IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 30, 2020, accepted October 1, 2020, date of publication October 28, 2020, date of current version January 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3034551

A Multiscale-Grid-Based Stacked Bidirectional GRU Neural Network Model for Predicting Traffic Speeds of Urban Expressways

DEQI CHEN, XUEDONG YAN, XIAOBING LIU, SHURONG LI, LIWEI WANG, AND XINMEI TIAN

MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

Corresponding author: Xuedong Yan (xdyan@bjtu.edu.cn)

This work was supported in part by the Fundamental Research Funds for Beijing Jiaotong University under Grant 2019JBZ003.

ABSTRACT In recent decades, studies on short-term traffic speed forecasting of the large-scale road are a new challenge for researchers and engineers. Especially based on deep learning neural networks, studies on short-term traffic forecasting have achieved mush-room growth. This study proposes a stacked Bidirectional Gated Recurrent Unit neural network model to predict the traffic speed of the expressway over different estimation time intervals in an effective manner. By building a multiscale-grid model, it can take less time to derive a set of key traffic parameters of different scales to predict traffic speed of the various-scale road. The speed prediction of small-scale sections can cover more detailed road spatial features preparing for Vehicle Navigation System, and the speed prediction of large-scale sections can establish the real-time traffic control strategies. In order to validate the effectiveness of the proposed model, we use the floating car data, with an updating frequency of 1 minute from the urban freeway of Beijing, for model training and testing. The experimental results show that the stacked BiGRU network with the multiscale-grid model enables to capture the spatial-temporal characteristics of traffic speed efficiently. Furthermore, the BiGRU with two layers (BiGRU-2L) outperforms benchmark models in the prediction of the traffic speed, which presents a significant advantage in reducing the overfitting problem, decreasing the excessive time-consuming and improving the effective use of limited computation resources.

INDEX TERMS Stacked bidirectional gated recurrent unit network, multiscale grid model, traffic speed prediction, floating car data.

I. INTRODUCTION

Recent years have witnessed increase ever-growing traffic demand, which gradually leads to the average speed of road network becoming slow down, causing a series of traffic issues (i.e. the reducing traffic efficiency, waste of time and traffic congestion), deteriorating the traveler' experiences, tremendously [1], [2]. Nowadays, it is widely acknowledged that short-term traffic prediction can serve as an effective tool to address or alleviate the problem. Short-term traffic prediction can be applied to detect the traffic speed of the road network efficiently for the rapid response to various traffic status. What's more, successful implementation of traffic speed prediction can not only identify the traffic state of the road network to enhance transportation professionals' decision-making process but also benefit travelers' route pre-planning and rescheduling [3].

With the rapid development of data storage and data processing techniques, the real-time traffic can be obtained by various sensors such as infrastructure sensors, mobile sensors, loop sensors, microwave sensors and traffic cameras and so on [4]. As a ubiquitous kind of mobile sensor, the floating cars can provide speed, time and position information to probe a city's rhythm and pulse [5], [6]. In comparison with the conventional data collecting method, the floating car data (FCD) has three important advantages. First, real-time traffic data (i.e. the traffic speed, time, longitude and latitude information) can be collected, and automatically sent

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

to a processing center where the traffic parameters (such as upstream and downstream speed, average speed, history speed, time slice etc.) are directly extracted [7]. Moreover, the FCD enables to monitor the large-scale road network and facilitate the large-scale speed prediction, which can cover a wider area than fixed sensors with observing a limited number of areas [8]. Finally, via GPS-equipped vehicles, high-quality data is collected with low costs and high efficiency [9]. All these characteristics have made FCD a popular mainstream data collection method for traffic speed prediction [10].

Based on the map-matching and Geographic Information System (GIS) techniques, lots of studies have been conducted to extract traffic parameters from the FCD [11], [12]. However, two major obstacles are revealed in applying these techniques to apply the FCD to road networks. On the one hand, it has been generally acknowledged that map-matching is a time consuming and complicated tool [13], making it challenging to match massive FCD trajectories across the whole urban road networks [14]. On the other hand, how to acquire high quality and timely updated map to ensure the accuracy of the matches has been a tough problem. Therefore, even if we can obtain a large amount of traffic information from the FCD, the prediction of traffic speed with the grid model using the FCD is widely recognized a critical problem to be solved [15]. To fill these gaps, the traffic grid model has been proposed in this paper to match the trajectories to road networks more effectively. The traffic grid model has been applied in the traffic field, traffic state evaluation [10], traffic jam control [16], traffic light control [17], traffic forecasting [18]. However, it should not be ignored that the traditional grid model has two defects. When using the grid model to mine the spatiotemporal patterns, it is difficult to process the problem of grid boundary [19], especially the trajectories point near the grid boundary [20]. Furthermore, the size of the traditional grid is fixed [9], which limits the extraction of multiscale road features. Therefore, this paper proposed a multiscale-grid model, which includes two methods (i.e. grid fusion and grid combination). Firstly, the grid fusion method transforms the trajectories into the grid sequences to solve the grid boundary problem. Second, the grid combination method generates various scale datasets and adopted to prediction models.

Overall, the existing traffic speed prediction methods can be divided into two categories: model-calculation based methods and data-driven methods. Model-calculation based methods are criticized for theoretical assumptions and computed with small amounts of empirical data [21]. The model-calculation based methods have offered valuable insights on traffic speed prediction, such as the statistical model (e.g. Autoregressive Integrated Moving Average (ARIMA)) [22], and traffic simulation model (e.g. cellular automata model) [23]. However, most of the model-calculation methods are limited by the lack of a great deal of actual data, insufficient computing resources and certain ideal assumptions [24]. Moreover, it is unreasonable for model-calculation based methods to predict large-scale road network speeds with massive traffic data rapidly. Due to the complexity of the real traffic scenarios, it is insufficient to predict traffic speed by simulation and statistical analysis. Different from model-calculation based methods, the data-driven methods have relaxed assumptions for inputs and the methods with unfixed structure and parameters, which are more capable of processing outliers, missing data, and noisy data [25]. For instance, Support Vector Machine (SVM) [26], Kalman filter (KF) [27] and artificial neural networks (ANN) [28] are typical data-driven methods, which have been successfully applied to traffic speed forecasting. Although the aforementioned machine learning models have made some achievements, the non-parametric based methods show poor performance in learning the spatial-temporal features of traffic speed [29].

With the development of Intelligent Transportation Systems, the short-term traffic speed predictions are shifting from traditional machine learning methods to deep learning algorithms [30]. The traffic speed prediction methods using deep learning algorithms can capture spatial-temporal correlations of the traffic flow. From the perspective of temporal features, the Recurrent Neural Networks (RNNs) contribute to the dynamic traffic evolution with the time series data [31]. Although RNN is adopted in capturing time series of traffic flow and speed, traditional RNN cannot capture the long-term dependence of the input sequence [3]. In order to overcome these shortcomings, long short-term memory (LSTM) was developed to predict short-term travel speed [32]. By comparison with traditional RNN, LSTM has more advantages in dealing with long time-series data, which enable to determine the optimal time lag for prediction speed automatically. Recent years, LSTM is becoming more and more popular in traffic forecasting, such as traffic flow prediction [33], short-term traffic prediction [34], traffic speed prediction [35] etc. One of the famous LSTM variants is the Gated Recurrent Unit (GRU) model, which has fewer neurons but can achieve the same or better performance than LSTM [36]. However, neither LSTM models nor GRU model can capture the spatial feature of traffic speed. In terms of spatial correlations, convolutional neural networks (CNNs) have been used to capture adjacent relations for spatial characteristics [37]. However, CNN cannot capture the temporal correlation, while LSTM fails to characterize the spatial correlation. The convolution LSTM (Conv-LSTM) was proposed to capture the spatial-temporal correlation of traffic flow [38]. Although Conv-LSTM has good prediction accuracy, it usually needs a long training time [39].

Consequently, to reduce the computational complexity, a novel multiscale-grid-based model has been proposed to extract the spatial features from large-scale networks. Meanwhile, to extract complete, sequential information about traffic speed temporal feature, the Bidirectional Gated Recurrent Unit (BiGRU) network is proposed to predict traffic speed. As the GRU variants, the concept of BiGRU comes from Bidirectional Recurrent Neural Networks (BiRNN) [40], in terms of two separate memory blocks processing sequence data

in both forward and backward directions, producing more accurate prediction results. Although the BiGRU had been applied in quite a few traffic fields, like the lane change manoeuvre [41], travel time prediction [42] and traffic volume prediction [43], multiscale-grid-based traffic speed forecasting need further study. Furthermore, since the shallow single-layer BiGRU only captures short-term memories [44], the stacked Bidirectional Gated Recurrent Unit (SBiGRU) model is proposed to long-term time series prediction. Later, we combine the multiscale-grid model with SBiGRU to propose a novel framework to predict the short-term traffic speed.

Compared with the existing literature, the proposed method has the following contributions:

First, it is completely based on the data, without the aid of any additional tools such as complex map-matching and digital maps. This feature makes the proposed method practical and accessible for researchers and engineers who are not familiar with the map-matching and GIS techniques.

Second, under the multiscale grid modelling, the road networks are transformed into discrete cells, and the FCD data are matched to the grids via a simple assignment process to extract spatial patterns, where the computation load is largely reduced compared to map-matching and Geographic Information System (GIS) techniques. The multiscale traffic grid combination can better reflect the topological structure of the road. It not only solves the grid boundary problem but can generate various scale datasets to feed to prediction models. The simplicity of the grid model and the parameter extraction, along with the constant input of the GPS data, makes the proposed method highly efficient in the traffic speed prediction.

Third, the novel SBiGRU with multiscale-grid-base model characterizes the spatial-temporal properties of the spatial-temporal predictors, capturing the spatial-temporal features of traffic speed in the large-scale road network for the short-term traffic speed prediction.

Fourth, this method is easily transferred to other cities for the prediction of traffic state.

Using the expressway of Beijing as a case study, the accuracy and stability of the proposed method are demonstrated. The rest of this paper is organized as follows. Section 2 describes the FCD data and details the proposed methodology. Section 3 conducts a case study and in-depth analysis of the experimental results. Finally, Section 4 ends with major conclusions and discussions for future research.

II. METHODOLOGY

A. DATA CLEANING

The obtained FCD were sampled at a rate of 1 minute (min), which saved in datasets, and their variables and attributes are listed in Table 1.

There may be some error data for original GPS, due to either blockage of GPS signals and hardware/software bugs during the data collection and the transmission process [45]. In order to provide good quality of data and ensure the accuracy of the derived results, it is important to detect and remove the wrong records that are incompatible with the physical phenomena of traffic. The data cleaning process is carried out according to the following two steps.

TABLE 1. The detailed data attributes of the FCD system.

Characteristic	Field Name	Field Type	Field Description
id	Terminal ID	String	6 bytes characters, marking each vehicle
t	GPS timestamp	Timestamp	Accurate to second
lng	Longitude	Floating	Accurate to six decimal places
lat	Latitude	Floating	Accurate to six decimal places
v	Vehicle speed	Integer	Kilometer per hour

Step1: Remove the data that are beyond the range of the traffic analysis region.

