Hierarchical Traffic Flow Prediction Based on Spatial-Temporal Graph Convolutional Network

Hanqiu Wang, *Graduate Student Member, IEEE*, Rongqing Zhang, *Member, IEEE*, Xiang Cheng, *Fellow, IEEE*, and Liuqing Yang, *Fellow, IEEE*

***Abstract*— In recent years, traffic flow prediction has attracted more and more interest from both academia and industry since such information can provide effective guidance for traffic management or driving planning and enhance traffic safety and efficiency. But due to the complicated spatial-temporal dependence in actual roads and the limitation of intersection monitoring equipment, there are still many challenges in spatial-temporal traffic flow prediction. In this paper, we propose a novel hierarchical traffic flow prediction protocol based on spatial-temporal graph convolutional network (ST-GCN), which incorporates both spatial and temporal dependence of intersection traffic to achieve a more accurate traffic flow prediction. Different from existing works, our proposed protocol with the Adjacent-Similar algorithm can also effectively predict the traffic flow of the intersections without historical data. Experiments based on practical traffic data of the city of Qingdao, China demonstrate that our proposed ST-GCN-based traffic flow prediction protocol outperforms the state-of-the-art baseline models. Moreover, as for the intersections without historical data, we can also obtain a good prediction accuracy.**

***Index Terms*— Graph convolutional network, intersection without historical data, spatial-temporal dependence, traffic flow prediction.**

I. INTRODUCTION

WITH the development of urbanization and vehicular technology, the number of urban vehicles is rapidly increasing in recent years. Traffic congestion has then become a big issue in daily urban transportation. Traffic forecasting, as an efficient means for transportation planning, traffic management, and traffic control, has attracted more and more interest from both academia and industry [1]. Accurate traffic flow prediction can provide effective guidance for traffic management or driving planning, and enhance the traffic safety and efficiency in Intelligent Transportation Systems (ITS) [2].

Manuscript received March 4, 2021; revised August 27, 2021 and November 17, 2021; accepted January 19, 2022. This work was supported in part by the Ministry National Key Research and Development Project under Grant 2020AAA0108101; in part by the National Natural Science Foundation of China under Grant 61901302 and Grant 62125101; in part by the Natural Science Foundation of Shanghai under Grant 20ZR1462400; in part by the Fundamental Research Funds for the Central Universities, China; in part by the State Key Laboratory of Advanced Optical Communication Systems Networks, China; and in part by the National Science Foundation under Grant CPS-2103256, Grant ECCS-2102312, and Grant ECCS-2128593. The Associate Editor for this article was S. Selpi. *(Corresponding author: Rongqing Zhang.)*

Hanqiu Wang and Rongqing Zhang are with the School of Software Engineering, Tongji University, Shanghai 200092, China (e-mail: 2031562@tongji.edu.cn; rongqingz@tongji.edu.cn).

Xiang Cheng is with the State Key Laboratory of Advanced Optical Communication Systems and Networks, School of Electronics, Peking University, Beijing 100871, China (e-mail: xiangcheng@pku.edu.cn).

Liuqing Yang is with the Department of Electrical and Computer Engineering, University of Minnesota, Twin Cities, Minneapolis, MN 55455 USA (e-mail: qingqing@umn.edu).

Digital Object Identifier 10.1109/TITS.2022.3148105

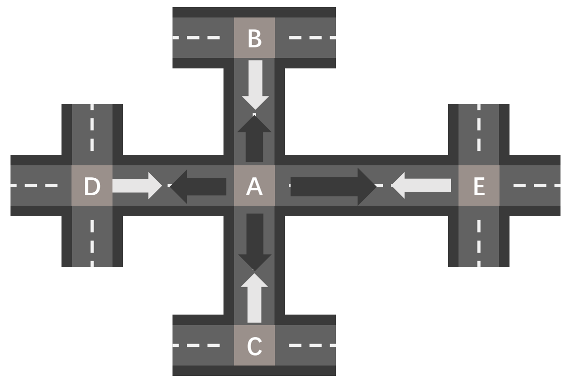


Fig. 1. Example of intersection map.

However, due to the complex spatial-temporal dependency in actual roads and the limitation of monitoring equipments, there are still many challenges in spatial-temporal traffic flow prediction.

In terms of spatial dependence, the topology of the city determines the interdependence among intersections. The traffic flow output at the upstream intersection directly affects the traffic input at the downstream intersection, and the traffic flow at the downstream intersection also feeds back to the upstream intersection. As shown in Fig. 1, the vehicles at the intersection A flow in four directions, respectively into the four adjacent intersections, while there are also vehicles in the corresponding four directions that merge into the intersection A. Therefore, the flow at the intersection A is closely related to the traffic at the intersections B, C, D, and E. Due to the mutual influence among intersections, they share similar traffic flow trends.

In terms of temporal dependence, the traffic flow changes dynamically with time, mainly as periodicity. For example, there may be a certain pattern in the flow of some specific time periods. The flow statistics of the two intersections on 8:00 am from September 1st to September 20th are illustrated in the Fig. 2. It can be seen that the overall trend of the 20-days traffic at the two intersections is similar. There may

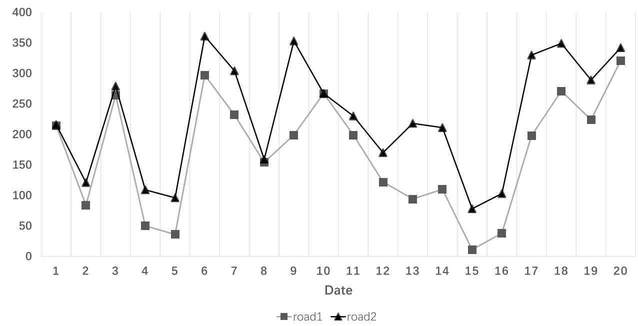


Fig. 2. Traffic flow at two intersections from 8:00 am on September 1 to 20.

be a time-dependent relationship between the traffic flow of current period and that of some previous periods.

In terms of intersection data, due to complicated geographic location and limited monitoring equipments, some intersections often fail to obtain historical traffic data. It is usually difficult to predict the flow data of no-data intersections.

Although many current works have contributed to traffic flow forecasting problems, there are still many challenges in spatial-temporal traffic flow prediction due to the complicated spatial-temporal dependence in actual roads and the limitation of intersection monitoring equipments. Therefore, in this paper, we propose a novel hierarchical traffic flow prediction protocol based on Spatial-Temporal Graph Convolutional Network (ST-GCN), which incorporates both spatial and temporal dependence of traffic flow to achieve a more accurate traffic flow prediction model. Moreover, our proposed protocol with the Adjacent-Similar algorithm can also effectively predict the traffic flow of the intersections without historical data, which has not been investigated in the literature. The main contributions of the paper can be summarized as follows:

- Different from existing works, we focus on a more challenging but practical traffic flow prediction scenario, which includes both the urban intersections with and without historical data. To the best knowledge of the authors, this is the first attempt to deal with the traffic flow prediction issue of intersections without historical data.
- In order to achieve effective traffic flow prediction for both intersections with and without historical data, we propose a novel ST-GCN-based hierarchical protocol for traffic flow prediction. The designed ST-GCN model is used to predict traffic flow of intersections with historical data, which exploits GCN to extract spatial dependence and GRU to extract temporal dependence. Moreover, we further provide an Adjacent-Similar algorithm to predict the intersections without historical data through their spatial-temporal correlations with the ones with historical data.
- We conduct various experiments based on practical traffic data of the city of Qingdao, China. The results have demonstrated that our proposed ST-GCN-based traffic flow prediction protocol outperforms the state-of-the-art baseline models. Moreover, as for the intersections without historical data, we can also obtain a good prediction accuracy. Our dataset and tensorflow implementation of ST-GCN are available at https://github.com/Wautumn/ST-GCN.

