than those of other periods. In many cases, the number of GPS record is 0, this is in line with the fact that fewer people travel at night. So we delete the data in this period, and the experiment only predicts the speed of the road from 6:01 to 23:59.
- For an online car-hailing order, if the positioning data of different terminals (GPS records are generated by both

[1] https://www.didiglobal.com/



(a)

(b)

Fig. 5. Experimental data sampling road: (a) South Second Ring Road, Baohe District, Hefei City, Anhui Province. (b) Ziyun Road, Shushan District, Hefei City, Anhui Province. It is noteworthy that the area shown in (b) contains (a), and the spatiotemporal features required by other baseline models are extracted from the area shown in (b).

the car terminal and the driver's mobile terminal.) does not intersect in the time dimension, we directly delete the order data.

2) *Standardization of Data:* The purpose of data normalization is to map all features to the same scale, so as to avoid the dominant role of some features due to the different dimensions. We use formula (23) to normalize each feature in the dataset, which maps all the original data to intervals [0-1].

$$x_{scale} = \frac{x - x_{min}}{x_{max} - x_{min}}. \tag{23}$$

3) *Filtering:* Because the number of online booking car in a road is relatively small, and there are many unknown factors in the road condition, which will lead to a lot of noise in the value y calculated directly by the formula (9). Here, we use the mean filter (also called simple blurring) to smooth the original speed value. The speed effect figure before and after smoothing is shown in Fig. 6, among them, we use a filter window size of 10 minutes.

C. Evaluation Criteria

Different measurement criteria have different values for different models. Mean absolute error (MAE), mean square error (MSE), root mean square error (RMSE) and R-squared are the most commonly used evaluation indicators for regression models. Moreover, MAE and RMSE were used in previous work [24], [25], [30]. In this paper, we use MAE, MSE and R-square as our evaluation indicators.

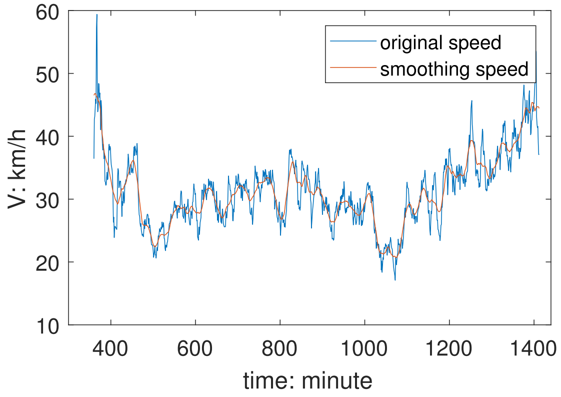


Fig. 6. Smoothing the predicted speed. The x-axis is time, and we express it in terms of the total minutes of the day.

D. Experiment Setup

We implement our D-LSTM model with python Keras [31], which uses tensorflow as backend. All experiments are executed on a server with 4 Tesla K80. In the parameter setting of feature extraction, we set the values of interval length, number of intervals and predicted time length to 10 minutes, 3, 10 minutes, respectively. In DTW layer, we set the start time of l_{old} is 30 minutes earlier than l_{now} and the end time is 30 minutes later than l_{now}. In attention-based LSTM layer, the number of neurons in the hidden layer of LSTM and the full connection layer of computing attention distribution α was set to 50. In fully connected layer, we set the number of neurons in the hidden layer to 50.

E. Performance Comparison

We compare the performance of the proposed method with the following basic and advanced methods. The optimal parameters of these methods are found by grid search in scikit-learn, and experiments are carried out under the optimal parameter set.

- K-NearestNeighbor (KNN): The K value of KNN is set to 5, and the predicted speed value is to average the y value of these five data.
- ANN: ANN is a single hidden layer neural network, and the number of neurons in the hidden layer is set to 50.
- LSTM: We set the number of neurons in the hidden layer to 50.
- SAE [24]: Stacked autoencoder (SAE) is a model used to learn general traffic flow features and train in a greedy layerwise fashion.
- DCRNN [25]: Diffusion Convolutional Recurrent Neural Network model combines spectral graph convolution with gated recurrent gate (GRU) for traffic forecasting.
- STGCN [30]: Spatio-Temporal Graph Convolutional Networks is a traffic prediction model using convolutional structure.
- A-LSTM: Attention-based LSTM only contains attention mechanism without DTW preprocessing.

