IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 14, 2021, accepted December 29, 2021, date of publication December 31, 2021, date of current version January 7, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3139934

A Hybrid Model Integrating Local and Global Spatial Correlation for Traffic Prediction

SIYUN FENG[1], JIASHUANG HUANG[1], QINQIN SHEN[2], QUAN SHI[2], (Member, IEEE), AND ZHENQUAN SHI[2]

[1]School of Information Science and Technology, Nantong University, Nantong 226000, China
[2]School of Transportation and Civil Engineering, Nantong University, Nantong 226000, China

Corresponding authors: Zhenquan Shi (szq@ntu.edu.cn) and Quan Shi (sq@ntu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771265 and Grant 62176131.

ABSTRACT Accurate traffic prediction can effectively alleviate traffic congestion problems. The complex spatial correlation of traffic flow contributes to the challenging prediction problem. Most current prediction methods focus on learning local spatial correlation, ignoring the spatial correlation of long-distance traffic flow. In this paper, we combine the improved Graph Convolutional Network (GCN) with Gated Recurrent Unit (GRU) to propose a hybrid model integrating local and global spatial correlation (T-LGGCN) for traffic prediction. The model consists of two parts: global spatial-temporal component and local spatial-temporal component. For the global spatial-temporal component, we construct the global correlation matrix to improve the GCN for obtaining the global spatial correlation. And GRU is stacked to obtain the global spatial-temporal correlation. For the local spatial-temporal component, we utilize the strategy of combining Fully Connected Layer (FCL) and GCN to analyze the local spatial correlation. Similarly, GRU is used to perform the output of local spatial-temporal correlation. The output of the two components is finally summed, and the prediction results are generated with the dense network. Experiments were conducted using the highway datasets PEMS04 and PEMS08 from the Caltrans Performance Measurement System, and the results show that our model significantly outperforms state-of-the-art baselines.

INDEX TERMS Traffic prediction, graph convolutional network, global spatial correlation, local spatial correlation.

I. INTRODUCTION

With the acceleration of traffic construction, a large number of highways have sprung up. As the connecting part between cities, highways play an important role in our daily life and have greatly improved the inter-city traffic capacity [1]. At the same time, due to the rapid economic development, car ownership is gradually increasing, and traffic problems are becoming more and more serious. Intelligent Transportation System (ITS) effectively integrates advanced science and technology in transportation, which can strengthen the connection between cars, roads, and travelers. Also, it can improve the rationality of traffic resource allocation and mitigate traffic problems [2].

Traffic prediction is an important part of ITS. Accurate traffic prediction is the key to implementing the ITS. Prediction results will directly influence the effect of traffic guidance [3]. Generally, effective traffic prediction focuses on capturing the spatial and temporal correlation of traffic flow [4], [5]. Temporal correlation means that the traffic flow state at one moment has a significant impact on the state at the next moment. For example, if traffic congestion occurred at the previous moment, it may still be congested at the next moment. Moreover, the temporal correlation also shows periodicity. For instance, the traffic peaks on weekdays are usually from 8:00 to 9:00 in the morning and from 5:00 to 7:00 in the evening, which happen at a similar time during the weekdays. All of this information is relevant to the traffic flow state at the future moment. Currently, temporal correlation features are commonly extracted by models such as Temporal Convolutional Network (TCN) [6], Recurrent Neural Network (RNN) [7], Long Short-Term Memory (LSTM) [8], and Gated Recurrent Unit (GRU) [9]. As for the spatial correlation, it exists in the traffic flow in the same direction. Specifically, when traffic flow at a sensor is congested, the traffic flow around it will also be affected.

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

Recently, the graph method is usually used to analyze the spatial correlation, such as Graph Convolutional Network (GCN) [10], Graph Attention Network (GAT) [11].