Step2: Remove the data in which the corresponding vehicle speeds are out of the range of 0-120 km/h.

B. THE METHOD FRAMEWORK

Figure 1 shows the framework of the prediction method that we applied in this study. Through constructing the grid model, the trajectories would be added a new attribute (i.e. the grid number). With the grid fusion and combination, the grid size and grid number might be changed at the same time. Afterwards, based on the different scales of grids, the corresponding time-spatial parameters of traffic speed were extracted,

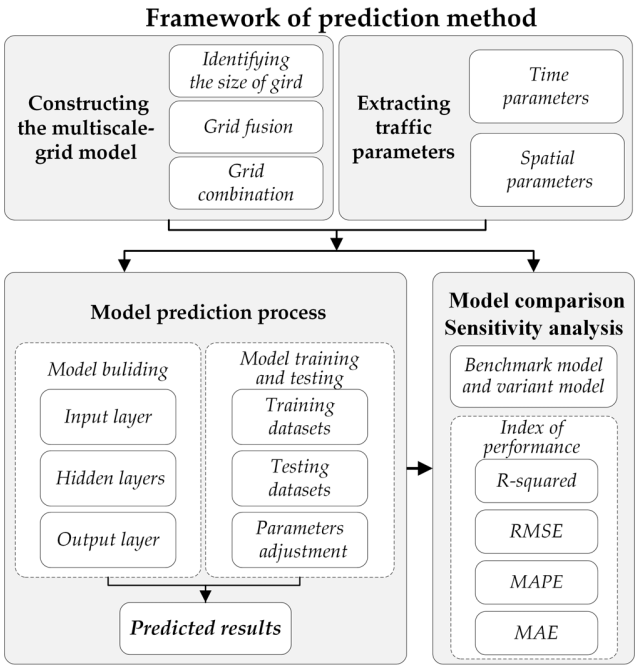
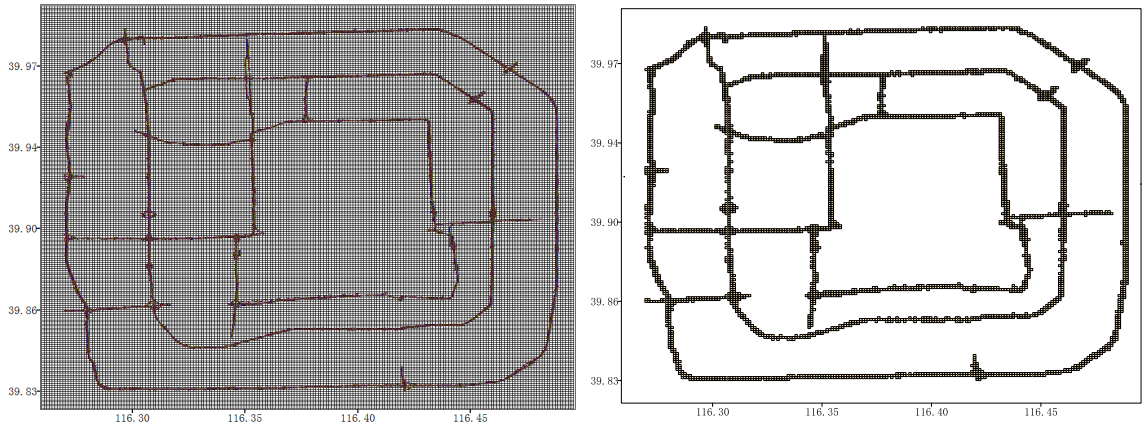


FIGURE 1. The flowchart of the prediction method.



(a) Grid model (b) Grid extracted

FIGURE 2. The FCD mapping to grid.

and the training datasets and testing datasets were made up, which were preparing for the model prediction.

C. THE MULTISCALE GRID MODEL

The traffic grid model is based on the City Management Grid Modelling Theory [46]. In this technique, a grid-wide network is established to transform roads into discrete grids; afterwards, the FCD will be matched to the grids in terms of a simple assignment process [14]. This method avoids the complexity and time-consuming calculation of map-matching algorithms as well as the needs of a detailed and timely updated map, achieving a considerable improvement in the efficiency of traffic prediction.

The multiscale grid model not only has the advantages of the traditional grid model but can capture road speed features of different scales more accurately and flexibly. In the multiscale grid model, a novel grid fusion method is proposed to solve the problem of GPS points deviation when the trajectory points match the grid. Furthermore, it can dynamically combine grids to predict road speeds of different scales, which overcomes the fixity of the traditional grid model.

1) IDENTIFYING THE SIZE OF GRID AND MATCHING THE FCD TRAJECTORIES TO GRIDS

In the current study, the area of each road is divided into grids with a fixed size. In constructing such grids, one crucial parameter needs to be specified: the size of each grid. If the size of the grid is too large, it may contain more parallel freeways in the same direction, and the traffic conditions between different roads cannot be differentiated. On the contrary, if the grid size is too small and cannot cover a freeway adequately, the FCD samples would be insufficient for some grids of the road, leading to the calculations of travel speed and vehicle positions inaccurate.

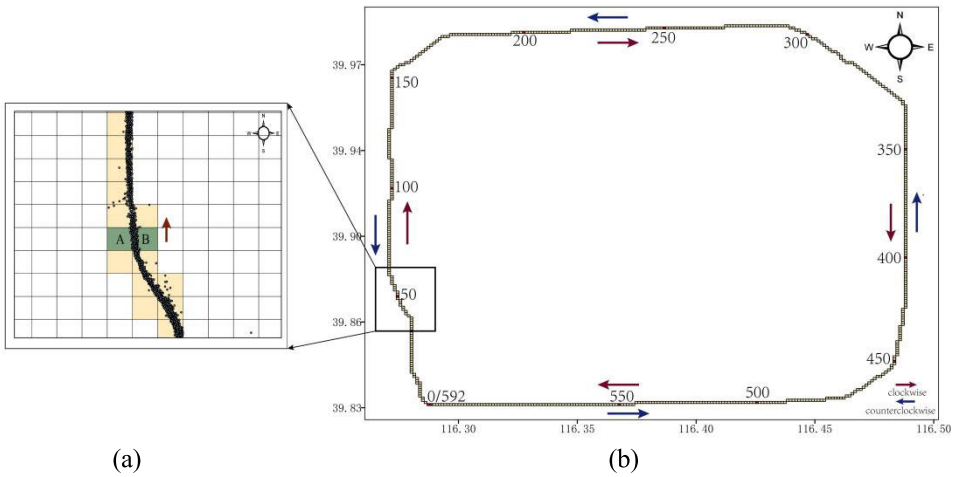
Considering the above factors, we empirically estimate the size of the grid $100 \times 100\ m^2$, reconstructing the freeway network. By taking the urban expressway in Beijing, China, as a case study, the trajectories of FCD will be represented by mapping to grids. As shown in Figure 2(a), the selected region (i.e. $20 \times 20\ km^2$) is split into $200 \times 200\ grid^2$ (i.e. each grid is $100 \times 100\ m^2$). Then we only need to extract the grid to match the trajectories point (see Figure 2(b)). In addition, to reduce the computational complexity, the extracted grids are renumbered.

2) GRID FUSION

When the freeway mapping to grids, how to define the relation of the downstream and upstream of the freeway has become the point of discussion; He *et al.* (2017) argued that it could select the grid with more GPS points (2017). However, due to the fixed size grid and the freeway network's structure irregular, sometimes the FCD on the urban expressway cannot match one certain grid adequately. As shown in Figure 3, the trajectories of vehicles are almost parallel to the intersection boundary of grid A and grid B; thus it is not adequate to estimate the traffic state, only by choosing either grid A or grid B. Therefore, it is necessary to merge the grids which located on the same longitudes or the same latitudes to estimate the same traffic state adequately. The process of the grid fusion is equivalent to the process of map matching dealing with anomalous trajectory points, which is based on trajectory without affecting other roads. Based on the above method, Ring 4 of Beijing is expressed with 592 grids and renumbered the 592 grids. The clockwise (i.e. CW) direction is grid number ordering from small to large. However, the counterclockwise (i.e. CCW) is opposite.

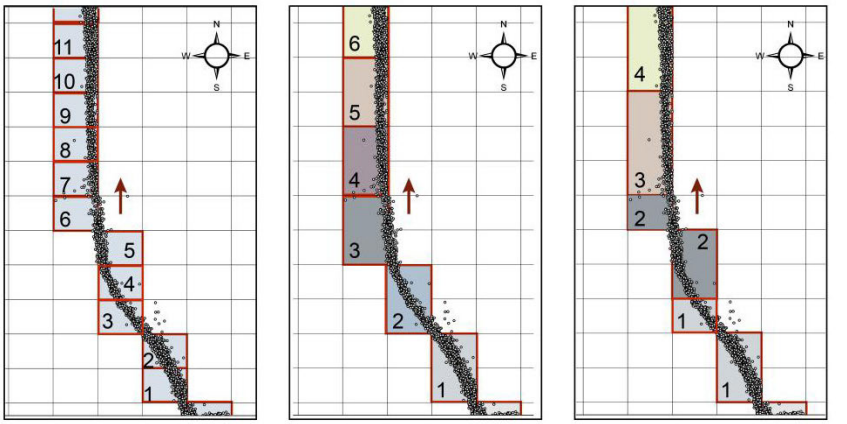
3) GRID COMBINATION

Differences from grid fusion, the target of the grid combination can improve the prediction accuracy in different grid combination scales. As shown in Figure 4, based on the multiscale-grid-combination method, the roads are matched with various scale grid-combination, which is divided into the road sections of different segmentation. Through renumbered



(a) (b)

FIGURE 3. The grid fusion and distinguish the direction.



(a) non-combination (b) two grids combination (c) third grids combination

FIGURE 4. The grid combination.

the combinations, a grid chain is made up, clearly. It can be seen that the range of grid numbers decreased with the increase of grid-scale combinations in the same study region. Additionally, it indicated that the dimension of FCD datasets decreases with enlarging grid-scale combinations in the same datasets. Finally, the multiscale datasets are constructed to prepare for prediction. Taking 592 grids of Beijing Ring 4 road as an example, provided that we combine two grids (see Figure 4(b)), and the dimensions of datasets are halved (i.e. 296 grids). Similarly, it will calculate various predictive accuracy by combining three grids, four grids or even more grids; which grid combined method is the best, requiring further discussion.

D. EXTRACTING TRAFFIC PARAMETERS

Traffic speed prediction at one location normally uses a sequence of speed values with n historical time steps as the input data, which can be extracted by floating car data. However, the speed may be influenced by the velocities of nearby locations, otherwise even locations far away, especially when traffic jam propagates through the traffic network [47]. Therefore, for the prediction model, we cannot only take into account the temporal features but also the spatial features. To analyze these spatial feature parameters' influence, the proposed multiscale grid model in this study does consider not only the spatial position of the grid in the road network but also the upstream and downstream traffic speed of different directions. In temporal feature parameters, when predicting the speed of large scale road network, the impact of historical average speed and historical median speed should not be neglected, which can extract adequately historical speed trend. Additionally, the different time slices in the grids and peak hours also require being considered. Besides, three categories of weather variables are considered, including temperature (measured by Celsius degree), weather state, wind speed (measured by mile per hour). In consequence, we selected eleven parameters: the upstream average speed U_t^s, the downstream average speed D_t^s, the time

TABLE 2. The parameter definition.