Among them, the Spatio-Temporal Graph features needed in models DCRNN and STGCN are extracted from the region shown in Fig. 5(b).

TABLE I
EVALUATION OF DIFFERENT PREDICTION MODELS

Model	5 minute			10 minute			15 minute		
	MAE	MSE	R^2	MAE	MSE	R^2	MAE	MSE	R^2
KNN	2.51	5.35	0.803	3.14	10.61	0.745	3.94	13.61	0.714
ANN	2.03	4.81	0.801	1.92	4.14	0.814	3.03	7.96	0.734
LSTM	1.87	4.58	0.815	2.01	5.36	0.807	3.05	7.35	0.788
SAE	2.21	4.67	0.813	2.06	4.53	0.811	3.28	8.67	0.793
DCRNN	1.81	4.53	0.822	1.70	4.03	0.851	2.69	6.36	0.821
STGCN	1.76	4.35	0.828	1.61	4.06	0.859	2.45	**5.94**	**0.826**
A-LSTM	1.82	4.42	0.820	1.67	4.11	0.846	2.71	6.15	0.812
D-LSTM	**1.72**	**4.21**	**0.833**	**1.53**	**3.98**	**0.862**	**2.43**	6.03	0.810

Because DTW algorithm in D-LSTM model needs to select the time and speed values of the day before prediction as reference sequence, the division of our training and test sets will differ from the traditional method of partitioning. At first, we used data of the latest 5 days to train. Then, we used the data of the previous day to train D-LSTM, ANN, LSTM and attention-based LSTM once more and forecast the road traffic speed in the next day. Finally, we conducted 21 sets of experiments, each of which is predicting the speed of road traffic in the coming day.

1) Performance on the Whole Testing Set: The input features are extracted from the last 3 time intervals and each time interval is 10 minutes in length. We set the prediction time to 5 minutes, 10 minutes and 15 minutes respectively. Then the performance of D-LSTM and the model compared with it is shown in Table I. All the values are averaged for 21 groups.

From the experimental results, we can draw the following conclusions: (1) all of the neural network models have better prediction performance than the statistical KNN model; (2) the prediction performance of LSTM is significantly better than that of the full connected neural network ANN; (3) the models A-LSTM, DCRNN and STGCN have a promising prediction effect, in addition, STGCN with the spatiotemporal characteristics achieves the best performance in the relative longer time interval 15 minutes based on the metrics of MSE and R^2; (4) D-LSTM, which adds DTW to A-LSTM, achieving the best predict performance among all comparative methods in the next 5 minutes and 10 minutes, which demonstrates the effectiveness of DTW.

2) Effect of the Model on Holidays: In Fig. 7, the speed-time chart of the road on an adjacent Friday and Saturday is shown. It can be seen that for the road we studied, the morning and evening peak of the weekend has little time fluctuation compared with that in the week. Based on this, there is no distinction between weekdays and weekends in this experiment.

Table II shows the performance of D-LSTM and other models in predicting the traffic speed in the next 10 minutes of the weekend. The values in Table II are the average of the six sets of values because the weekend in the data only contains 6 days. It can be seen that the D-LSTM model with DTW layer has a significant advantage in predicting the speed of road traffic on weekends.

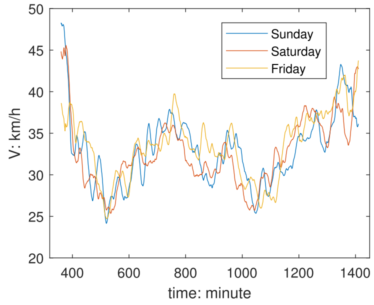


Fig. 7. Speed-time chart of the road on the adjacent Friday and weekend.