The spatial correlation analysis of traffic flow is a recent research hotspot. Given the existing research, two main issues need to consider:

(1) Current studies focus on the study of local spatial correlation [12]. Suppose one sensor represents one node, local spatial correlation means that the algorithm only considers the spatial correlation between directly adjacent or 2^{nd} order adjacent nodes [13]. However, spatial correlation also exists in a more extensive transportation network. Global spatial correlation refers that the algorithm simultaneously considers the spatial information of distant nodes. The specific manifestation of global spatial correlation is, for instance, traffic congestion can affect not only the neighboring nodes but also the reachable long-distance nodes [12]. In addition, areas with similar functions will have a similar traffic flow state even if they are a far distance from each other. For example, the traffic flow state around schools in different areas is similar. They are all under peak traffic during the morning and evening. Traffic around malls on holidays is more congested than that on weekdays. Considering the spatial information of these distant nodes can also improve the algorithm prediction accuracy. The existing studies are based on the physical adjacency to characterize spatial correlation, ignoring that regions with similar traffic flow state may be very distant from each other [14]. Therefore, based on the local spatial correlation, it is necessary to consider introducing the global spatial correlation to further enhance the spatial correlation analysis capability.

(2) GCN shows good prediction performance on the non-Euclidean datasets [15]. By aggregating the spatial information through the topology of the road network, it can perform a better analysis of the spatial correlation of traffic flow [16], [17]. However, it is already a well-known problem that multilayer GCNs can cause smoothing [18]. T-GCN [19] stacked two layers of GCN and one layer of GRU, and the prediction curves have shown smooth. If the models fail to predict the fluctuations of traffic flow, it will affect the quality of subsequent traffic guidance, so this problem also needs to be addressed urgently.

In this end, we propose a hybrid model integrating local and global spatial correlation (T-LGGCN) for traffic prediction. We model each sensor as a node to analyze the spatial-temporal correlation of traffic flow. We use T-GCN as the base structure to build the global spatial-temporal component and the local spatial-temporal component. Firstly, regardless of the physical connectivity of the sensors, the correlation between any two sensors is analyzed using the correlation method to construct the global correlation matrix. After that, an improved graph convolution is used to encode the global spatial features. Then, the local spatial correlation between sensors and their neighboring sensors is considered. The sensor's features are first analyzed by Fully Connected Layer (FCL), and the output features are put into GCN for local spatial correlation analysis. Finally, a layer of GRU is stacked on each part for temporal correlation analysis. The output of the two parts are summed and the predicted values are obtained through the dense layer.

The contributions of our work are as follows:

(1) We design the global spatial-temporal component, which considers the spatial correlation of traffic flow between non-first-order neighboring sensors. The correlation matrix is used to realize the weighted summation of spatial features of sensors. And then, combined with GRU, it can effectively analyze the global spatial-temporal correlation between sensors.

(2) We develop a local spatial-temporal component. It combines the sensor nodes' own features and the spatial correlation between first-order neighboring nodes. This component can effectively analyze the local spatial-temporal correlation by weighted fusion of its own features and first-order neighboring spatial correlation features. Meanwhile, it avoids the smoothing problem of GCN and enhances the ability to predict the fluctuation of traffic flow.

(3) We conduct extensive experiments on two real datasets, PEMS04 and PEMS08. The experiments show that our model has a better prediction effect.

The rest of this paper is organized as follows. In section 2, we introduce the related work about traffic prediction. In section 3, we present the model proposed in this paper. In section 4, we use two datasets of different area sizes to validate and analyze our model. In section 5, we discuss our model. In section 6, we summarize the work of this paper.

II. RELATED WORK

Traffic prediction focuses on how to predict the traffic flow state of the next moment based on historical and real-time data, such as volume, density, speed, etc. Accurate traffic prediction can provide real-time traffic information for travelers, which is important to improve road capacity. Traffic flow is complex, variable, and uncertain, and the main characteristics are: nonlinearity, stochasticity, periodicity, and spatial-temporal correlation [20]. To address these characteristics, many prediction methods have been available. The methods can be classified into traditional methods and deep learning methods according to their development process [15].

A. TRADITIONAL METHODS

Traditional methods include classical statistical methods and machine learning methods. Classical statistical methods are more used in the early stage of prediction, such as Historical Average (HA) [21], Auto-Regressive Integrated Moving Average (ARIMA), and its variants [22], [23]. These models perform well under smooth traffic flow. However, when the traffic flow changes drastically, they will show obvious shortcomings and cannot better explore the nonlinearity and uncertainty of traffic flow. Besides, these methods only consider the temporal correlation, ignoring the spatial correlation. As the demand for prediction accuracy increases, the prediction methods gradually turn to machine learning methods such as Support Vector Regression [24], K-Nearest Neighbor [25], and Bayesian model [26], etc.