II. Related Works

Traffic flow prediction is an important topic in ITS and many achievements have been made in the literature. The methods of traffic flow prediction can be mainly divided into two categories: parametric models and non-parametric models.

The parametric models presuppose the regression function and simplify problem to a known function form. The main methods include Autoregressive Integrated Moving Average (ARIMA), Kalman filter, and so on. The ARIMA model needs stationary time series data and it can only capture linear relationship. In [3], the authors used different time-oriented temporal data to predict the traffic flow by the ARIMA model. The Kalman filter model predicts future traffic conditions based on the traffic state of the previous and current moments [4]. In [5], Gong *et al.* analyzed the influence factors of traffic volume based on grey entropy and selected the main influencing factors to establish the prediction model based on Kalman filter. These models are simple and fast, but they cannot take the spatial features of the road networks into consideration.

The non-parametric models are able to learn any function form freely from training data without strong assumptions, and thus they can fit most of the function forms and automatically learn statistical rules from traffic data. The main models include K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and so on. In [6], the authors proposed a three-stage framework based on KNN. This structure use information of related stations by introducing the distance metric. The main function of SVM is to get the optimal separating hyperplane such that the margin between the training data is maximum [7]. In [8], a model to predict the short-time traffic flow volume by the introduction of SVM with a time-dependent structure was proposed.

The development of neural networks provides a better solution for traffic flow forecasting. These models can achieve more complicated data modeling and higher prediction accuracy. In [9], traffic flows among adjacent road links in a transportation network were modeled as a Bayesian network. The joint probability distribution between the cause nodes (data utilized for forecasting) and the effect node (data to be forecasted) in a constructed Bayesian network was described as a Gaussian Mixture Model (GMM). Huang *et al.* [10] proposed a network architecture consisting of a Deep Belief Network (DBN) and verified that the network can capture random features from traffic data on multiple datasets. In [11], a deep-learning-based traffic flow prediction method was proposed, which considered the spatial and temporal correlations inherently. A stacked autoencoder model was used to learn generic traffic flow features. Long Short-Term Memory (LSTM) network is a deep learning approach which is capable of learning long-term dependencies [12]. Poonia *et al.* [13] applied the LSTM for momentary traffic stream forecast. In [14], a traffic forecast model based on LSTM network was proposed. The proposed LSTM network considered temporal-spatial correlation in the traffic system via a two-dimensional network which was composed of many memory units. Gated Recurrent Unit (GRU) is a variant of LSTM. GRU has only two gates: update gate and reset gate [15]. In [16], the authors combined the spatio-temporal analysis with GRU. In this model, time correlation analysis and spatial correlation analysis were performed on the collected traffic flow data, and the GRU was used to process the spatio-temporal feature information.

However, these networks are not well-performing in dealing with spatial dependence. Instead, Convolutional Neural Network (CNN) can deal with spatial dependence better. In [17], a ST-ResNet model to collectively forecast the inflow and outflow of crowds in each and every region of a city was proposed. This model employs the residual neural network framework to model the temporal closeness, period, and trend properties of crowd traffic and residual convolutional units to model the spatial properties of crowd traffic. In [18], the authors proposed a multitask deep-learning framework that can simultaneously predict the node flow and edge flow throughout a spatio-temporal network. In [19], the authors applied 3D CNNs to learn the spatio-temporal correlation features jointly from low-level to high-level layers for traffic data and designed an end-to-end structure, named as MST3D by multiple 3D CNNs.

Considering the superiority of CNN in dealing with the spatial dependence and the superiority of LSTM in dealing with the temporal dependence, recent studies combined these two models to exploit the spatial-temporal dependence of urban traffic for prediction. In [20], the authors proposed an end-to-end framework called DeepTransport, in which CNN and LSTM are utilized to obtain spatial-temporal traffic information within a transport network topology and an attention mechanism is introduced to align spatial and temporal information. Yao *et.al* [21] proposed a Spatial-Temporal Dynamic Network (STDN), in which a flow gating mechanism is introduced to learn the dynamic similarity between locations, and a periodically shifted attention mechanism is designed to handle long-term periodic temporal shifting.

Despite CNN makes great progress in traffic prediction tasks, CNN is essentially applicable to Euclidean spaces such as images and regular grids, and has certain limitations for complex topological traffic networks. In recent years, Graph Convolutional Network (GCN) has been proposed based on CNN, which solves the problem of non-Euclidean structure through the adjacency matrix among nodes [22]. GCN can learn node feature information and structure information at the same time, and it is extremely suitable for structure of any topology. In the prediction of urban traffic using GCN, the intersections are regarded as vertices in the graph, and the roads between the intersections are represented by the edges. Through the graph convolution operation, the spatial dependence between the intersections is obtained. In [23], Yu *et.al.* proposed a deep learning framework, i.e., Spatio-Temporal Graph Convolutional Networks (STGCN), to tackle the time series prediction problem in traffic domain. The authors formulated the problem on graphs. In [24], the authors proposed a Attention based Spatial-Temporal Graph Convolutional Network (ASTGCN) model, which can process the traffic data directly on the original graph-based traffic network and effectively capture the dynamic spatial-temporal features. In [25], the authors enhanced the traditional GCN with node adaptive parameter learning and data-adaptive graph generation modules and proposed the Adaptive Graph Convolutional Recurrent Network (AGCRN), which can learn node-specific patterns and is applicable to separate traffic series sources.

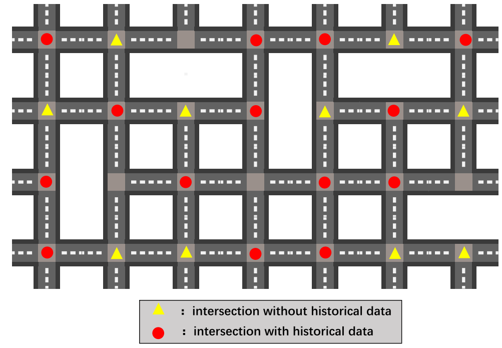


Fig. 3. Intersection distribution map.

Inspired by GCN, in this paper, we propose a novel spatial-temporal traffic flow prediction protocol based on GCN, which incorporates both spatial and temporal dependence of intersection traffic to achieve a more accurate traffic flow prediction. Moreover, our proposed protocol with the Adjacent-Similar algorithm can also effectively predict the traffic flow of the intersections without historical data, which has not been investigated in the literature.