Parameter notation	Definition
TS_t^s	Time slice of time interval t at grid s
T	Length of the historical traffic speed sequence
N_t^s	The amount of track points of time interval t at grid s
v_t^s	Speed of track point of time interval t at grid s
U_t^s	Upstream average speed of time interval t at grid s
D_t^s	Downstream average speed of time interval t at grid s
GN_t^s	Grid number of time interval t
M_{t+1}^s	Historical median speed of time interval t+1 at grid s
H_{t+1}^s	Historical average speed of time interval t+1 at grid s
V_t^s	Average speed of time interval t at grid s
P_t^s	Peak hour (morning 1; evening 2; off-peak 0)
C_t^s	Weather state (sunny 0, rain 1, cloudy 2)
LT_t^s	Lowest temperature
HT_t^s	Highest temperature
W_t^s	Wind Speed (measured by a mile per hour)

slice TS_t^s, the grid number GN_t^s, the historical median speed M_{t+1}^s, the historical average speed H_{t+1}^s, average speed V_t^s, the peak hour P_t^s, the weather state C_t^s, wind speed W_t^s, the lowest temperature LT_t^s and the highest temperature HT_t^s (see the Table. 2) to predict the speed in future time series (i.e. $t + 1$, $t + 2$, $t + 3$), respectively. According to the grid-wide network, we extracted the twelve variables (i.e. TS_t^s, GN_t^s, U_t^s, D_t^s, M_{t+1}^s, H_{t+1}^s, V_t^s, P_t^s, C_t^s, LT_t^s, HT_t^s, W_t^s) of multivariate time series in different slices (e.g. 5 min, 10min, 20min, 30min, etc), respectively.

The definition for the Average speed of time interval t at grid s is given by:

$$V_t^s = \frac{\sum_{i=1}^{N} v_t^s}{N_t^s}, \quad s = 0, 1, 2, \ldots, S, \ t = 0, 1, 2, \ldots, T \quad (1)$$

For the CW direction, the $V_t^{s\pm1}$ is calculated as s-1; by contrast, in the CCW direction it is calculated as $s + 1$. When the grid $s = 0$, the grid s of U_t^s is selected as the maximum number S. The definition for the upstream average speed of time interval t at grid s is given by:

$$U_t^s = \frac{V_t^{s\pm1}}{N_t^{s\pm1}}, \quad s = 0, 1, 2, \ldots, S, \ t = 0, 1, 2, \ldots, T \quad (2)$$

For the CW direction, the $V_t^{s\pm1}$ is calculated as $s + 1$; by contrast, in the CCW direction it is calculated as s-1. When the grid s selected the maximum number S, the grid s of D_t^s is the minimum number. The definition for the downstream average speed of time interval t at grid s is given by:

$$D_t^s = \frac{V_t^{s\mp1}}{N_t^{s\mp1}}, \quad s = 0, 1, 2, \ldots, S, \ t = 0, 1, 2, \ldots, T \quad (3)$$

Because the numerical value of GN_t^s has an influence on the calculation of weight matrix, one-hot encoding is adopted to avoid the influence of numerical value on the weight matrix.

Suppose the traffic network consisting of grid s, with twelve variables, and we require predicting the traffic speeds over ($t + 1$, $t + 2$, $t + 3$, etc.) time series. The input can be characterized as a data matrix (i.e. X_T^s), and the output can be expressed as a vector (i.e. Y_T^s), (4), as shown at the bottom of the next page.

The predicted target value is defined as:

$$Y_T^s = \left[y_{t-n}^s, y_{t-n+1}^s \cdots y_{t-1}^s, y_t^s\right]^T \quad (5)$$

Therefore, based on the data matrix $[X_T^s, Y_T^s]$, it is further split into the training dataset (i.e. $[X_T^s, Y_T^s]_{tr}$, whose proportion is 70%) for training and the test dataset (i.e. $[X_T^s, Y_T^s]_{te}$, whose proportion is 30%) for testing.

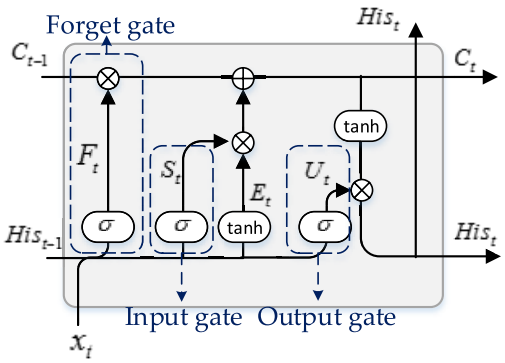
To improve the effectiveness of the optimal solution by eliminating, the data normalization is proposed; this method uses the proportion of calculation to make the dataset fall into the specific range and removes the limitation of data units by converting the dataset to no pure dimensional data for the model training and testing. Additionally, in accordance with excluding the effects of large eigenvalues, normalization can improve the training speed. The minimum-maximum method (see Equation (6)) is one of the major normalized methods.

$$X_T' = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (6)$$

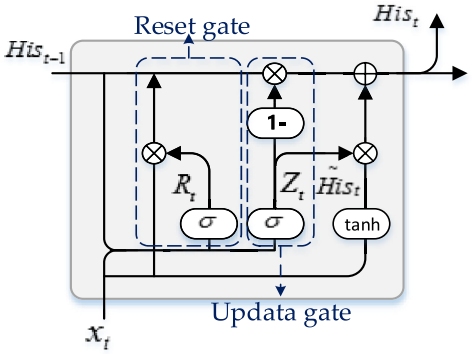
E. LSTM AND LSTM VARIANTS MODEL

Similar to other structures of a machine learning model for the traffic speed forecast, LSTM and GRU both contain two stages: the training stage and the prediction stage. In the course of the training stage, by enhancing the memory of iterative information with the threshold layers, these two structures are more conducive to calculate the biases, weights and other parameters, which could identify and store historical speed information, reducing the training time of the model. During the speed prediction stage, the predictive speed of the time series can be obtained by performing a vector operation on the input data based on the training model [48].

The architectures of the LSTM and GRU are seemingly similar, which consists of three layers: the input layer, the memory blocks and the output layer. As shown in Figure 5, in the structure of memory blocks, the function σ (i.e. Sigmoid function whose scale is [0, 1]) can control the information maintained in each cell of the memory block,



(a) LSTM



(b) GRU

FIGURE 5. The structure of memory blocks of LSTM and GRU.

effectively allowing memory blocks to selectively decide what to keep or drop from the memory, called "gate". Therefore, the data flow can be represented as follows: the data from the input layer is passed to the memory blocks, and the memory blocks are updated with values based on the gates. Eventually, the output layers are calculated by combining the memory with the previous state of the network as well as the current input.

However, there are significant differences in the structure of memory blocks, compared with the LSTM layer being made up of input gates S_t, forget gates F_t and output gates U_t (see Figure 5(a)) the GRU layer is composed of two gates(see Figure 4(b)): reset gate R_t and update gate Z_t (see Figure 5(b)), contributing to fewer parameters and faster convergence [49]. It is noted that the update gate is used to control the extent to which the state information of the previous moment is brought into the current state. The larger the value of the update gate, the more the state information of the previous moment is brought in. The reset gate controls how much information is conveyed to the current candidate set $\tilde{His}_t$ in the previous state. The smaller the reset gate is; the less information is passed to the previous state.

The LSTM layer transition equations are the following:
Gates:

$$F_t = \sigma\left(W_f * [His_{t-1}, x_t] + B_f\right) \tag{7}$$
$$S_t = \sigma\left(W_s \cdot [His_{t-1}, x_t] + B_s\right) \tag{8}$$
$$E_t = \sigma\left(W_e \cdot [His_{t-1}, x_t] + B_e\right) \tag{9}$$

Input transform:

$$C_t = F_t * C_{t-1} + E_t * S_t \tag{10}$$

Memory update:

$$U_t = \sigma\left(W_U \cdot [His_{t-1}, x_t] + B_U\right) \tag{11}$$
$$His_t = U_t * \tanh\left(C_t\right) \tag{12}$$

The GRU layer transition equations are the following:
Gates:

$$R_t = \sigma\left(W_r * [His_{t-1}, x_t]\right) \tag{13}$$
$$Z_t = \sigma\left(W_z * [His_{t-1}, x_t]\right) \tag{14}$$

Memory update:

$$\tilde{His}_t = \tanh\left(W_{\tilde{His}} \cdot [R_t * His_{t-1}, x_t]\right) \tag{15}$$
$$His_t = (1 - Z_t) * His_{t-1} + Z_t * \tilde{His}_t \tag{16}$$
$$y_t = \sigma(W_y \cdot His_t) \tag{17}$$

Activation functions:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{18}$$
$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{19}$$

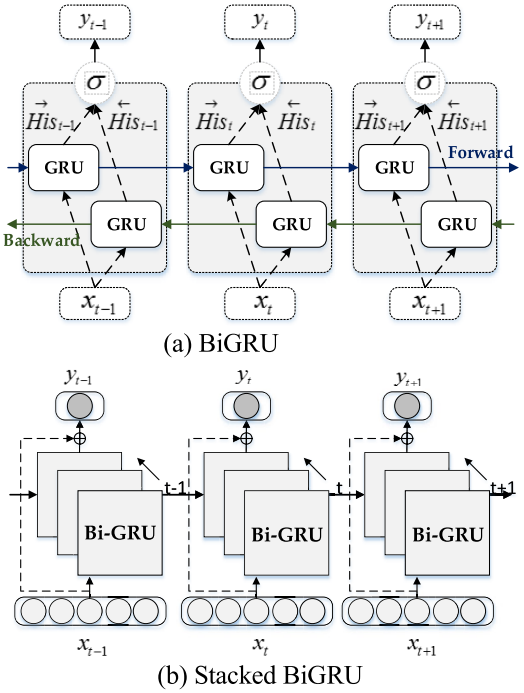
where the operator [ ] denotes that two vectors are connected; the operator $*$ denotes the product of matrices; x_t is the input vector at the time step t; W_f, W_s, W_e, W_r, W_z, $W_{\tilde{His}}$ present the weights; B_f, B_s B_e present the bias vectors; σ is the Sigmoid function; tanh is non-linear activation functions;

To better solve the problem of exploding/vanishing gradient and make the deep networks converge fast, the Rectified Linear Unit (ReLU) is proposed [50].