B. DEEP LEARNING METHODS

With the rapid development of artificial intelligence technology, more and more scholars are using deep learning methods, especially neural networks, to solve traffic prediction problems [27]. Neural networks can better fit nonlinear mapping relationships and efficiently capture the internal features of traffic flow. Commonly used networks include Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Graph Convolutional Network (GCN). CNN [28], [29] can efficiently process grid-type data. It can only handle Euclidean data since its translation invariance. RNN [30], LSTM, and GRU are suitable for processing time-series data [31-33]. These networks rely on the sequential temporal order of the data itself. We usually use them to model the temporal correlation of traffic flow. Graph structured data has emerged in recent years. Researchers have started to study how to build deep learning on graphs [34]. GCN [35] is an important branch of graph neural networks. It has been applied to traffic prediction and is effective in extracting the spatial correlation of traffic flow [36]–[38].

In order to fully analyze the complex characteristics of traffic flow such as nonlinearity, randomness, and spatial-temporal correlation, hybrid neural network structure is usually utilized instead of a single neural network structure. Zheng *et al.* [39] developed an attention-based Conv-LSTM module to extract spatial features and temporal features separately. Zhang *et al.* [19] constructed a GCN-GRU two-layer network structure to analyze the spatial-temporal correlation and achieved good results when applied to the Los Angeles freeway dataset. Zhang *et al.* [40] used GCN and feedforward neural networks to consider time, space, weather conditions, and the date to predict highway traffic flow.

The original GCN can only analyze the geographical correlation of sensors, which does not reflect the deep spatial correlation. Some studies have been carried out to improve the GCN. Li *et al.* [41] combine GCN and GRU to construct a DCRNN network. The model replaces the parameter matrix in GRU with the convolution of the parameter matrix and the Laplacian matrix. Based on DCRNN, Guo *et al.* [42] constructed a dynamic Laplacian matrix by summing the initial Laplacian matrix and the parameterized residual matrix. Wu *et al.* [43] proposed that initial graph structures do not always reflect true spatial correlation. They use adaptive matrices to learn internal spatial relationships for prediction. Li *et al.* [44] put aside the inherent road network structure. They proposed an adaptive graph convolution structure to automatically learn the interrelationships between nodes through training. Lv [45] *et al.* encoded the non-Euclidean correlation and semantic structure of the road network into multiple graphs and then constructed a multi-graph convolutional network to mine these correlations. Li [17] *et al.* used data-driven adjacency matrices to mine the spatial relationships of graph structures.

Compared with existing methods, our model combines the local and global spatial correlation to explore the spatial correlation of traffic flow comprehensively. In the actual road, spatial correlation does not necessarily exist between adjacent sensors. In the same traffic flow direction, spatial correlation also exists between non-first-order neighboring sensors. Through the global spatial-temporal component, we can discard the useless sensor spatial information and improve the capability of global spatial-temporal correlation analysis. Also, in the local spatial-temporal component, we separate node feature learning and neighboring nodes spatial correlation mining, which can effectively avoid the smoothing problem. By combining analysis from both global and local perspectives, our model can deeply analyze the spatial-temporal correlation of traffic flow and improve the prediction capability.

III. METHODOLOGY

A. PROBLEM DEFINITION

The relevant definitions covered in this paper are as follows:

Definition 1: Geographic adjacency matrix $A \in R^{N\times N}$, where A is composed of a_{ij}, N is the number of sensors. We use the distance between sensors to calculate the value of a_{ij} by the threshold Gaussian kernel [46], which is calculated as follows:

$$a_{ij} = \begin{cases} e^{-\frac{dist(i,j)^2}{\sigma^2}}, & dist(i,j) < \lambda \\ 0, & dist(i,j) \geq \lambda \end{cases} \tag{1}$$

where a_{ij} represents the adjacent weight between sensor i and sensor j, $dist(i,j)$ is the distance between sensor i and sensor j, σ^2 is the variance, and λ is the threshold.