III. Preliminaries

A. Problem Formulation

In this paper, the objective of our investigation is to predict the future traffic flow of urban intersections in a generalized and practical scenario that some of the intersections have historical traffic flow data while others do not. As shown in Fig. 3, circle points represent intersections with historical data, and triangle points represent intersections without historical data. In our assumption, there is no certain distribution pattern among circle and triangle points in our investigated scenario.

The relevant definition involved in the problem scenario is explained. The feature matrix $X \in \mathbb{R}^{T*N*P}$ represents the traffic flow matrix at time T, where N represents the number of intersections and P denotes the number of intersection features.

B. Spatial-Temporal Dependence Modeling

In order to effectively exploit both spatial and temporal information for traffic flow prediction of intersections, we employ an efficient spatial-temporal dependence model to explore the spatial and temporal correlations from the road topology and the limited historical traffic data. It mainly includes spatial dependence modeling, temporal dependence modeling, and other influencing factors modeling.

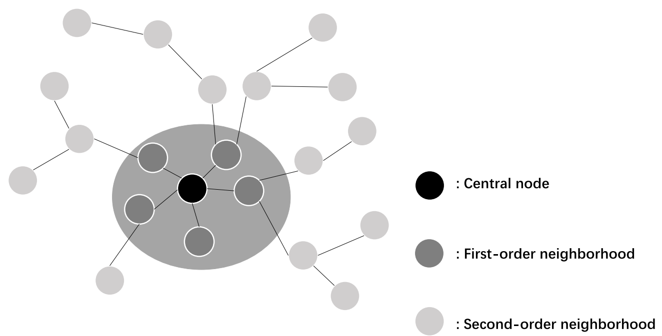


Fig. 4. Graph convolution to obtain first-order neighborhood information of central node.

1) Spatial Dependence Modeling: CNN is often used to deal with spatial dependence problems, but it can only process data of Euclidean structure. However the actual structure of the city is non-Euclidean, and thus we exploit GCN to model the spatial dependence of the topology of intersections.

The objective of GCN is to extract the spatial features of irregular topological graphs. At present, there are two mainstream implementation methods: spatial domain method and spectrum domain method. These two methods understand graphs from two different perspectives.

The spatial domain method extracts the spatial features by finding the neighboring points of each vertex and extracting the neighbor node features [27]. However, each node has a different number of neighbors, and the calculation must target each node. Hence this method is suitable for situations with a simple graph structure and few nodes.

The spectrum domain method uses graph theory to realize the convolution on graphs and utilizes the eigenvalues and eigenvectors of the Laplacian matrix of graphs to study the properties of graphs [28]. We implement the graph convolution from the perspective of the spectrum domain.

The propagation rules of each convolutional layer are as follows:

$$H^{(l+1)} = \sigma(\widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}H^{(l)}W^{(l)}) \tag{1}$$

where σ represents the activation function, $W^{(l)}$ represents the parameter matrix of layer l, $\widetilde{A}$ is $A + I_N$, A is the adjacency matrix of the undirected graph G, I_N is the identity matrix, and $\widetilde{D}$ is the degree matrix of $\widetilde{A}$.

By multiplying the adjacency matrix A and the feature matrix $H^{(l)}$ of layer l, each layer of GCN obtains the summary of the neighboring features of each vertex. Then a matrix $H^{(l+1)}$ that aggregates the features of adjacent vertices can be obtained by multiplying a parameter matrix $W^{(l)}$. We add an identity matrix I_N to the adjacency matrix A, that is $\widetilde{A}$, therefore the feature information of each node is retained. The normalization of the adjacency matrix is to maintain the original distribution of features when multiplying with the feature matrix. Therefore, we multiply the adjacency matrix by $\widetilde{D}^{-1}$ and obtain a symmetric and normalized adjacency matrix: $\widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}$.

As shown in Fig. 1, one-layer graph convolution operation can obtain the spatial information of its own node and the first-order neighborhood for the center point.

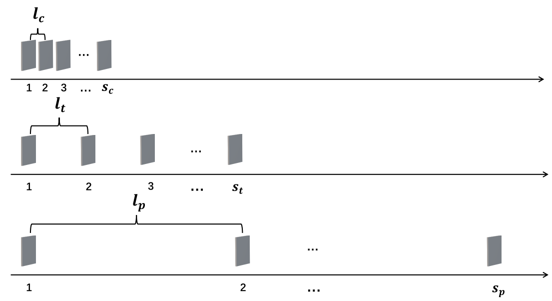


Fig. 5. Temporal dependence feature.

Therefore, the propagation rule of a two-layer GCN can be expressed as:

$$f(X, A) = \sigma(\hat{A}Relu(\hat{A}XW_0)W_1) \tag{2}$$

where $\hat{A} = \widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}$, X is the feature matrix of the traffic flow, $X \in \mathbb{R}^{T*N*P}$, and W_0 and W_1 represent the weight matrices of the first layer and second one in GCN, respectively.

2) Temporal Dependence Modeling: According to the periodicity of historical traffic, we extract three characteristic periods of the flow time period. The time intervals are l_c, l_t, l_p, and the corresponding time spans are s_c, s_t, s_p, respectively.

Thus, the temporal dependence of traffic flow data is reflected in Fig. 5. For example, we suppose l_c is five minutes, l_t is one day, and l_p is one week. Then the traffic flow at time t is periodically correlated with the traffic flow of the previous s_c five minutes, the previous s_t day, and the previous s_p week.

Therefore, the temporal-dependent flow characteristics of time t are:

$$\begin{aligned}\{[X_{t-l_c}, X_{t-2*l_c}, \ldots, X_{t-s_c*l_c}],\\ [X_{t-l_t}, X_{t-2*l_t}, \ldots, X_{t-s_t*l_t}],\\ [X_{t-l_p}, X_{t-2*l_p}, \ldots, X_{t-s_p*l_p}]\}\end{aligned} \tag{3}$$

where $X_t \in \mathbb{R}^{N*P}$ represents the traffic feature matrix at time t. l_c, l_t and l_p are the time intervals of three periodicity, and s_c, s_t and s_p are the corresponding time periods.

Moreover, we use GRU to capture the dependence of the large time step distance. It controls the information flow through gates that can be learned. GRU introduces the reset gate and update gate, which can modify the calculation method of hidden state in Recurrent Neural Network (RNN). For the historical traffic data in three time periods, we establish three corresponding GRU models to learn the temporal traffic information of different periods.

3) Extra Influencing Factors Modeling: In addition to spatial-temporal dependence, traffic flow is also affected by weather conditions, holidays, and other factors. Therefore, extra influencing factors are also added to the provided network. The daily weather is coded according to the average temperature of the day and the weather conditions, and the date is coded according to holiday information.