TABLE II
PERFORMANCE EVALUATION OF DIFFERENT MODELS OVER THE WEEKEND

	MAE	MSE	R^2
KNN	3.23	9.75	0.721
ANN	2.46	5.44	0.759
LSTM	1.91	4.79	0.802
SAE	2.36	5.19	0.773
DCRNN	1.83	4.66	0.807
STGCN	1.76	4.17	0.815
A-LSTM	1.83	4.26	0.814
D-LSTM	**1.66**	**4.11**	**0.823**

3) Effect of DTW-Layer and Attention-Based Layer: In order to verify the effectiveness of DTW as a data preprocessing algorithm and the effect of attention mechanism in the model, we design the following comparative models: (1)A-LSTM which contains the attention mechanism without DTW preprocessing; (2)D-DCRNN which adds DTW preprocessing before DCRNN method: (3)D-STGCN which represents STGCN with DTW preprocessing added. Fig. 8 shows that the indicators of LSTM, attention-based LSTM (A-LSTM) and D-LSTM in MAE and MSE increase with the number of training rounds. From the graph, we can see that attention-based LSTM has better prediction performance than LSTM, while D-LSTM, which introduces DTW algorithm, can continue to improve prediction performance on the basis of A-LSTM. In addition, we have studied the role of DTW algorithm in data preprocessing steps of existing advanced algorithms. From Fig. 9, we can see that D-DCRNN reduces MAE by 7.6% and MSE by 1.7% compared with DCRNN, and D-STGCN reduces MAE by 4.9% and MSE by 1.2% compared with STGCN.

4) Effect of Parameters: For the input features, we choose the interval length of 10 minutes, the number of intervals is 2, 3, 4 and 5, and the predicted time length is 5 minutes, 10 minutes, 15 minutes and 30 minutes. Furthermore, D-LSTM selected different number of hidden layer neurons:

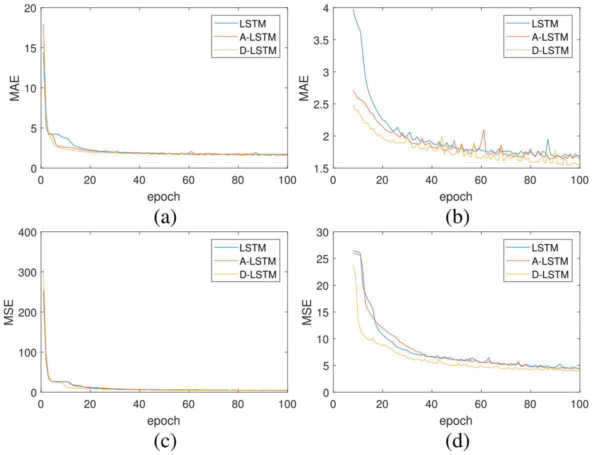


Fig. 8. (a) The variation of the MAE of LSTM, attenion-based LSTM and D-LSTM with the number of training rounds, where A-LSTM is used to represent attenion-based LSTM. (b) Partial enlargement in (a). (c) MSE indicators of different models. (d) Partial enlargement in (c).

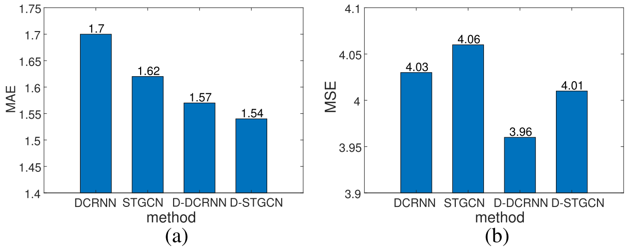


Fig. 9. DTW is applied to preprocess the input data of the existing methods.

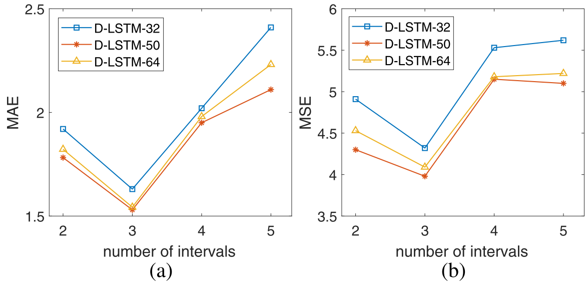


Fig. 10. Effect of time dimension on D-LSTM model performance.

the number of hidden layer neurons of D-LSTM-32, D-LSTM-50 and D-LSTM-64 are 32, 50 and 64, respectively. All models are trained using an *Adam* optimizer with a learning rate of 0.001, a batch size of 30, and epoch of 100. Fig. 10 shows the effect of different time intervals on the performance of these models when the time length to be predicted is 10 minutes. It can be seen that when the number of interval is set to 3, the performance is the best. Fig. 11 shows the performance in predicting different time lengths when the number of interval is set to 3, from which we can see that the model can achieve the best results in predicting the next 10 minutes. When the number of hidden layer neurons in LSTM is set to 50, the D-LSTM model has the best performance.