$$\text{ReLu} = \begin{cases} x, & x > 0 \\ 0, & x < 0 \end{cases} \tag{20}$$

$$X_T^s = \begin{bmatrix} TS_{t-n}^s, & GN_{t-n}^s, & U_{t-n}^s, & D_{t-n}^s, & M_{t-n+1}^s, & H_{t-n+1}^s, & V_{t-n}^s, & P_{t-n}^s, & W_{t-n}^s, & C_{t-n}^s, & LT_{t-n}^s, & HT_{t-n}^s \\ TS_{t-n+1}^s, & GN_{t-n+1}^s, & U_{t-n+1}^s, & D_{t-n+1}^s, & M_{t-n+2}^s, & H_{t-n+2}^s, & V_{t-n+1}^s, & P_{t-n+1}^s, & W_{t-n+1}^s, & C_{t-n+1}^s, & LT_{t-n+1}^s, & HT_{t-n+1}^s \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ TS_{t-1}^s, & GN_{t-1}^s, & U_{t-1}^s, & D_{t-1}^s, & M_{t-1}^s, & H_{t-1}^s, & V_{t-1}^s, & P_{t-1}^s, & W_{t-1}^s, & C_{t-1}^s, & LT_{t-1}^s, & HT_{t-1}^s \\ TS_t^s, & GN_t^s, & U_t^s, & D_t^s, & M_t^s, & H_t^s, & V_t^s, & P_t^s, & W_t^s, & C_t^s, & LT_t^s, & HT_t^s \end{bmatrix} \tag{4}$$

As the GRU variants, through two separate memory blocks, BiGRU can process sequence data in both forward and backward directions (see Figure 6(a)), which ensures that it can better deal with large-scale data. Furthermore, compared with BiGRU, the SBiGRU has more hidden layers (see Figure 6(b)), which can capture long-term memories more accurately to carry out the traffic speed forecasting for the various grid scales. It is noted that the number of hidden layers of the SBiGRU model has a significant influence on prediction accuracy. It is generally acknowledged that the deeper neural network layers, the better robustness and accuracy are [51]. Nevertheless, too many hidden layers will increase model complexity, which may easily lead to overfitting. Therefore, to measure the influence of the number of hidden layers, the various multilayer model will be tested in this experiment.



(a) BiGRU

(b) Stacked BiGRU

FIGURE 6. The structure of BiGRU and Stacked BiGRU.

F. THE MULTISCALE GRID MODEL WITH THE SBIGRU NETWORK

In this section, a novel architecture (i.e. the combining of the multiscale grid model with the SBiGRU network) is proposed for the large scale traffic speed prediction (see Figure 7). In the preparation stage, it can be seen that utilizing the multiscale grid combination method extracts spatial characteristics of traffic speed to build different datasets as input variables. In the prediction stage, we establish a double-layer memories blocks to extract the long-term memories and short-term memories of traffic speed in different grid combinations. The first BiGRU layer is used to learning the spatial-temporal characteristics and present to the second hidden layer to predict the traffic speed in the large scale road network. In the course of the training procedure of the proposed model, the training object is minimizing the root mean squared error (RMSE) between the estimated values and ground-truth values. The weight and bias can be learned through the training process. As shown in Table 3, the training process can be divided into two phases, including the first phase for extracting the spatial features by the multiscale grid model and the second phase for learning the parameters.

TABLE 3. Pseudo-code of the process to train the BiGRU.

Algorithm1: The Training process of the stacked Bi-GRU network

Input: The historical FCD dataset, Time lag: φ , Grid combination: s

Output: The stacked BiGRU model with learnt parameters

Phase 1: Feature extraction using multiscale grid model

  for time intervals t = 1 to T do

    for all grid combinations, s = 0 to S do

      Calculate $TS_t^s, GN_t^s, U_t^s, D_t^s, M_{t+1}^s, H_{t+1}^s, V_t^s$

    end for

  end for

Sort the variable importance by random forest and stored in $[X_T^s, Y_T^s]$

End Phase 1

Phase 2: Procedure BiGRU Training

  Initialize a null set: R←Ø

  for all available time intervals, t = 1 to T do

  $x_T^S = [TS_t^s, GN_t^s, U_t^s, D_t^s, M_{t+1}^s, H_{t+1}^s, V_t^s]$

  for all available time intervals, $t = \varphi$ to T do

  $X_T^S = [x_{t-1}, x_{t-2}, \ldots, x_{t-\varphi}]$

  A training observation is put into R

  end for

  Initialize all the weighted and intercept parameters

  repeat

  Randomly extract a batch of samples R_s from R

  Estimate the parameters by minimizing the objective function RMSE within R_s

  until convergence criterion met

End Phase 2

G. THE INDEX OF PERFORMANCE

To compare the performance of our proposed model with other state-of-the-art methods, measurement is required to evaluate the prediction performance of models. In the traffic speed prediction field, the Mean Absolute Error (MAE), the Root-Mean-Square Error (RMSE), mean absolute percentage error (MAPE) and R-square (R^2) are used to evaluate the benefits and drawbacks of this model. The definitions are as follows:

$$MAE(Y_i, \hat{Y_i}) = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y_i}| \tag{21}$$

$$RMSE(Y_i, \hat{Y_i}) = \left[\frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y_i}|^2\right]^{\frac{1}{2}} \tag{22}$$

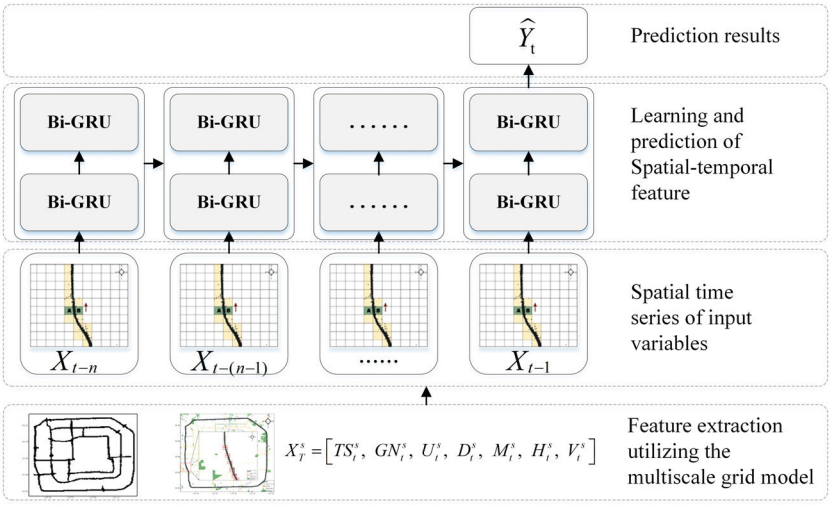


FIGURE 7. The framework of the multiscale grid model with the stacked BiGRU network.

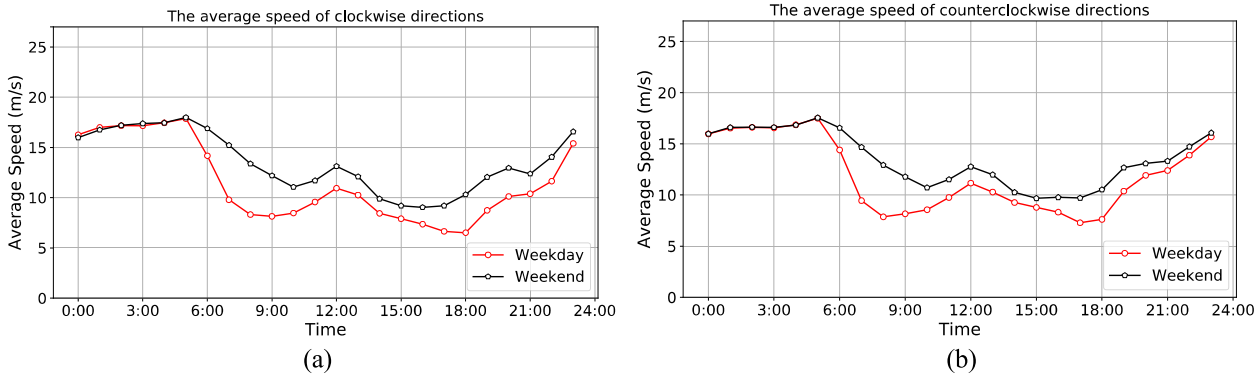


FIGURE 8. The average speed of ring 4 of Beijing.

$$MAPE(Y_i, \hat{Y}_i) = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i - \hat{Y}_i|}{Y_i} \tag{23}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \overline{Y}_i)^2} \tag{24}$$

where n is the number of test samples, Y_i is the real traffic speed, $\hat{Y}_i$ is the traffic speed to be predicted. $\overline{Y}_i$ is the average speed.

III. RESULT ANALYSIS AND COMPARISON

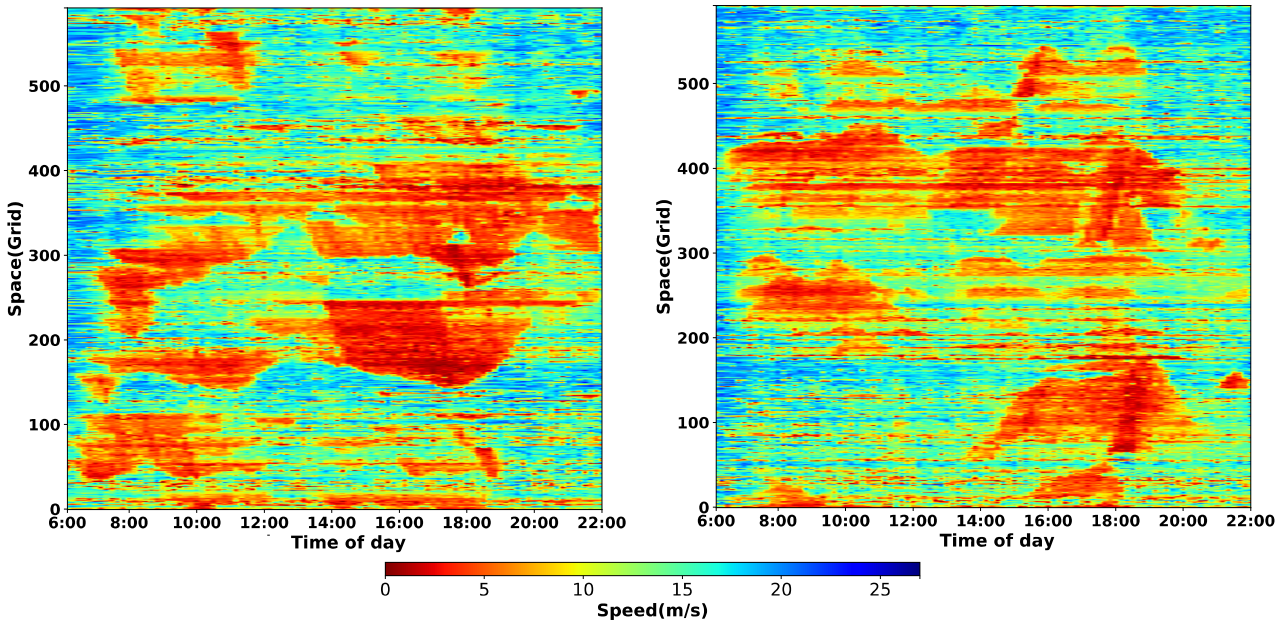
A. DATA DESCRIPTION AND EXPERIMENTAL SETUP

1) DATA DESCRIPTION

In order to prove the accuracy, stability, and validity of our prediction model, the FCD data provided by the Beijing Taxi company are utilized as the dataset. The Beijing Taxi company provided 40-day FCD ranged from May 1 to July 31, 2016, involving FCD in Wednesday, Friday (a total of 26 weekdays) and Sunday (a total of 14 weekends), which include 16931629 floating cars' trajectories (i.e. 28.7GB in size) with the longitude and latitude range as <116.25, 116.50> and <39.83, 39.99>.