C. Intersection Without Historical Data

Different from existing works on traffic flow prediction that merely investigated the intersections with historical data, in our work, we consider a more general traffic flow prediction

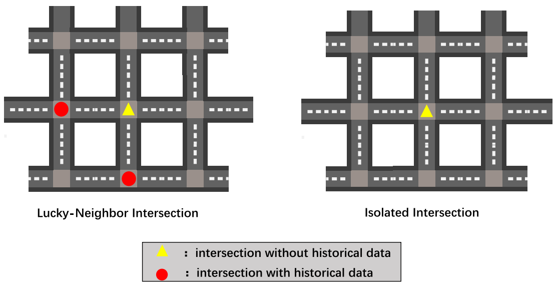


Fig. 6. Intersection without historical data.

scenario that includes the intersections both with and without historical data, and exploit the topology and similarity information of the road network to achieve effective traffic flow prediction of intersections without historical data. According to the situation of adjacent intersections, the intersections without historical data can be divided into the following two categories:

1) Lucky-Neighbor Intersections: As shown in the left side of Fig. 6, circle points are intersections with historical data, and triangle points are intersections without historical data. In some directions of a Lucky-Neighbor intersection, there are adjacent intersections with historical data.

2) Isolated Intersections: As shown in the right side of Fig. 6, for an Isolated intersection, there is no adjacent intersection with historical data in any direction of it.

IV. Hierarchical Traffic Flow Prediction Protocol

A. Hierarchical Traffic Flow Prediction Architecture

To achieve effective traffic prediction for both intersections with and without historical data, we propose a hierarchical prediction architecture as shown in Fig. 7. First, we construct the ST-GCN model to predict future flow of intersections with historical data. Then, we design an Adjacent-Similar algorithm to predict the future traffic of intersections without historical data.

B. Design of the ST-GCN Network

1) ST-GCN Cell: As shown in Fig. 8(a), we first use graph convolution to obtain the spatial dependence, and then update the current state according to the GRU to obtain the new state and output.

2) GRU Network: Taking the time interval of s_c as an example, the time series are $(X_{t-s_c*l_c}, X_{t-(s_c-1)*l_c}, X_{t-(s_c-2)*l_c}, \ldots, X_{t-l_c})$. Then put this sequence data into GCN model with GCN Cell as the unit in chronological order, and the time dependence relationship with s_c is finally obtained.

3) ST-GCN Network: The complete structure of ST-GCN is shown in Fig. 8(b). The input data are a time series of three periodic features and extra influencing factors features. The structure of three periodic features has been introduced in the GRU network. In our proposed model, extra influencing factors include weather and holidays factors. For weather factors, considering both the weather conditions and the temperature, we categorize every-hour weather into four weather conditions, that is sunny, cloudy, rainy, and foggy, which correspond to weather codes 0 to 3, respectively. Moreover, we count the temperature value per hour. For holiday factors, we label every day holiday or not. Holiday factors mainly consider the influence of working days and holidays, where the working day feature corresponds to 0, and the holiday feature corresponds to 1. Hence, for each moment, we can obtain the characteristics of extra influencing factors and put them into the network after standardization.

We construct three GRU networks with the same structure and get the sum of these models. Then, we add the obtained results and other influencing factors. Finally, we get the output through the activation function.

C. Adjacent-Similar Algorithm

We further propose an Adjacent-Similar algorithm including the Adjacent algorithm and the Similar algorithm to predict the traffic flow of intersections without historical data based on the traffic flow of intersections with historical data from the ST-GCN model.

1) Adjacent Algorithm: After the traffic prediction of intersections with historical data, we propose an Adjacent algorithm to predict the traffic of Lucky-Neighbor intersections.

For Lucky-Neighbor intersections, we use urban taxi trajectory data as auxiliary data. As shown in Fig. 9, there are adjacent intersections with historical data in the north, south, and west directions of intersection A, and there is no adjacent intersection in the east direction.

Firstly, we meature the distance cross intersections by auxiliary data. The time required for vehicles of intersections B, C and D arrive at intersection A can be estimated from the urban taxi trajectory. Therefore, we can calculate a part of the flow at the intersection A. We use $f_{loa}^{t,dir}$ to indicate the flow of intersection loa in the direction of dir at time t. Among them, the directional flow at the intersection is the total flow divided equally by the number of branches. Assuming that the number of branches at the intersection is m, taking the north direction as an example, the directional flow is:

$$f_{loa}^{t,north} = f_{loa}^{t,all} * \frac{1}{m}. \tag{4}$$

Therefore, a part of the flow at the intersection A can be calculated as follows:

$$f1_A^t = f_D^{t-\Delta t1,north} + f_C^{t-\Delta t2,east} + f_B^{t-\Delta t3,south}. \tag{5}$$

We can find that there is an intersection in the east of the intersection A. However, due to the lack of data, we cannot obtain the flow data of the adjacent intersection in this direction. Therefore, we need to compensate the total flow. We make the average distribution of the flow in (5), assuming that there are m branches in the actual center intersection. When $f1_A^t$ calculates n branches, the total flow f_A^t can be expressed as:

$$f_A^t = f1_A^t * \frac{m}{n}. \tag{6}$$

Hence, as shown in Fig. 9 step 3, the yellow arrow represents the compensation flow, and thus we can estimate

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

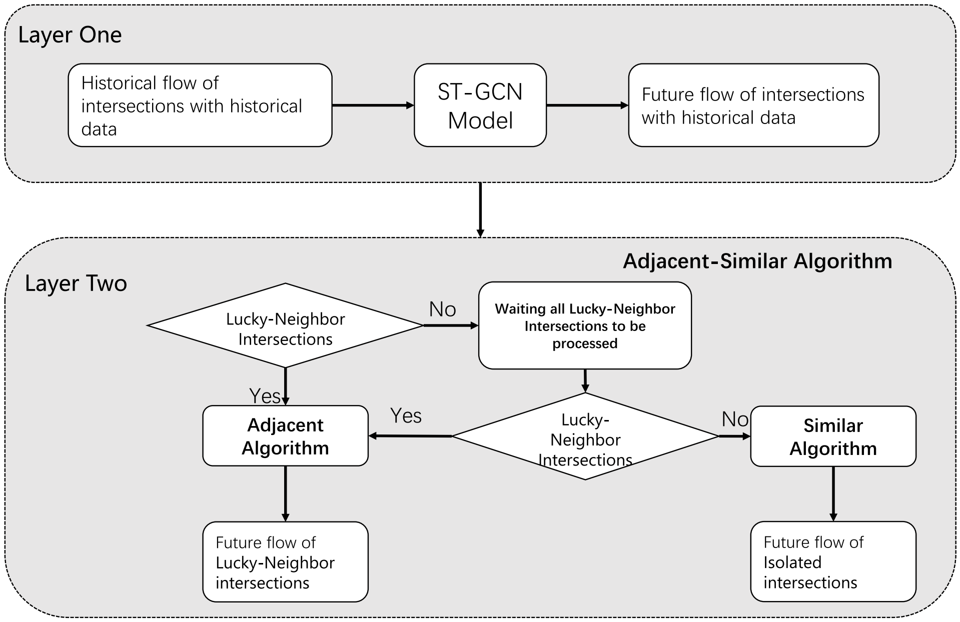
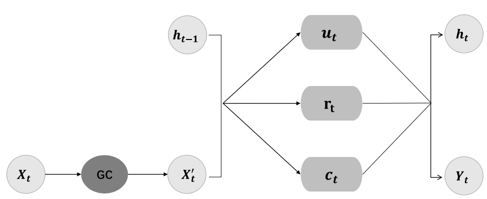
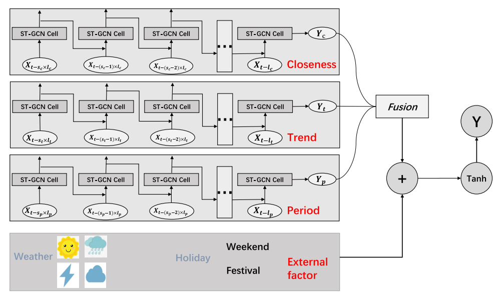


Fig. 7. Hierarchical traffic flow prediction protocol architecture.