5) Effect of Time Feature: In this part, we compare and analyze the time feature in A-LSTM and A-LSTM′, where the time feature is not included in the input of A-LSTM′. Under the evaluation metric of MAE, A-LSTM and A-LSTM′ achieve 1.67 and 1.70 respectively, while under the evaluation metric of MSE, A-LSTM and A-LSTM′ achieve 4.11 and 4.21, respectively. In the A-LSTM model, whether the time features in the input features are included or not has little influence on the prediction results. However, from the experimental results in Table I, it can be seen that the introduction of DTW layer effectively fine-tuning the time features to fit the model well.

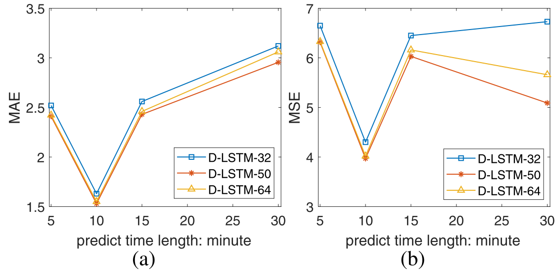


Fig. 11. The influence of predicted time length on D-LSTM model performance.

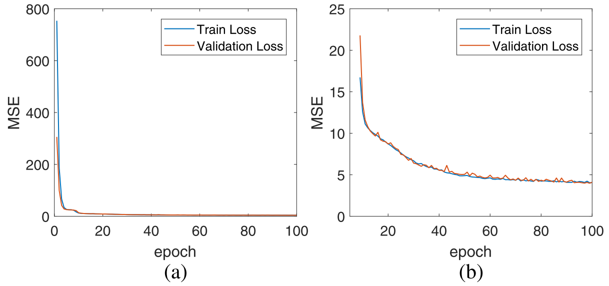


Fig. 12. (a) Influence of training rounds on verification loss and training loss of D-LSTM model. (b) Partial enlargement in (a).

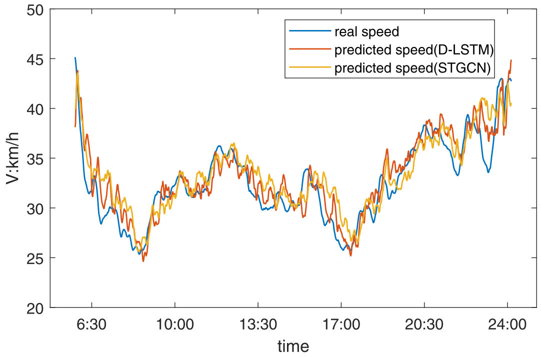


Fig. 13. Prediction display of D-LSTM model on road traffic speed.

6) Prediction Display: In Fig. 12, we show how the validation loss and train loss of the D-LSTM model vary with the number of training rounds. It can be seen that when the training epochs greater than 70, the performance slightly remain stable. Fig. 13 shows the results of the D-LSTM model and the STGCN model for road speed prediction. It can be seen that the prediction performance of these two models will be worse in the period of 5:00-6:00 and 22:30-23:59. The reason is that the number of online booking cars varies greatly in these periods. Compared with STGCN, D-LSTM has a slightly better prediction effect in the entire prediction process.

V. Conclusion

In this paper, we propose the D-LSTM model to study the short-term traffic speed prediction problem based on GPS positioning data of online car-hailing. The D-LSTM model is based on LSTM neural network, using DTW algorithm and attention mechanism to enhance the prediction ability of LSTM under different features. DTW algorithm is a dynamic time warping algorithm, which plays an important role in D-LSTM short-term traffic speed prediction. In view of the strong periodicity of traffic flow data, we assume that as long as the waveform formed by a feature and the predicted value is similar in each cycle, we can also warping the input feature to improve the performance of the model to a certain extent, otherwise, the warping of non periodic data by DTW algorithm may show little effect to the prediction performance. We evaluate our model on a car-hailing location dataset in Hefei. The experimental results show that our method is superior to several competitive methods.