Based on the multiscale grid model, ring 4 of Beijing is mapped with the 592 fixed girds. Then, the trajectories traveling along the CW and CCW direction of ring 4 of Beijing were extracted, respectively. To intuitively show the whole pattern of speed changes, Figure 8 shows the trends of the average speed of CW and CCW direction in both weekday and weekend, respectively. It can be seen that the trend of average speed fluctuation is the constant in the evening (0:00-5:00) whether on the weekday or weekend. At the same time, in the daytime (5:00-23:00) the average speed of the weekend is faster than the weekday in general. Therefore, when exploring the change of speed, it tends to choose the daytime (6:00-22:00).

To see more detail of the speed fluctuation, the spatial-temporal distribution of average speed was presented, where the redder colour, the slower the speed. As shown in Figure 9, it is noted that in the morning peak hour (7:00-10:00) the gridlocks are mainly concentrated below the grid 300 in the CW



(a) the CW direction (b) the CCW directions

FIGURE 9. Spatial-temporal distribution of speed.

direction; nevertheless, the gridlocks of the CCW direction are opposite, which located above grid 240. Moreover, in the evening peak hour (17:00-19:00), between 180 and 250 grids, unlike the CCW, there are gridlocks in the CW direction. It is worth mentioning that there are gridlocks around grid 400 in both morning and evening peaks. It can be concluded that there are some tidal phenomena below the grid 300. Furthermore, the gridlock in the CW direction has a clearer boundary, suggesting that the change of speed trend is more regular than CCW.

2) FEATURE SELECTION

According to the feature selection, the few most important variables or parameters will be identified, and the new subset is input into the prediction models, which can reduce spatial-temporal complexity of predictor variables, result in less computation, fewer parameters, and save the cost of observing the feature [52]. The random forest (RF) algorithm is adopted to calculate the importance of the variables. In the RF algorithm, variable importance can be obtained by calculating the out of bag error (OOB error) and Gini coefficient. OOB error has stronger generalization ability, even if there are continuous variables and categorical variables, the accuracy of OOB error will not be affected. Therefore, according to OOB error, Figure 10 shows the variable importance.

From Figure 10, it can be observed that the seven categories of spatial-temporary variables, including average speed, upstream average speed, downstream average speed, historical average speed, historical median speed, the time slice and grid number, are the dominating factors, and the

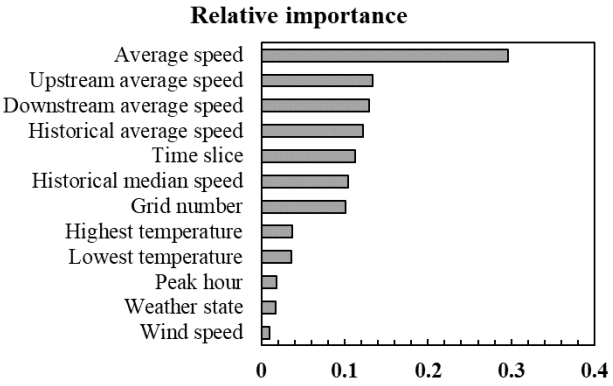


FIGURE 10. Variable importance ranking by the random forest.

most important is the average speed. However, other variables, such as highest temperature, peak hour, weather have few contributions (less than 5%) to the prediction. In this paper, by considering the trade-off between the computational efficiency and predictive performance, we select seven categories of variables: the average speed, the upstream average speed, the downstream average speed, the historical average speed, the historical median speed, the time slice and the grid number.

3) PARAMETER SETTINGS

According to the data description, the dataset is split into four parts: the CW dataset and the CCW dataset in the weekend or weekday. Later, the dataset for different estimation time steps is built. Citing the 10 min time interval (i.e.144 time

slices per day) with non-combination (i.e.592 grids) in weekdays (26 days) as an example, the dimension matrix of $[X_T^s, Y_T^s]_{10\,\mathrm{min}}$ is [592 × 144 × 26, 8]. According to the rule of the dataset classification, the training dataset $[X_T^s, Y_T^s]_{tr}$ is [592 × 144 × 26 × 70%, 8], the test dataset $[X_T^s, Y_T^s]_{te}$ is [592 × 144 × 26 × 30%, 8].

Based on the minimum-maximum normalization method, the training dataset and the test dataset are normalized. In the proposed model, we need to consider the following hyperparameters: the dimension of hidden units, and the dimension of the mini-batches, and the epochs. For the activation function, the Rectified Linear Unit (*ReLu*) is proposed, which can solve the problem of exploding/vanishing gradient better. In order to prevent over-fitting, the dropout is set to 0.5. Meanwhile, the Adam algorithm is used to update the parameters of the neural network. The hyper-parameter settings are shown in Table 4.

TABLE 4. Hyper-parameter settings.

Hyperparameter	Values
Hidden units	200
Mini-batches	64
Epoch	200
Dropout	0.5
Activation function	ReLu
Loss function	RMSE
Optimizer	Adam

B. EXPERIMENTAL RESULTS

1) MODEL COMPARISONS

In this section, the proposed model SBiGRU and benchmark models are trained on the training set and validated on the test set, respectively. In the benchmark models, we considered four benchmark methods, which includes traditional time-series prediction model (i.e. ARIMA), data-driven methods (i.e. SVM), deep learning approaches (i.e., LSTM, GRU) and the stacked deep learning models LSTM-2L (i.e. LSTM with two hidden layers) and GRU-2L (i.e. GRU with two hidden layers). For the ARIMA model, based on the best AIC (Akaike Information Criterion), the optimal model is obtained as ARIMA (3, 1, 1). In the parameter selection of the SVM model, the kernel function is set as radial basis function (RBF), and the penalty coefficient ''C'', and the parameter ''Gamma'' were determined by the cross validation. For the stacked deep learning models, in order to reduce model complexity and avoid over-fitting problems, the deep learning model with two hidden layers (i.e. LSTM-2L, GRU-2L) is set as the benchmark models. At the same time, for the fair comparison, the BiGRU-2L (i.e. BiGRU with two hidden layers) is selected as the target model in the SBiGRU models. To ensure fairness, the aforementioned benchmark algorithms have the same input features (the same category and the time interval). The multiscale-grid data is set to dataset uncombined (i.e. 592 grids are uncombined). The time lag is set to 10 min, and the hyperparameters are set the same as the proposed model. Then we use the RMSE, R^2, and MAE to measure the total predictive accuracy of fitting in the whole test data, and use MAPE to measure the models' predictive performance.

The experiment platform is server with 32 CPU cores (Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.1GHz), 64G RAM, and GPU (NVIDIA GeForce RTX 2080). The experiment utilizes python 3.6.1 with scikit-learn [53], tensorflow [54], keras [55] on Windows 10 for comparing the models.

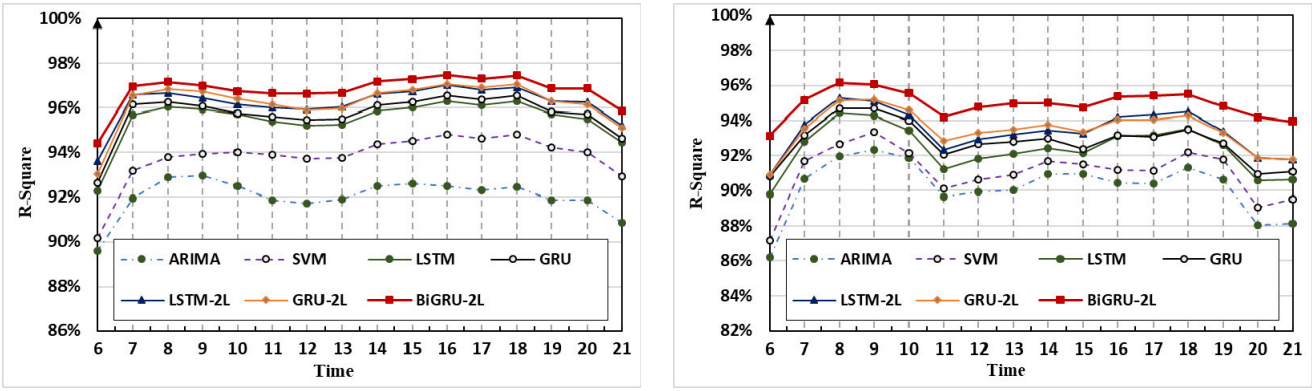
Table 5 shows the comparison of the predictive performance of the BiGRU-2L and the benchmark models in different directions. It can be seen that the deep learning model work better than the ARIMA and SVM. Moreover, the BiGRU-2L outperforms the benchmark model in the four measurements of predictive performance. The BiGRU-2L achieves the best predictive performance measured, which outperforms the ARIMA with the improvement of 6.04%, 0.97, 7.47% and 1.96 on R^2, MAE, MAPE and RMSE. The reason is that the BiGRU-2L has a significant advantage in the training process with input sequences in forward and backward directions and summarizing the temporal information from past and future contexts.

TABLE 5. The predictive performance comparison with full grid combination.

Model	CW direction				CCW direction			
	R^2(%)	MAE	MAPE	RMSE	R^2(%)	MAE	MAPE	RMSE
ARIMA	89.92%	2.01	16.94%	3.23	89.88%	2.98	19.24%	3.96
SVM	91.14%	1.61	16.33%	2.87	90.96%	2.74	18.68%	3.31
LSTM	94.44%	1.36	12.24%	1.62	93.19%	2.04	16.35%	2.43
GRU	94.48%	1.35	12.10%	1.61	93.40%	1.98	16.21%	2.34
LSTM-2L	95.49%	1.30	11.33%	1.56	93.54%	1.84	15.43%	2.22
GRU-2L	95.50%	1.14	9.87%	1.42	94.01%	1.82	14.82%	2.20

Figure 11 compares the R-Square values of different methods in the direction of CW and CCW. As shown in Figure 11(a) and (b), the BiGRU-2L model reveals the best performance among the models in terms of the R-Square values. In addition, it can be found that the forecasting accuracy in CW direction is better than that in CCW direction. The reason is that the number of exit and entrance in the CW direction is fewer, less impact of speed, the change of speed trend in the CW direction is more regular, and has higher prediction precision. It is also noticed that there is the inflection point of R-square in CCW direction in 11:00-12:00. This can be attributed to the fact that the speed fluctuates greatly during this period (see Figure 8 and Figure 9), which has a certain influence on prediction accuracy.