(a) Structure of ST-GCN cell.



(b) Structure of ST-GCN network model.

Fig. 8. ST-GCN network.

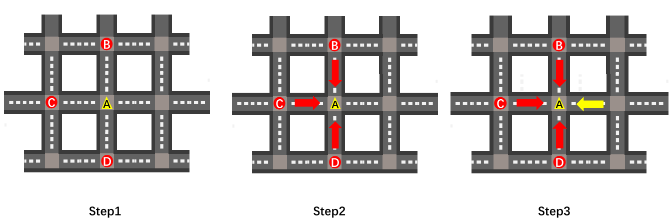


Fig. 9. Adjacent algorithm for Lucky-Neighbor intersections.

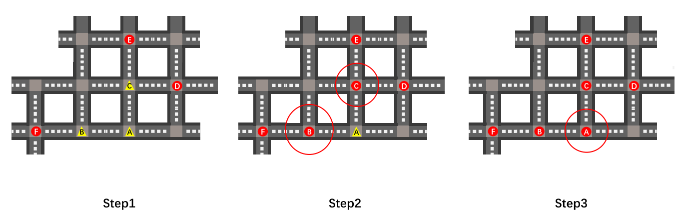


Fig. 10. Adjacent algorithm for isolated intersections.

the flow of the intersection A by the flow of the adjacent intersection with historical data.

After the calculation of Lucky-Neighbor intersections, the attributes of these intersections can be updated as intersections with historical data. Therefore, some Isolated intersections can be transformed to Lucky-Neighbor intersections. As shown in Fig 10, the intersection A originally belonged to Isolated intersections. However, since the traffic calculation of the intersection B and intersection C have been completed, the intersection A has transformed to a Lucky-Neighbor intersection. Thus, we can calculate the traffic flow of the intersection A using the above algorithm for Lucky-Neighbor intersections.

2) Similar Algorithm: However, there is a special case of the Isolated intersections, that is, after processing all existing Lucky-Neighbor intersections, some Isolated intersections still have no adjacent intersections. We cannot utilize the Adjacent algorithm for Lucky-Neighbor intersections to calculate their traffic flow. Therefore, we propose to predict their traffic by exploring their similar intersections according to the topological structure and location factors.

In urban intersections, some intersections may share similar traffic flow tendency. For example, most of the flow changes in the business center area of the city are similar. People flow from the living area into the central area during the morning rush hour, and return to the living area from the central area

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

during the evening rush hour. Another example is that the traffic flow in the commercial area is mostly from afternoon to evening, with less traffic during the day.

Therefore, since there is no historical data for this type of intersection, we try to find their similar intersections. The future traffic at these Isolated intersections can be similarly predicted by the future traffic at the similar intersection.

We take taxi trajectory data as a small sample of urban traffic. Based on the urban taxi data, calculate the taxi traffic at each intersection in the past, and then compare the difference between taxi traffic of intersections without historical data and taxi traffic of intersections with historical data. The intersection with the smallest difference is selected as the most similar intersection. Hence, the traffic flow of these Isolated intersection can be replaced by their similar intersection.

3) Adjacent-Similar Algorithm: We combine the proposed Adjacent algorithm and the Similar algorithm to obtain the Adjacent-Similar algorithm. For intersection without historical data, we first judge whether it is a Lucky-Neighbor intersection. If it is a Lucky-Neighbor intersection, we calculate its future traffic flow based on the traffic of its adjacent intersections, and update its attributes. If it is not a Lucky-Neighbor intersection, we wait until the calculation of all the Lucky-Neighbor intersections has been finished.

After processing all Lucky-Neighbor intersections, we judge the attributes of remaining intersections again. At this time, the attributes of some Isolated intersections may change to the Lucky-Neighbor intersections. Thus we can still calculate their traffic flow through the Adjacent algorithm based on the traffic volume of their adjacent intersections. We repeat the above iterations until no Isolated intersection can be converted into a Lucky-Neighbor intersection.

For the remaining Isolated intersections, we cannot use the Adjacent algorithm to predict their traffic flow. Therefore we look for their similar intersections among the intersections with historical data through the Similar algorithm, and employ the traffic flow of similar intersections as their future traffic.

V. Experiments and Analysis

A. Data Sets

In this section, we evaluate the performance of the hierarchical traffic flow prediction protocol by conducting various experiments based on real traffic data. We use the traffic data from Qingdao, China for specific prediction experiments.

The data sets mainly include two parts:

- Traffic information of intersections. The data descributes the vehicle passing information of 100 intersections in Qingdao, China from September 1st to September 20th including the intersection ID, vehicle ID, timestamp, and so on.
- Taxi trajectory information. The data describes the Global Positioning System (GPS) information of some taxis from September 1st to 20th, including the latitude, longitude, vehicle ID, timestamp, and so on.

The forecast target is the traffic volume of all intersections, including 100 intersection with historical data and 35 intersection without historical data from September 21st to 24th.

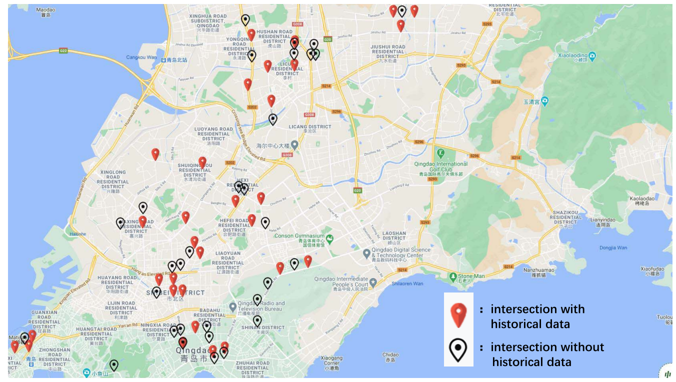


Fig. 11. Map intersection projection (Qingdao, China).

The distribution of these intersections are shown in Fig. 11. We can find that there is no certain distribution pattern between intersections with historical data and intersections without historical data in our investigated scenario.