For future work, we plan to investigate the following aspects: (1) find a more efficient mapping method than DTW algorithm; (2) with the deepening of knowledge map related research, we can build a knowledge map in ITS, thus greatly improving the prediction effect of the model; (3) extend D-LSTM model to other domains.

References

[1] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1624–1639, Dec. 2011, doi: 10.1109/TITS.2011.2158001.

[2] B. Yao *et al.*, "Short-term traffic speed prediction for an urban corridor," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 2, pp. 154–169, Feb. 2017, doi: 10.1111/mice.12221.

[3] 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.

[4] G. F. Newell, "A simplified theory of kinematic waves in highway traffic, Part I: General theory," *Transp. Res. B, Methodol.*, vol. 27, no. 4, pp. 281–287, Aug. 1993.

[5] S. R. Chandra and H. Al-Deek, "Predictions of freeway traffic speeds and volumes using vector autoregressive models," *J. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 53–72, 2009.

[6] T. L. Pan, A. Sumalee, R. X. Zhong, and N. Indra-payoong, "Short-term traffic state prediction based on Temporal–Spatial correlation," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1242–1254, Sep. 2013.

[7] M. G. Karlaftis and E. I. Vlahogianni, "Statistical methods versus neural networks in transportation research: Differences, similarities and some insights," *Transp. Res. C, Emerg. Technol.*, vol. 19, no. 3, pp. 387–399, Jun. 2011.

[8] 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.

[9] 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, Apr. 2017.

[10] S.-H. Huang and B. Ran, "An application of neural network on traffic speed prediction under adverse weather condition," Ph.D. dissertation, Univ. Wisconsin-Madison, Madison, WI, USA, 2003.

[11] G. Fusco, C. Colombaroni, and N. Isaenko, "Short-term speed predictions exploiting big data on large urban road networks," *Transp. Res. C, Emerg. Technol.*, vol. 73, pp. 183–201, Dec. 2016.

[12] J.-D. Chang, "Spatial-temporal based traffic speed imputation for GPS probe vehicles," in *Proc. 5th Int. Conf. Netw., Commun. Comput.*, 2016, pp. 326–330.

[13] R. Bellman and R. Kalaba, "On adaptive control processes," *IRE Trans. Autom. Control*, vol. 4, no. 2, pp. 1–9, Nov. 1959.

[14] P. Senin, "Dynamic time warping algorithm review," *Inf. Comput. Sci. Dept. Univ. Hawaii at Manoa Honolulu, USA*, vol. 855, nos. 1–23, p. 40, 2008.

[15] C. Myers, L. Rabiner, and A. Rosenberg, "Performance tradeoffs in dynamic time warping algorithms for isolated word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 28, no. 6, pp. 623–635, Dec. 1980.

[16] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 1, pp. 43–49, Feb. 1978.

[17] T. M. Rath and R. Manmatha, "Word image matching using dynamic time warping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Jun. 2003, pp. II–II.

[18] K. Zhang, Z. Liu, and L. Zheng, "Short-term prediction of passenger demand in multi-zone level: Temporal convolutional neural network with multi-task learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1480–1490, Apr. 2020.

[19] R. Sun, Q. Cheng, F. Xie, W. Zhang, T. Lin, and W. Y. Ochieng, "Combining machine learning and dynamic time wrapping for vehicle driving event detection using smartphones," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–14, 2019.

[20] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, 2001, pp. 1–11.

[21] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "Segmenting time series: A survey and novel approach," in *Data Mining in Time Series Databases*. Singapore: World Scientific, 2004, pp. 1–21.

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

[23] O. A. Abidogun, "Data mining, fraud detection and mobile telecommunications: call pattern analysis with unsupervised neural networks," Ph.D. dissertation, Univ. Western Cape, Cape Town, South Africa, 2005.

[24] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[25] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," 2017, *arXiv:1707.01926*. [Online]. Available: http://arxiv.org/abs/1707.01926

[26] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[27] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.