Figure 12 gives the overall prediction errors produced by these different methods on both weekends and weekdays. It is found that the prediction errors in the weekend are smaller than that in weekday. As shown in Figure 12(a) and (b), the BiGRU-2L reveals better performance than other models



(a) the CW direction

(b) the CCW direction

FIGURE 11. R-Square values of different model.

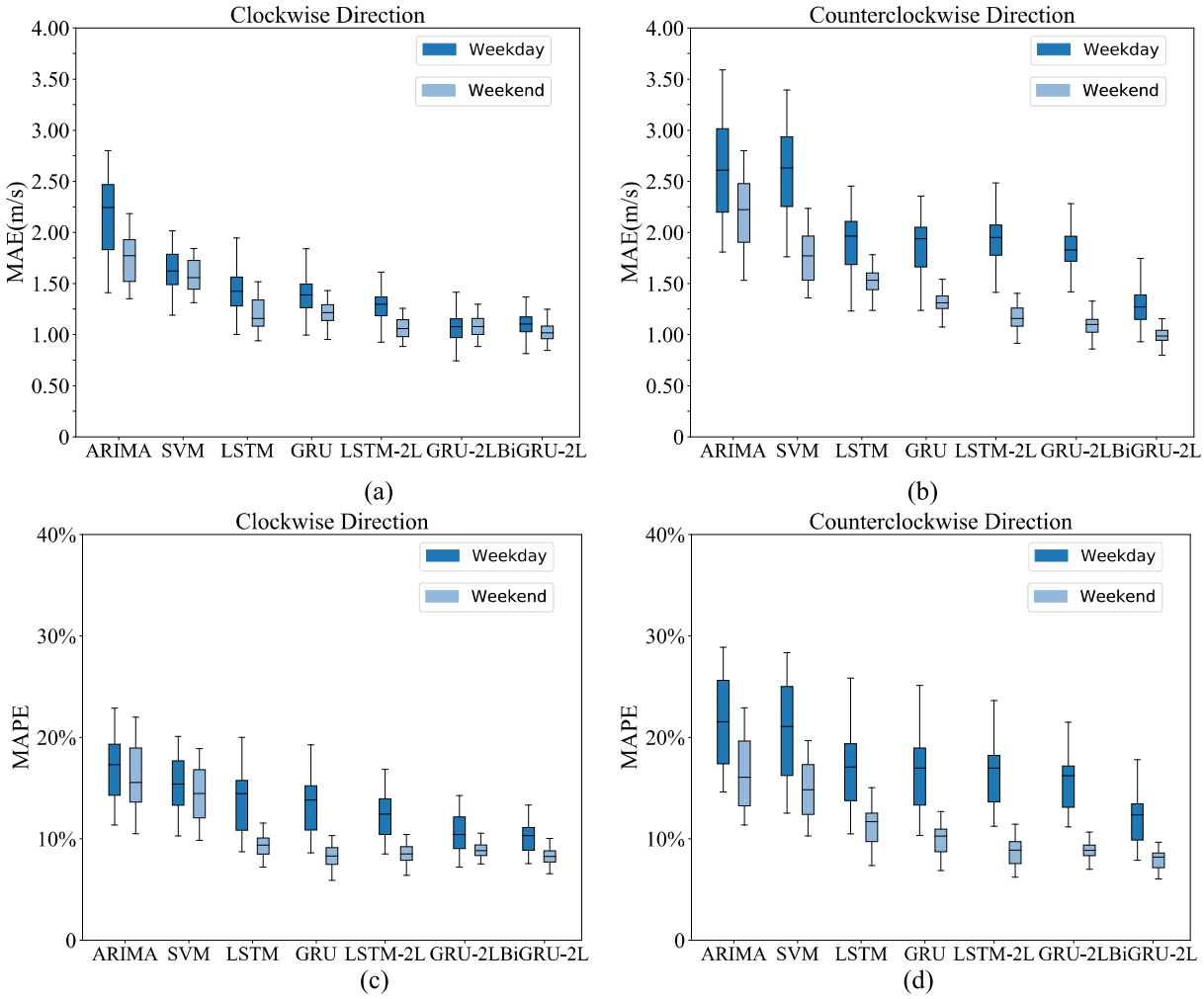


FIGURE 12. Comparison of the prediction errors for each method.

in terms of the maximum, minimum and median of errors in both CW and CCW direction. Furthermore, it can be seen that the BiGRU-2L model has a smaller interquartile range, and the error distribution is more concentrated than those of other models. Moreover, the MAPEs of CW and CCW direction of the BiGRU-2L are lower than others (see Figure 12(c) and (d)), indicating that the BiGRU-2L is more accurate and stable.

Figure 13 demonstrates the comparison of seven models: ARIMA, SVM, LSTM, GRU, LSMT-2L, GRU-2L,

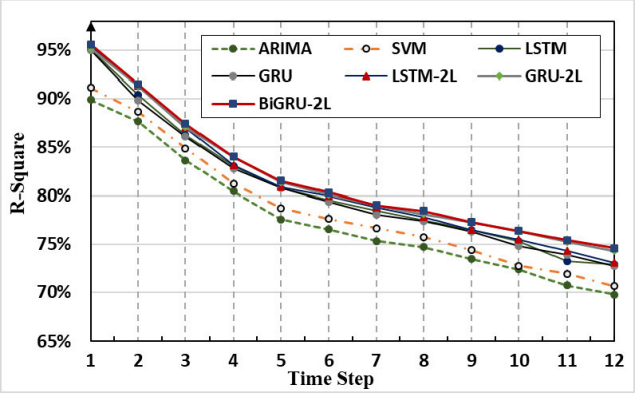


FIGURE 13. Comparison of different methods in terms of various time steps.

BiGRU-2L in terms of various time steps in the CW direction. As shown in Figure 13, the ARIMA and SVM performed worst, and the one-layer model (i.e. LSTM, GRU) have similar predictive performance, but they are inferior to the stacked models (i.e. LSTM-2L, GRU-2L, BiGRU-2L). Among the deep learning models, the BiGRU-2L is capable of providing reliable prediction whose index of performance R^2 is better than the deep learning models. Meanwhile, with the increase of time steps, the R^2 of BiGRU-2L is the slowest reduction. The R^2 of BiGRU-2L decreases by 21.98% from 1 to 12 time steps, which is less than the LSTM, GRU, LSTM-2L and GRU-2L (i.e. 23.34%, 23.41%, 23.21%, 22.12%). Furthermore, the average R^2 (i.e. 81.786%) of 12 time steps of BiGRU-2L is higher than the other six models (i.e. 77.682%, 78.695%, 80.748%, 80.589%, 81.111%, 81.622%).

Figure 14 shows some samples of heats maps of the ground-true speed in different directions of the fourth ring road and forecasting results of one-time step by BiGRU-2L, where the deeper color area means traffic congestion. The spatial distribution of low-speed area (i.e. traffic congestion area) is illustrated by the heats maps, where the fluctuation of speed is great (i.e. from the free flow to congested traffic flow). In addition, from the samples of visualization, we can

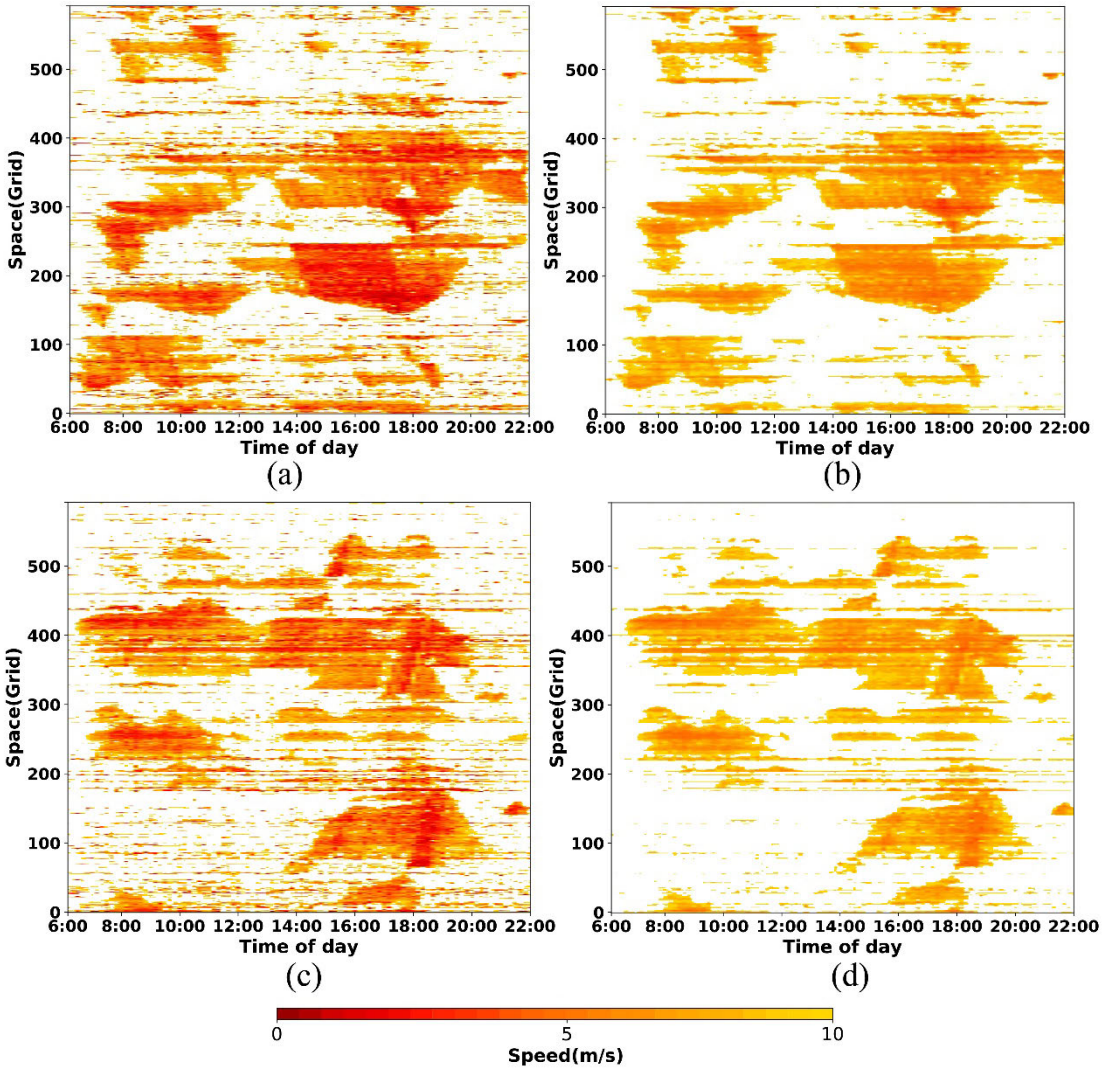
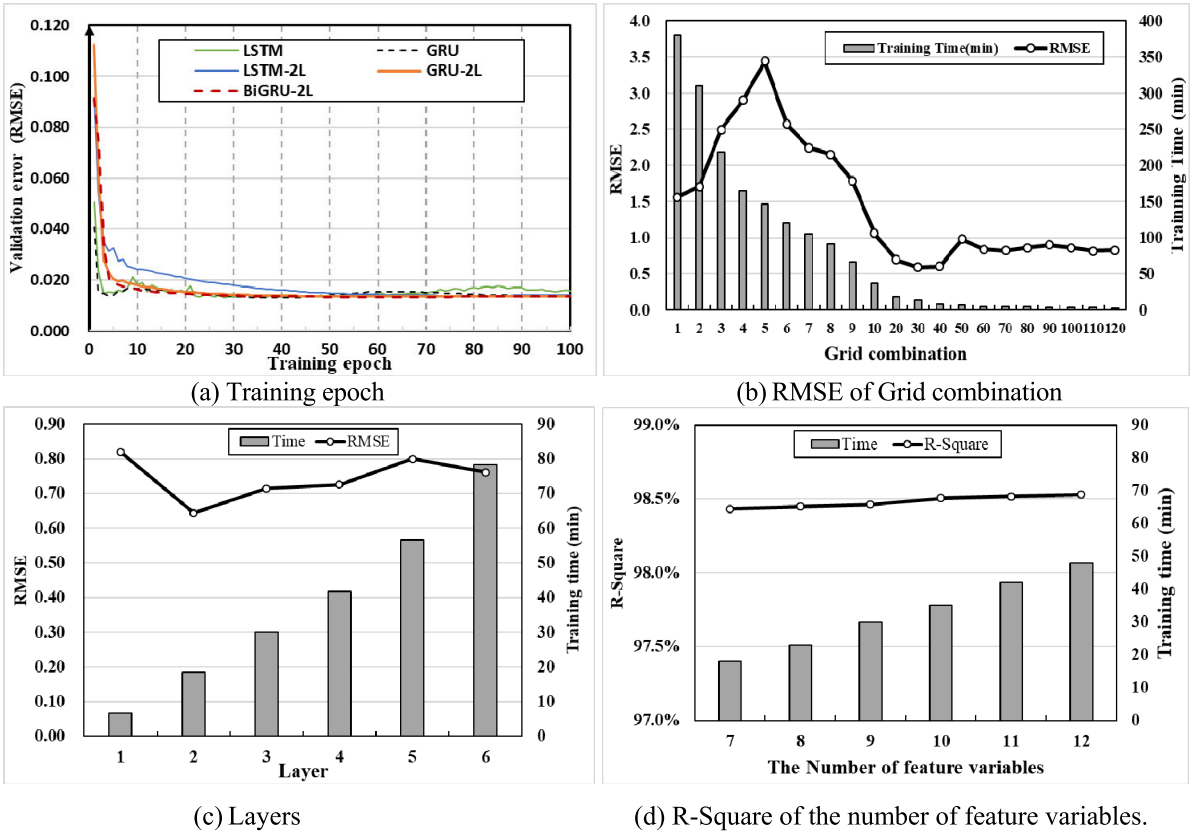