B. Data Preprocessing

The traffic flow passing by every intersection every five minutes from 7:00 am to 19:00 pm is counted as the traffic information of the intersection. We construct the adjacency matrix for the 100 intersections. According to the urban network structure, if the intersection i is adjacent to the intersection j, the A_{ij} of the adjacency matrix is 1, otherwise the $A_{ij} = 0$.

So the experimental data mainly includes a 100*100 urban adjacency matrix composed of the neighboring relations of 100 intersections, and the daily five-minute traffic data of each intersection. We divide the training set, validation set and test set in a ratio of 6:2:2.

C. Evaluation Metrics

In order to evaluate the accuracy of the model, we use the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE) to evaluate the result.

The smaller the RMSE and MAE values are, the better prediction result is.

D. Model Parameters

1) Training Parameters: The basic parameters of the model mainly include learning rate, batch size, training epoch, and so on. In this experiment, the learning rate is 0.001, the batch size is 32, and the training epoch is 2000. We use the Adam optimizer for training.

2) GCN Parameters: The number of hidden units has a great effect on the model. In order to find the best parameter results, we do experiments on different parameter selections on the validation set. Fig. 12 show the experiment results. It can be seen that the best number of hidden layer units is 32.

3) Time Periodic Parameters: According to equ. 5, we presuppose l_c is five minutes, l_t is one day, l_p is one week. When $s_c = 5$, $s_t = 3$, $s_p = 1$, the error is the smallest.

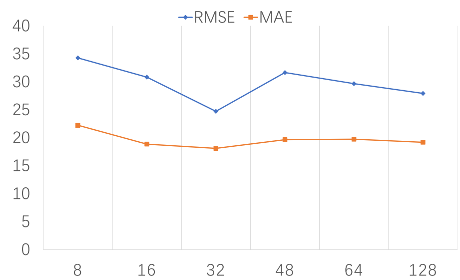


Fig. 12. The relationship between the number of hidden units and RMSE and MAE.

TABLE I
EXPERIMENTS ON EXTRA INFLUENCING FACTORS

	RMSE	MAE
No extra influencing factors	23.14	15.49
Extra influencing factors	22.37	15.11

4) Extra Influencing Factors: We compare the results of the model with and without extra influencing factors on the validation set. As illustrated in Table I, the addition of weather and holiday features has a good effect on the improvement of the model performance.

E. Experimental Results

1) Intersections With Historical Data: We compare our proposed ST-GCN model with the ARIMA, GRU, GCN, CNN, LSTM, ST-ResNet, STGCN, ASTGCN models.

- ARIMA [3]: This model uses time-oriented temporal data to explore the linear relationship among traffic flow.
- GRU: In the GRU model, we only consider the temporal dependence of the model. A unified GRU model is established for all intersections to excavate the periodic dependence contained in the historical flow data. We choose the same periodic length as the ST-GCN model.
- GCN: In the GCN model, we only consider the spatial dependence of the model. Without considering the long-term and short-term time dependence of traffic flow, it only uses the traffic flow of all intersections at the previous time to predict the traffic flow of all intersections at the next time.
- CNN: We use a two-layer CNN to replace the graph convolution in the ST-GCN model. The CNN structure is Conv2d-Pooling-Conv2d-Pooling-FC-Activation. Conv2d is the convolution layer, Pooling is the max-pooling layer, FC is the fully connected layer, and Activation is the activation layer. This model is consistent with the ST-GCN model in temporal dependence.
- LSTM: We use LSTM instead of the GRU module in the ST-GCN model. In LSTM, information retention is determined by the input gate and forget gate. In terms of spatial dependence, we still use graph convolution to explore spatial dependence, which is consistent with the ST-GCN model.

TABLE II
COMPARSION WITH BASELINES

Method	RMSE	MAE
ARIMA	29.78	24.04
GRU	31.09	22.71
GCN	43.21	36.37
CNN	26.28	19.41
LSTM	32.08	24.67
ST-ResNet	32.98	23.74
STGCN	29.95	21.77
ASTGCN	22.45	17.81
ST-GCN	**23.14**	**15.49**

- ST-ResNet [17]: This model employs convolution-based residual networks to model nearby and distant dependencies between any two regions in a city.
- STGCN [23]: The authors formulated the problem on graphs and built the model with complete convolutional structures.
- ASTGCN [24]: ASTGCN mainly consists of three independent components to respectively model three temporal properties of traffic flows and each component contains the spatial-temporal attention mechanism.

We compare our proposed ST-GCN model with above models. The result can be found in Table II. We conclude that out model has the following advantages:

- High prediction precision: It can be found in Table II that the error of the ST-GCN model is smaller than almost most baseline models, which indicates that the prediction precision result of the ST-GCN model is remarkable. For example, the RMSE of ST-GCN is 22.29% less than ARIMA, and the MAE is 35.56% less than ARIMA. This is because the ARIMA model can't excavate the complex temporal and spatial dependence in traffic flow.
- Spatial-temporal dependence prediction capability: To verify whether the ST-GCN model has the ability to portray spatial and temporal features from traffic data, we compare the ST-GCN model with the GRU model, the GCN model, and other spatial-temporal models. In terms of spatial dependence, we compare the ST-GCN model with the GRU model. The GRU model shares the same temporal dependence with the ST-GCN model, but it can not capture the spatial dependence among intersections. From Table II, we can see that compared with the GRU model, the RMSE of ST-GCN is reduced by 25.57% and the MAE of ST-GCN is reduced by 31.79%. The ST-GCN model achieves better results in terms of spatial dependence because it introduces the GCN to obtain the spatial relationship among intersections. In terms of temporal dependence, we compare the ST-GCN model with the GCN model. These two models adopt the same GCN structure for spatial dependence processing. In terms of time dependence, the GCN model does not consider the periodic characteristics of time series. According to the experimental results, the RMSE of ST-GCN is reduced by 46.44% compared to GCN and the

TABLE III
EXPERIMENTS ON 30MIN PREDICTION

Method	RMSE	MAE
ST-ResNet	68.17	42.06
ASTGCN	42.07	33.72
ST-GCN	**41.26**	**25.96**

MAE is reduced by 57.40%. The ST-GCN model achieves better results in terms of temporal dependence because it takes into account the periodicity of the time series and fully exploits the time dependence of the historical traffic sequence.

We also compare our ST-GCN model with some other spatial-temporal traffic prediction models, such as the ST-ResNet, STGCN and ASTGCN models. Compared with the ST-ResNet model, the RMSE and MAE of our designed ST-GCN model are reduced by 29.83% and 34.75%, respectively. This is because as a contrast to GCN employed in our model, CNN used in ST-ResNet ignores the actual arrangement of intersections, so this model is more suitable for regional traffic prediction. Compared with the STGCN model, the RMSE and MAE of our ST-GCN model are less than the STGCN model. In terms of capturing temporal dependence, the GRU model is better than convolution, and CNN is not very sensitive to long time series. Therefore, it is better to use GRU for mining temporal dependence. Compared with the ASTGCN model, we can find that the RMSE of the ASTGCN model is a little less than that of our ST-GCN, but the ASTGCN model needs to calculate the spatial-temporal attention matrix among intersections at each moment, and thus it requires much more time and space complexity that is not appropriate for our short-term and real-time traffic prediction problems.