Definition 2: Global correlation matrix $C \in R^{N\times N}$, where C_{ij} stands for the spatial influence relationship of sensor i on j. This matrix describes the correlation between sensors within the research network. If the value is not 0, the value indicates the degree of spatial correlation of sensor i on j.

Definition 3: Spatial-temporal feature matrix $X_T^S \in R^{T\times S}$, where T is the total time steps, S is the total number of sensors. The spatial-temporal matrix is constructed using the full amount of feature data of the traffic flow in the network to be studied. Each column represents a sensor in the road network, and the rows represent the values of one-time slice for each sensor. The structure is as follows:

$$X_T^S = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^s \\ x_2^1 & x_2^2 & \cdots & x_2^s \\ \vdots & \vdots & \cdots & \vdots \\ x_t^1 & x_t^2 & \cdots & x_t^s \end{bmatrix} \tag{2}$$

where $S = \{1, 2, \ldots, s\}$ represents the set of sensors. $T = \{1, 2, \ldots, t\}$ denotes the set of time steps. And X_T^S is the spatial-temporal matrix, which contains s sensors and the

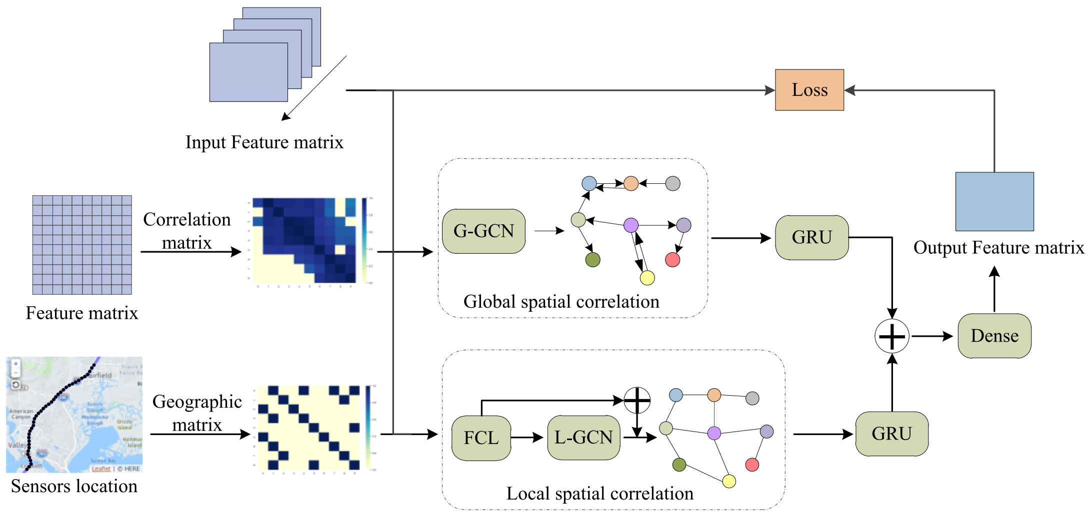


FIGURE 1. The T-LGGCN structure which is composed of two parts: Global spatial-temporal component and local spatial-temporal component. The feature matrix is used to calculate the correlation matrix. And the error between input feature matrix and output feature matrix is adjusted by the loss function.

time step is t. X_t^s is the value of sensor numbered s at the moment t.

Therefore, based on the above definitions, assuming that the current moment is t, our prediction task is to forecast the traffic flow in the future period (i.e., y_{t+1}) based on historical traffic flow data (i.e., X_T^S), geographic adjacency matrix (i.e., A) and global correlation matrix (i.e., C):

$$y_{t+1} = f(X_{t-T}^S, \cdots, X_{t-1}^S, X_t^S; A; C) \tag{3}$$

where T represents the time step of the input and f is the model of this paper.

B. MODEL OVERVIEW

Fig. 1 shows the structure of the model proposed in this paper. The model is composed of two components for modeling the local spatial-temporal correlation and global spatial-temporal correlation. The global spatial-temporal component consists of a global graph convolution and a GRU network. We first establish the global correlation degree matrix and use the global graph convolution to extract the global spatial features. Then the GRU is combined to capture the global spatial-temporal correlation of traffic flow. Besides, the local spatial-temporal component is stacked by a layer of FCL, GCN, and GRU. The first layer of FCL is used to extract the nodes' own features, and the output is utilized to extract the local spatial correlation by the GCN. The output of these two layers is fused and input to GRU to obtain the local spatial-temporal correlation of traffic flow. Finally, the output of the two components is summed, and we use the dense layer to control the output steps in order to obtain the prediction results. We will describe each module in the following subsections.