FIGURE 14. Comparison the ground-true speed and the predicted speed by BiGRU-2L: (a) the ground-true speed in the CW direction; (b) the predicted speed in the CW direction; (c) the ground-true speed in the CCW direction; (d) the predicted speed in the CCW direction.



(a) Training epoch

(b) RMSE of Grid combination

(c) Layers

(d) R-Square of the number of feature variables.

FIGURE 15. Parameter tuning and sensitivity analysis of BiGRU model.

find that the BiGRU-2L with multiscale-grids can primarily capture the spatial-temporal characteristics of traffic speed, and make an accurate prediction. The combination of traffic speed prediction and visualization can make great progress for the traffic operators to detect and forecast grid with traffic jams and design proactive strategies to avoid the congested status of roads effectively.

2) SENSITIVITY ANALYSIS

In this section, this paper conducts the sensitivity analysis and parameter tuning on Bi-GRU, where we investigated four kinds of parameters, including some samples of prediction, the type of training epochs, grid combination, hidden layers and the number of feature variables.

Figure 15(a) shows the validation error after each training epoch is recorded for BiGRU-2L and the benchmark models. It can be seen that the RMSE of BiGRU drops rapidly and decreases gradually, which indicates that the convergence speed of BiGRU-2L model is high. Furthermore, compared with other benchmark models, the process of train epoch of BiGRU-2L changed comparatively smooth and steadily. To conclude, when high-efficient computation resources are not reachable, it can reduce the number of training epochs to make predictive performance slightly sacrificed, and it is an acceptable way.

Figure 15(b) shows the RMSE values and training time of the different grid combination prediction results. For the grid combination method, we focus on two methods: two-grids combination method and thirty-grids combination method. The two-grids combination method has a special advantage to cover spatial road features more detailed, which does not only ensure the prediction accuracy but also avoid the issue of data missing in one grid combined method. However, the two-grids combination takes more training time (i.e. 310 min). Compared with two-grids combination, the thirty-grids combination method has the lowest value of RMSE and less time-consuming. This phenomenon indicates that the spatial characteristics of 30 grids (i.e. the road length is 3 km) are more reasonable and more suitable for road operation characteristics.

Figure 15(c) and d demonstrate that we train the BiGRU-2L with 30 grids combined method under different layers and variables, respectively, with 100 training epochs. It can be observed that with the increase of the hidden layers, the training time also increases correspondingly. Although the single-layer model needs the least training time (i.e. 7 min), it has the worst prediction performance (i.e. RMSE is 0.82). Compared with the single-layer model, the six-layer model has better prediction performance, but it takes the longest training time (i.e. 78 min). Furthermore, the RMSE values present a down and up trend with the increase

of layers, whereas the lowest point is the two-layer model, whose phenomenon is caused by overfitting issues. Therefore, the BiGRU-2L model can ensure the prediction accuracy (i.e. the RMSE is 0.64), reduce the time consumption, and avoid the overfitting problem.

Figure 15(d) shows the influence of the number of variables on the prediction performance and the training time. It intuitively reveals that the BiGRU-2L with 12 variables takes 48 min to train 100 epochs, which seems to be computationally expensive. Although the prediction performance improved slightly with the increase of the selected feature variables, the training time increased significantly. As the Figure 10 analysis, the number of feature variables refers to seven, with the predictive performance slightly sacrificed (i.e. 0.097% loss only), less training time consuming (i.e. decreased by 166%), under the case where high-efficient computational resources are not reachable.

IV. DISCUSSION AND CONCLUSION

In this paper, a deep learning algorithm, named multiscale grid model with the stacked BiGRU network, is proposed for marvelous scale road network speed prediction. The proposed architecture is fused by a multiscale grid model and a two-layer BiGRU structure. With the multiscale grid model grid, we extract spatial variables and reconstruct the input matrix of the deep learning framework. Subsequently, the proposed deep learning architecture (i.e. BiGRU-2L) is established for capturing temporal properties of traffic speed, and the results are promising.

The traditional data-collecting devices cover only fixed points of the traffic network, such as the traffic microwave sensors, which are impossible to produce a wide range of information about travel speed within the network [24], [56]. In comparison, based on the massive numbers of FCD that are highly spatial-temporal detailed, the speed of large-scale road network can be accurately predicted. As for the spatial feature attribute extraction, Yu *et al.* (2016) transformed the network-wide traffic speed into a series of static images and captured the spatial characteristics of speed by Deep Convolutional Neural Networks (i.e. DCNN) [29]. However, DCNN could not process the data of Non Euclidean Structure, which can't extract directional feature of speed and cost a great quantity of time for training [57]. In this study, though the multiscale grid model, the directional feature of speed was extracted with less time consuming, namely, CW and CCW direction of Beijing Fourth Ring Road. Furthermore, with the multiscale grid model, the various grid combinations have the special purposes, and the thirty grids combined method is most accurate, which can establish the real-time traffic controlling strategies for traffic managers to avoid the congested. In addition, the two grids combined method can cover spatial road features more detailed preparing for Vehicle Navigation System.

At the stage of traffic speed prediction, the feature selection process can help proposed model reduce training time with a less loss in the predictive performance (measured by RMSE), which is consistent with the studies of Jintao Ke *et al.* (2017) [50]. The experimental result illustrates that the BiGRU-2L outperforms the benchmark algorithms (i.e. LSTM, GRU, LSMT-2L, GRU-2L) in the measurements of RMSE, R^2, MAE, and MAPE in large samples, indicating that the proposed model performs better at capturing the spatial-temporal characteristics for the short-term traffic speed forecasting. Particularly, for the multi-timestep speed prediction, the BiGRU-2L achieves more accurate results than other models.

Future work will concentrate on exploring the novel deep learning structure based on the LSTM variants and introducing a novel unfixed grid model to extract the spatial-temporal features to improve the accuracy of the model. Meanwhile, we will also adopt Conv-LSTM model to predict the multiscale-grid-based speed. Furthermore, further study on the performance comparison between grid model method and map matching method. A dataset with longer temporal dimensions and broader spatial dimensions could be employed to train and test the proposed model. In addition, the proposed model can be applied to other cities for traffic speed prediction.

ACKNOWLEDGMENT

The authors would like to thank the Institute of Beijing Taxi Company for providing the data in this study.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest in any aspect of the data collection, analysis, or the funding received regarding the publication of this paper.

REFERENCES

[1] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: Where we are and where we're going," *Transp. Res. C, Emerg. Technol.*, vol. 43, pp. 3–19, Jun. 2014.

[2] F. Batool and S. A. Khan, "Traffic estimation and real time prediction using ADHOC networks," in *Proc. IEEE Symp. Emerg. Technol. (ICET)*, 2005, pp. 264–269.

[3] 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.

[4] Z. Abbas, A. Al-Shishtawy, S. Girdzijauskas, and V. Vlassov, "Short-term traffic prediction using long Short-term memory neural networks," in *Proc. IEEE Int. Congr. Big Data (BigData Congress)*, vol. 1, Jul. 2018, pp. 57–65.

[5] R. Bauza and J. Gozalvez, "Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications," *J. Netw. Comput. Appl.*, vol. 36, no. 5, pp. 1295–1307, Sep. 2013.

[6] X. Kong, F. Xia, Z. Ning, A. Rahim, Y. Cai, Z. Gao, and J. Ma, "Mobility dataset generation for vehicular social networks based on floating car data," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 3874–3886, May 2018.

[7] M. Rahmani, H. N. Koutsopoulos, and A. Ranganathan, "Requirements and potential of GPS-based floating car data for traffic management: Stockholm case study," in *Proc. 13th Int. IEEE Conf. Intell. Transp. Syst.*, Sep. 2010, pp. 730–735.

[8] H. Bar-Gera, "Evaluation of a cellular phone-based system for measurements of traffic speeds and travel times: A case study from Israel," *Transp. Res. C, Emerg. Technol.*, vol. 15, no. 6, pp. 380–391, Dec. 2007.

[9] Y. Liu, X. Yan, Y. Wang, Z. Yang, and J. Wu, "Grid mapping for spatial pattern analyses of recurrent urban traffic congestion based on taxi GPS sensing data," *Sustainability*, vol. 9, no. 4, p. 533, Mar. 2017.