- The superiority of components: To verify the superiority of the component of GCN and GRU in our ST-GCN model, we replace the GCN in ST-GCN with a two-layer CNN structure, and the GRU with LSTM, corresponding to the CNN model and LSTM model in Table II. It can be seen from the results that compared with the two-layer CNN, the RMSE of ST-GCN is reduced by 11.94%, and the MAE is reduced by 20.19%, which indicates that the results of the two-layer GCN are much better than CNN, and the effect is better in mining the dependency relationship among non-Euclidean structure nodes. Compared with LSTM, the RMSE and MAE of ST-GCN are reduced by 27.86% and 37,21%, respectively. Moreover, the processing speed of GRU is faster and the model complexity is lower than LSTM. This is because GRU combines the forget gate and the input gate of LSTM into a single update gate, and also mixes the cell state and hidden state. Therefore, the final model is simpler than the standard LSTM model and can get better results.
- Long-term prediction capability: Although our problem is mainly targeted at short-term forecasting, we also do experiments to verify the long-term forecasting ability of our proposed ST-GCN model. We have experimented with ST-GCN, ST-ResNet, and ASTGCN models for 30-min traffic prediction, and the experimental results are shown in Table III. We can see that our model also achieves a better performance in a relatively long-term prediction, which indicates that our method is not sensitive to the prediction time intervals. Therefore, our ST-GCN model can be applied not only for short-term prediction but also for long-term prediction.

TABLE IV
NO DATA INTERSECTION ALGORITHM RESULTS

Method	RMSE	MAE
Adjacent Algorithm	67.52	55.35
Similar Algorithm	104.49	66.14
Adjacent-Similar Algorithm	**65.21**	**53.79**

In general, compared with other traffic prediction methods, our ST-GCN model has a smaller error and performs better in terms of spatial-temporal dependence. Therefore, for short-term prediction, the ST-GCN model can achieve quick and real-time prediction with high accuracy. At the same time, in the long-term prediction, our model also has a great performance, which shows that our model is the most suitable for short-term and real-time traffic prediction.

2) Intersections Without Historical Data: The result of the intersections without historical data is obtained in Table IV. We analyze the Adjacent-Similar algorithm through the analysis of the Adjacent algorithm and the Similar algorithm, respectively.

The logic of the Adjacent algorithm particularly conforms to the actual flow of traffic flow. The main reasons for errors of this algorithm are the calculation of directional flow and the limitation of taxi data. In terms of the calculation of directional flow, in this algorithm, the traffic flow of each direction is evenly distributed. Actually, the flow of traffic is not the same in each direction. If we can obtain the direction flow of each intersection, the error will be significantly reduced. In terms of the taxi data, the taxi trajectory data is used to calculate the time required for vehicles cross two intersections. However, taxis may pick up and wait for passengers, so the time required cross two intersections may have errors. Complete taxi data or more accurate measurement of distance between intersections may greatly improve the algorithm result.

According to the Similar algorithm, the flow similarity between intersections is compared mainly based on taxi trajectory data. From the similarity sorting results, it can be seen that most of the alternative intersections are adjacent intersections, or intersections share similar structures.

For example, as shown in Fig. 13, intersection 100060 is an intersection with historical data, and intersection 100061 is an intersection without historical data. According to the Similar algorithm, intersection 100060 is the most similar intersection with intersection 100061. It can be seen from the map that intersection 100061 is the adjacent intersection of 100060, so they share the similar traffic change trend. And the structure of the two intersections is also basically similar. Therefore the future traffic flow of the intersection 100061 can be approximately replaced by the predicted traffic flow of intersection 100060.

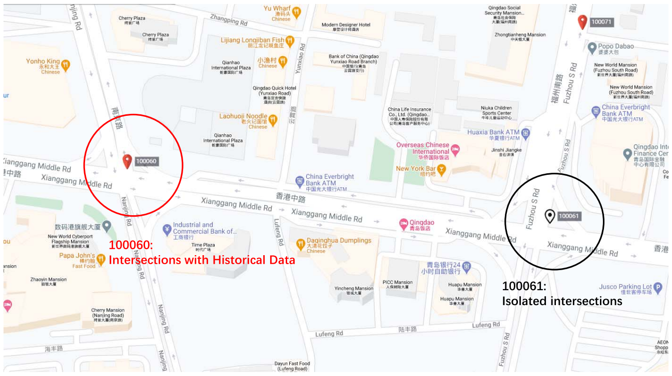


Fig. 13. Example of similar intersections.

The error of the Similar algorithm is mainly due to the limitation of reference intersections. For each intersection without historical data, we can find a most similar intersection according to the result of taxi flow error sorting. Projecting these intersections on the map, we can find that most of the similar intersections with small errors are adjacent intersections, or non-adjacent intersections but with close distances and similar structures. But for intersections with large errors, most of the them are non-adjacent, or the distances are relatively far. In other words, the intersection with the smallest flow difference may cannot replace the flow of this intersection. To solve this question, we need to obtain enough reference intersections so that for each intersection without historical data, we can always find the intersection with the similar flow.

The Adjacent-Similar algorithm is the combination of the Adjacent algorithm and the Similar algorithm. The RMSE of the Adjacent-Similar algorithm is 65.21 and the MAE is 53.79. The prediction accuracy with our designed Adjacent-Similar algorithm is acceptable and can provide efficient guidance for practical applications.

VI. Conclusion

In this paper, we focused on a more challenging but practical traffic flow prediction scenario, which includes both intersections with and without historical data. In order to achieve effective traffic flow prediction for these intersections, we proposed a novel ST-GCN-based hierarchical traffic flow prediction protocol. The designed Adjacent-Similar algorithm therein is able to predict the intersections without historical data with a good prediction accuracy through their spatial-temporal correlations with the ones with historical data. We further conducted various experiments based on practical traffic data of the city of Qingdao, China. The results have verified that the efficiency and advantages of our proposed ST-GCN-based traffic flow prediction protocol compared with many existing baseline models.

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.

[2] A. Abadi, T. Rajabioun, and P. A. Ioannou, "Traffic flow prediction for road transportation networks with limited traffic data," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 653–662, Aug. 2014.

[3] H. Dong *et al.*, "Road traffic flow prediction with a time-oriented ARIMA model," in *Proc. 5th Int. Joint Conf. INC*, Seoul, South Korea, Nov. 2009, pp. 1649–1652.

[4] P. J. Hargrave, "A tutorial introduction to Kalman filtering," *IEE Colloq. Kalman Filters: Introduction, Appl. Future Develop.*, London, U.K., Feb. 1989, pp. 1-1–1-6.

[5] Y.-S. Gong and Y. Zhang, "Research of short-term traffic volume prediction based on Kalman filtering," in *Proc. 6th Int. Conf. Intell. Netw. Intell. Syst.*, Shenyang, China, Nov. 2013, pp. 99–102.