C. SPATIAL CORRELATION

GCN is a popular neural network for processing spatial correlation. The topology of the road network is represented as an undirected graph $G = (V, E)$, where V is the set of nodes representing the sensors, and E is the set of edges, denoting the adjacency of the sensors. Convert the adjacency of the graph G into the adjacent matrix by the method as defined in Definition 1, then the propagation rule of GCN is as follows:

$$H^{(l+1)} = \sigma(\hat{A}H^{(l)}W^{(l)}) \tag{4}$$

where $\hat{A} = \tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}$, $\tilde{A} = I + A$, $I \in R^{N\times N}$ is the identity matrix, $A \in R^{N\times N}$ represents the adjacency matrix of the graph G, $\tilde{D}_{i,i} = \sum_j \tilde{A}_{i,j}$, $H^{(l)}$ denotes the output of the lth layer, $H^{(0)} = X_T^R$, $W^{(l)}$ is the weight parameter matrix of the lth layer.

The adjacency matrix represents the geographical structure of the real road network. Equation 4 uses $\hat{A}$ to fuse the features of adjacent sensors so that sensors can obtain the new feature representation. However, considering the geographic conditions of the highway, such as freeway hubs or ramps, the spatial correlation between some neighboring sensors will be weaker than non-adjacent sensors. Although the adjacency matrix can show the intuitive sensors' adjacency, it does not express the internal spatial influence. Therefore, we analyze the spatial correlation from two perspectives, global spatial correlation, and local spatial correlation, respectively.

1) LOCAL SPATIAL CORRELATION COMPONENT

It can be found that the more adjacent nodes, the smaller the weight of its node will be considered in the feature aggregation process during the convolution operation of equation 4. Therefore, in order to mine the spatial characteristics of the nodes themselves, we use the Approximate personalized propagation of neural predictions (APPNP) model to mine the local spatial correlation of traffic flow. APPNP [47] utilizes PageRank for node feature propagation, using PageRank to encode features

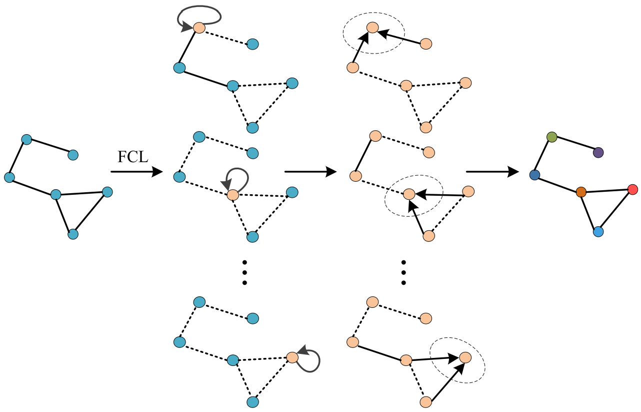


FIGURE 2. Local spatial process. It consists of a fully connected layer and a GCN layer, which is used to capture the local spatial correlation.

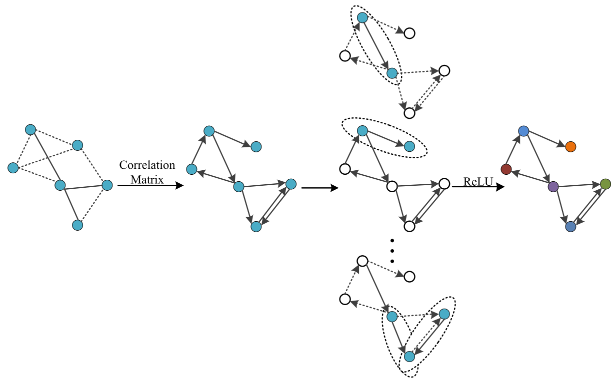


FIGURE 3. Global spatial process. It models the global spatial correlation between the distant roads. The process includes a correlation matrix construction and a GCN layer.