[10] D. Chen, X. Yan, F. Liu, X. Liu, L. Wang, and J. Zhang, "Evaluating and diagnosing road intersection operation performance using floating car data," *Sensors*, vol. 19, no. 10, pp. 1–20, 2019.
[11] S. Kim and J.-H. Kim, "Adaptive fuzzy-network-based C-measure map-matching algorithm for car navigation system," *IEEE Trans. Ind. Electron.*, vol. 48, no. 2, pp. 432–441, Apr. 2001.
[12] C. Chen, Y. Ding, X. Xie, S. Zhang, Z. Wang, and L. Feng, "TrajCompressor: An online Map-matching-based trajectory compression framework leveraging vehicle heading direction and change," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 2012–2028, May 2020.
[13] M. Quddus and S. Washington, "Shortest path and vehicle trajectory aided map-matching for low frequency GPS data," *Transp. Res. C, Emerg. Technol.*, vol. 55, pp. 328–339, Jun. 2015.
[14] Z. He, L. Zheng, P. Chen, and W. Guan, "Mapping to cells: A simple method to extract traffic dynamics from probe vehicle data," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 3, pp. 252–267, 2017.
[15] M. Gao, T. Zhu, X. Wan, and Q. Wang, "Analysis of travel time patterns in urban using taxi GPS data," in *Proc. IEEE Int. Conf. Green Comput. Commun. IEEE Internet Things IEEE Cyber, Phys. Social Comput.*, Aug. 2013, pp. 512–517.
[16] J. Long, Z. Gao, P. Orenstein, and H. Ren, "Control strategies for dispersing incident-based traffic jams in two-way grid networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 469–481, Jun. 2012.
[17] T. Tan, F. Bao, Y. Deng, A. Jin, Q. Dai, and J. Wang, "Cooperative deep reinforcement learning for large-scale traffic grid signal control," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2687–2700, Jun. 2020.
[18] C.-W. Huang, C.-T. Chiang, and Q. Li, "A study of deep learning networks on mobile traffic forecasting," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–6.
[19] J. Du Chung, O. H. Paek, J. W. Lee, and K. H. Ryu, "Temporal pattern mining of moving objects for location-based service BT—Database and expert systems applications," Tech. Rep., vol. 2, pp. 331–340.
[20] Y. Chen, P. Yuan, M. Qiu, and D. Pi, "An indoor trajectory frequent pattern mining algorithm based on vague grid sequence," *Expert Syst. Appl.*, vol. 118, pp. 614–624, Mar. 2019.
[21] K. Y. Chan, T. S. Dillon, J. Singh, and E. Chang, "Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and Levenberg–Marquardt algorithm," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 644–654, Jun. 2012.
[22] C. Chen, J. Hu, Q. Meng, and Y. Zhang, "Short-time traffic flow prediction with ARIMA-GARCH model," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 607–612.
[23] W. Hu, L. Yan, and H. Wang, "Traffic jams prediction method based on two-dimension cellular automata model," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 2023–2028.
[24] Y. Gu, W. Lu, L. Qin, M. Li, and Z. Shao, "Short-term prediction of lane-level traffic speeds: A fusion deep learning model," *Transp. Res. C, Emerg. Technol.*, vol. 106, pp. 1–16, Jun. 2019.
[25] 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.
[26] J. Wang and Q. Shi, "Short-term traffic speed forecasting hybrid model based on Chaos–Wavelet analysis-support vector machine theory," *Transp. Res. C, Emerg. Technol.*, vol. 27, pp. 219–232, Feb. 2013.
[27] S. Zhang, Y. Song, D. Jiang, T. Zhou, and J. Qin, "Noise-identified Kalman filter for short-term traffic flow forecasting," in *Proc. 15th Int. Conf. Mobile Ad-Hoc Sensor Netw. (MSN)*, Dec. 2019, pp. 462–466.
[28] A. Raza and M. Zhong, "Hybrid lane-based short-term urban traffic speed forecasting: A genetic approach," in *Proc. 4th Int. Conf. Transp. Inf. Saf. (ICTIS)*, Aug. 2017, pp. 271–279.
[29] H. Yu, Z. Wu, S. Wang, Y. Wang, and X. Ma, "Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks," *Sensors*, vol. 17, no. 7, pp. 1–16, 2017.
[30] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 166–180, Jan. 2018.
[31] H. Van Lint and C. Van Hinsbergen, "Short-term traffic and travel time prediction models," *Artif. Intell. Appl. Crit. Transp. Issues*, vol. 22, no. 1, pp. 22–41, 2012.
[32] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[33] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *Proc. 31st Youth Acad. Annu. Conf. Chin. Assoc. Autom. (YAC)*, Nov. 2016, pp. 324–328.
[34] Y. Tian and L. Pan, "Predicting short-term traffic flow by long short-term memory recurrent neural network," in *Proc. IEEE Int. Conf. Smart City/SocialCom/SustainCom (SmartCity)*, Dec. 2015, pp. 153–158.
[35] J. Wang, R. Chen, and Z. He, "Traffic speed prediction for urban transportation network: A path based deep learning approach," *Transp. Res. C, Emerg. Technol.*, vol. 100, pp. 372–385, Feb. 2019.
[36] K. Zhang, L. Wu, Z. Zhu, and J. Deng, "A multitask learning model for traffic flow and speed forecasting," *IEEE Access*, vol. 8, pp. 80707–80715, 2020.
[37] G. Dai, C. Ma, and X. Xu, "Short-term traffic flow prediction method for urban road sections based on Space–Time analysis and GRU," *IEEE Access*, vol. 7, pp. 143025–143035, 2019.
[38] D. Ma, B. Sheng, S. Jin, X. Ma, and P. Gao, "Short-term traffic flow forecasting by selecting appropriate predictions based on pattern matching," *IEEE Access*, vol. 6, pp. 75629–75638, 2018.
[39] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with conv-LSTM," in *Proc. 9th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2017, pp. 1–6.
[40] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[41] X. Liu, J. Liang, and B. Xu, "A deep learning method for lane changing situation assessment and decision making," *IEEE Access*, vol. 7, pp. 133749–133759, 2019.
[42] J. Zhao, Y. Gao, Y. Qu, H. Yin, Y. Liu, and H. Sun, "Travel time prediction: Based on gated recurrent unit method and data fusion," *IEEE Access*, vol. 6, pp. 70463–70472, 2018.
[43] P. Wang, W. Xu, Y. Jin, J. Wang, L. Li, Q. Lu, and G. Wang, "Forecasting traffic volume at a designated cross-section location on a freeway from large-regional toll collection data," *IEEE Access*, vol. 7, pp. 9057–9070, 2019.
[44] A. Karpathy, J. Johnson, and L. Feifei, "Visualizing and understanding recurrent networks," 2015, *arXiv:1506.02078*. [Online]. Available: https://arxiv.org/abs/1506.02078
[45] B. Deng, S. Denman, V. Zachariadis, and Y. Jin, "Estimating traffic delays and network speeds from low-frequency GPS taxis traces for urban transport modelling," *Eur. J. Transp. Infrastruct. Res.*, vol. 15, no. 4, pp. 639–661, 2015.
[46] X.-J. Zhao and J.-Y. Zhao, "Research on model of resource management for traffic grid," *Procedia Eng.*, vol. 15, pp. 1476–1480, Jan. 2011.
[47] Z. Cui, R. Ke, and Y. Wang, "Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction," pp. 1–12, 2018, *arXiv:1801.02143*. [Online]. Available: https://arxiv.org/abs/1801.02143
[48] W. Xiangxue, X. Lunhui, and C. Kaixun, "Data-driven short-term forecasting for urban road network traffic based on data processing and LSTM-RNN," *Arabian J. Sci. Eng.*, vol. 44, no. 4, pp. 3043–3060, Apr. 2019.
[49] B. Xiao, Y. Liu, and B. Xiao, "Accurate State-of-Charge estimation approach for lithium-ion batteries by gated recurrent unit with ensemble optimizer," *IEEE Access*, vol. 7, pp. 54192–54202, 2019.
[50] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 8609–8613.
[51] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4580–4584.
[52] J. Ke, H. Zheng, H. Yang, and X. Chen, "Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach," *Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 591–608, Oct. 2017.
[53] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[54] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467
[55] N. Ketkar, "Introduction to Keras BT," in *Deep Learning With Python: A Hands-On Introduction*, N. Ketkar, Ed. Berkeley, CA, USA: Apress, 2017, pp. 97–111.
[56] J. Zhang, Y. Zheng, D. Qi, R. Li, X. Yi, and T. Li, "Predicting citywide crowd flows using deep spatio-temporal residual networks," *Artif. Intell.*, vol. 259, pp. 147–166, Jun. 2018.

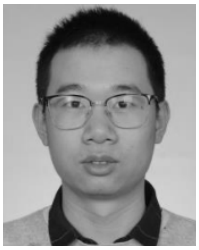
[57] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," 2017, *arXiv:1707.01926*. [Online]. Available: http://arxiv.org/abs/1707.01926

DEQI CHEN received the B.S. degree in transportation and planning engineering from Inner Mongolia University, China, in 2016. He is currently pursuing the Ph.D. degree with the MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, School of Traffic and Transportation, Beijing Jiaotong University, Beijing, China. His current research interests include data driven short-term traffic prediction and intelligent transportation systems.

XUEDONG YAN received the Ph.D. degree in civil engineering from the University of Central Florida, Orlando, FL, USA. He is currently a Professor with the School of Traffic and Transportation, Beijing Jiaotong University, Beijing, China, where he is also the Executive Director of the MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport and the Vice President. His research interests include traffic safety, driving simulation, urban planning, and management.

XIAOBING LIU is currently pursuing the Ph.D. degree in traffic planning and management with Beijing Jiaotong University, China. He is also one member of the MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport in China. His current research interests include transportation energy, transportation policy and planning, and traffic big data analysis. He has conducted some data-based research in sharing mobility, especially the management and optimization of on-demand ridesharing services.

SHURONG LI received the B.S. degree in transportation and planning engineering from the Beijing Institute of Technology, China, in 2016. She is currently pursuing the Ph.D. degree with the MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, School of Traffic and Transportation, Beijing Jiaotong University, Beijing, China. Her current research interests include adaptive traffic signal control and automatic vehicle trajectory planning.



LIWEI WANG received the B.S. degree in traffic and transportation from Beijing Jiaotong University, Beijing, China, in 2016, where he is currently pursuing the Ph.D. degree with the MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, School of Traffic and Transportation. His current research interests include traffic congestion identification and intelligent transportation systems.

XINMEI TIAN received the B.S. degree in traffic engineering from the Xi'an University of Architecture and Technology, China, in 2018. She is currently pursuing the M.Sc. degree with the MOT Key Laboratory of Transport Industry of Big Data Application Technologies for Comprehensive Transport, School of Traffic and Transportation, Beijing Jiaotong University, Beijing, China. Her current research interests include arterial traffic signal coordination control and intelligent transportation systems.