[6] H. Hong, W. Huang, X. Zhou, S. Du, K. Bian, and K. Xie, "Short-term traffic flow forecasting: Multi-metric KNN with related station discovery," in *Proc. 12th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Zhangjiajie, China, Aug. 2015, pp. 1670–1675.

[7] V. Vapnik, "Support-vector networks," in *Proc. Conf. Mach. Learn.*, 1995, pp. 273–297.

[8] Q. Li, "Short-time traffic flow volume prediction based on support vector machine with time-dependent structure," in *Proc. IEEE Intrumentation Meas. Technol. Conf.*, Singapore, May 2009, pp. 1730–1733.

[9] S. Sun, C. Zhang, and G. Yu, "A Bayesian network approach to traffic flow forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 124–132, Mar. 2006.

[10] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: Deep belief networks with multitask learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2191–2201, Oct. 2014.

[11] 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, Dec. 2015.

[12] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," in *Proc. 9th Int. Conf. Artif. Neural Netw.*, Edinburgh, U.K., Sep. 1999, pp. 850–855.

[13] P. Poonia and V. K. Jain, "Short-term traffic flow prediction: Using LSTM," in *Proc. Int. Conf. Emerg. Trends Commun., Control Comput. (ICONC3)*, Lakshmangarh, India, Feb. 2020, pp. 1–4.

[14] Z. Zhao *et al.*, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, 2017.

[15] J. Chung *et al.*, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proc. NIPS Deep Learn.*, Dec. 2014, pp. 1–5.

[16] 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.

[17] J. Zhang *et al.*, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2017, pp. 1–7.

[18] J. Zhang, Y. Zheng, J. Sun, and D. Qi, "Flow prediction in spatio-temporal networks based on multitask deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 3, pp. 468–478, Mar. 2020.

[19] C. Chen *et al.*, "Exploiting spatio-temporal correlations with multiple 3D convolutional neural networks for citywide vehicle flow prediction," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Singapore, Apr. 2018, pp. 893–898.

[20] X. Cheng, R. Zhang, J. Zhou, and W. Xu, "Deeptransport: Learning spatial-temporal dependency for traffic condition forecasting," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Rio de Janeiro, 2018, pp. 1–8.

[21] H. Yao *et al.*, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proc. 23rd AAAI Conf. Artif. Intell.*, 2019, pp 5668–5675.

[22] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA, 2016, pp. 3844–3852.

[23] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3634–3640.

[24] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spa-tialtemporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 922–929.

[25] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," 2020, *arXiv:2007.02842*.

[26] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–8.

[27] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2014–2023.

[28] A. James and T. Don, "Diffusion-convolutional neural networks," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Red Hook, NY, USA, 2016, pp. 2001–2009.

Hanqiu Wang (Graduate Student Member, IEEE) received the B.E. degree from Tongji University in 2020, where she is currently pursuing the master's degree in software engineering. Her research interests include data mining and traffic flow prediction.

Rongqing Zhang (Member, IEEE) received the B.S. and Ph.D. degrees (Hons.) from Peking University, Beijing, China, in 2009 and 2014, respectively.

From 2014 to 2018, he worked as a Post-Doctoral Research Fellow at Colorado State University, Fort Collins, CO, USA. Since 2019, he has been an Associate Professor with Tongji University, Shanghai, China. He has authored or coauthored two books, two book chapters, and over 100 papers in refereed journals and conference proceedings. His current research interests include vehicular communications and networking, UAV communications, and autonomous driving. He was also awarded as a International Presidential Fellow of Colorado State University in 2017. He was a recipient of the Academic Award for Excellent Doctoral Students, Ministry of Education of China. He was a co-recipient of the First-Class Natural Science Award, Ministry of Education of China. He received the Best Paper Awards at IEEE ITST 2012, ICC 2016, GLOBECOM 2018, and ICC 2019. He is currently serving as an Associate Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and *IET Communications*.

Xiang Cheng (Fellow, IEEE) received the Ph.D. degree jointly from Heriot-Watt University and The University of Edinburgh, Edinburgh, U.K., in 2009. He is currently a Boya Distinguished Professor with Peking University. He led the establishment of three Chinese standards (including one industry standard and two group standards) and participated in the formulation of ten 3GPP international standards and two Chinese industry standards. His research interests include channel modeling, wireless communications, and data analytics, subject on which he has published more than 200 journals and conference papers, nine books, and holds 17 patents. He is a Distinguished Young Investigator of China Frontiers of Engineering. He was a recipient of the IEEE Asia Pacific Outstanding Young Researcher Award in 2015. He received the National Science Fund for Distinguished Young Scholars in 2021. He was a co-recipient of the 2016 IEEE JSAC Best Paper Award: Leonard G. Abraham Prize, and the First-Rank and Second-Rank Award in Natural Science, Ministry of Education, in China. He has also received the Best Paper Awards at IEEE ITST 2012, ICCC 2013, ITSC 2014, ICC 2016, ICNC 2017, GLOBECOM 2018, ICCS 2018, and ICC 2019. He received the First Prize and the Second Prize in Nature Science Award of Ministry of Education of China, the First Prize in Nature Science Award of Chinese Institute of Electronics, the Young Science and Technology Award of Chinese Institute of Electronics, and the First Prize in the Technology Invention Award of the China Institute of Communications. He has served as the symposium lead chair, co-chair, and member of the technical program committee for several international conferences. He is also a Subject Editor of *IET Communications* and an Associate Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, the IEEE WIRELESS COMMUNICATIONS LETTERS, and the *Journal of Communications and Information Networks*. He was a Distinguished Lecturer of the IEEE Vehicular Technology Society. He was a Highly Cited Chinese Researcher in 2020. He is an IEEE Distinguished Lecturer. In 2021, he was selected into two world scientist lists, including World's Top 2% Scientists released by Stanford University and Top Computer Science Scientists released by Guide2Research.

Liuqing Yang (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Minnesota, Minneapolis, MN, USA, in 2004. She is currently a Professor with the University of Minnesota. Before joining the University of Minnesota, she has been a Faculty Member with the Department of Electrical and Computer Engineering, University of Florida (2004–2010) and Colorado State University (2010–2020). Her research interests include communications and networking—subjects on which she has published more than 370 journals and conference papers, four book chapters, and five books. She was a recipient of the ONR Young Investigator Program (YIP) Award in 2007, the NSF Faculty Early Career Development (CAREER) Award in 2009, and the Best Paper Award at IEEE ICUWB 2006, ICCC 2013, ITSC 2014, GLOBECOM 2014, ICC 2016, WCSP 2016, GLOBECOM 2018, ICCS 2018, and ICC 2019. She is the Editor-in-Chief of *IET Communications*, a Executive Editorial Committee (EEC) Member of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and a Senior Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING. She has also served as an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE INTELLIGENT SYSTEMS, and *PHYCOM: Physical Communication*, and as a program chair, a track/symposium, or a TPC chair for many conferences.