[28] S. Hao, D.-H. Lee, and D. Zhao, "Sequence to sequence learning with attention mechanism for short-term passenger flow prediction in large-scale metro system," *Transp. Res. C, Emerg. Technol.*, vol. 107, pp. 287–300, Oct. 2019.

[29] Z. Zhang, M. Li, X. Lin, Y. Wang, and F. He, "Multistep speed prediction on traffic networks: A deep learning approach considering spatio-temporal dependencies," *Transp. Res. C, Emerg. Technol.*, vol. 105, pp. 297–322, Aug. 2019.

[30] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional neural network: A deep learning framework for traffic forecasting," *CoRR*, vol. abs/1709.04875, 2017. [Online]. Available: http://arxiv.org/abs/1709.04875

[31] F. Chollet *et al.* (2015). *Keras*. [Online]. Available: https://github.com/fchollet/keras

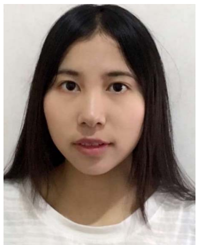
Xianwei Meng received the M.Sc. degree from the Shenyang Institute of Automation, Chinese Academy of Sciences, in 2005. He is currently pursuing the Ph.D. degree with the University of Science and Technology of China. His main research interests include data mining, machine learning, and satellite navigation application. He has 30 patents granted, including nine invention patents.

Hao Fu received the B.S. degree from Northwestern Polytechnical University, China, in 2018. He is currently pursuing the master's degree with the School of Computer Science and Technology, University of Science and Technology of China, China. His main research interests include machine learning, deep learning, and data mining.

Liqun Peng received the Ph.D. degree in transportation engineering from the Wuhan University of Technology, China, in 2015. He is currently an Associate Professor with the School of Transportation and Logistics, East China Jiaotong University. Since 2016, he had been admitted to University of Alberta as a Postdoctoral Fellow, where he had worked on studying new algorithms and information technologies to model and optimize the cooperative intelligent transportation systems (C-ITS). His research interests are in the area of connected vehicle and traffic big data, specifically for improving roadway mobility and safety on both arterial and freeway. He had been awarded an Outstanding Ph.D. Thesis Award at the 10th Annual Meeting of China Intelligent Transportation Research Academic in 2015, and a Best Paper Award at the 13th Annual China Academic Conference for Ph.D. Candidates in 2015.

Guiquan Liu received the B.Sc. and Ph.D. degrees from University of Science and Technology of China, in 1996 and 1999, respectively. He is currently an Associate Professor with the School of Computer Science and Technology, University of Science and Technology of China. His main research interests include machine learning, data mining, social network analysis, and pattern recommendation. He has published over 60 papers in refereed conferences and journals.

Yang Yu received the B.S. degree from the Dalian University of Technology, China, in 2016, and the M.S. degree from the University of Science and Technology of China, China, in 2019. His research interests include machine learning and data mining.



Zhong Wang received the B.S. degree from Anhui Normal University, China, in 2018. She is currently pursuing the M.S. degree with the University of Science and Technology of China, China. Her research interests include network embedding and deep learning.

Enhong Chen (Senior Member, IEEE) received the B.Sc. degree from Anhui University in 1989, the master's degree from the Hefei University of Technology, in 1992, and the Ph.D. degree in computer science from USTC, in 1996.

He is a Professor and a Vice Dean of the School of Computer Science, University of Science and Technology of China (USTC). He is also the Vice Director of the National Engineer Laboratory for Speech and Language Information Processing, the Director of the Anhui Province Key Laboratory of Big Data Analysis and Application, and the Chairman of Anhui Province Big Data Industry Alliance. His current research interests are data mining and machine learning, especially social network analysis and recommender systems. He has published more than 200 papers on many journals and conferences, including international journals such as IEEE Trans, ACM Trans, and important data mining conferences, such as KDD, ICDM, NIPS. His research is supported by the National Natural Science Foundation of China, National High Technology Research and Development Program 863 of China. He won the Best Application Paper Award on KDD2008 and Best Research Paper Award on ICDM2011.

Dr. Chen is a CCF Fellow, winner of the National Science Fund for Distinguished Young Scholars in 2013, scientific and technological innovation leading talent of 'Ten Thousand Talent Program' in 2017, and member of the Decision Advisory Committee of Shanghai Since June, 2018.