for each root node and increase the chance of transmission back to the root node. In this way, the model can balance the need of retaining local features and mining neighborhood features. The model calculation rules are as follows:

$$\begin{aligned} Z^{(0)} &= H = f_\theta(X) \\ Z^{(k+1)} &= (1-\alpha)\hat{A}Z^{(k)} + \alpha H \end{aligned} \tag{5}$$

where X represents the input of the nodes, f_θ denotes a neural network. We use f_θ to extract each sensor's self-features, α to represent the percentage of self-features. As shown in Fig. 2, we mine the spatial correlation of the nodes themselves and their first-order neighbors, so we set k to 1. The local spatial correlation mining formulas are as follows:

$$\begin{aligned} Z^{(0)} &= W_L^{(1)}X + b_L^{(1)} \\ LGCN(X, A) &= \sigma((1-\alpha)\hat{A}Z^{(0)} + \alpha Z^{(0)}) \end{aligned} \tag{6}$$

We use the FCL to extract node features. $W_L^{(1)}$ represents the weight matrix of the FCL, and $b_L^{(1)}$ is the bias matrix. $LGCN(\cdot)$ denotes the output of local spatial correlation.

2) GLOBAL SPATIAL CORRELATION COMPONENT

Spatial correlation does not only exist between neighboring sensors. As far as the whole road network is concerned, spatial correlation exists between sensors separated by long distances. So, we implicitly express the global spatial correlation of the road network. For sensors data, we use the Pearson correlation coefficient method to analyze the correlation between sensors in the studied network. We set a correlation threshold k to select high correlation sensors. If the correlation is greater than k, the correlation value is kept; otherwise, it is set to 0. In this way, we construct the correlation matrix C. And then, we use it to aggregate the highly correlated sensors' features through the GCN convolution method.

The correlation between two sensors is analyzed through the Pearson correlation coefficient method, which is calculated as follows:

$$C_{ij} = \frac{\sum_{t=1}^{T}(x_t^i - \bar{X}_i)(x_t^j - \bar{X}_j)}{\sqrt{\sum_{t=1}^{T}(x_t^i - \bar{X}_i)^2}\sqrt{\sum_{t=1}^{T}(x_t^j - \bar{X}_j)^2}} \tag{7}$$

where $X_i = (x_1^i, x_2^i, \ldots, x_t^i)$ represents the feature of traffic flow of sensor i, $\bar{X}_i$ is the mean value of X_i. Similarly, $X_j = (x_1^j, x_2^j, \ldots, x_t^j)$ represents the feature of traffic flow of sensor j, $\bar{X}_j$ is the mean value of X_j.

The node relationship described by the correlation matrix is a directed weighted graph, as shown in Fig. 3. The connections between nodes represent the influence weights, and the directions are the influence relationships. By the convolution of the correlation matrix and the feature matrix, the high correlation node features can be aggregated, which can deeply mine global spatial correlations. Therefore, the calculation rule of the global graph convolutional network based on the correlation matrix used in this paper is updated as follows:

$$GGCN(X, C) = \sigma(CXW_G^{(1)}) \tag{8}$$

where $W_G^{(1)}$ represents the weight matrix of the global graph convolutional network, $GGCN(\cdot)$ is the output of the global spatial correlation.

D. TEMPORAL CORRELATION

GRU is a mainstream neural network that addresses time series prediction problems. It can avoid gradient explosion and disappearance of RNN. GRU contains three parts: the input layer, the hidden layer, and the output layer. The core algorithm lies in the computation process in the unit block of the hidden layer, as shown in Fig. 4.

The local and global spatial correlation output is input into GRU separately. Take the local spatial correlation output as an example, and the GRU calculation rules are as follows:

$$r_t^l = \sigma(W_r^l[LGCN(X, A), h_{t-1}^l] + b_r^l) \tag{9}$$

$$z_t^l = \sigma(W_z^l[LGCN(X, A), h_{t-1}^l] + b_z^l) \tag{10}$$

$$\tilde{h}_t^l = tanh(W_{\tilde{h}}^l[LGCN(X, A), (r_t^l * h_{t-1}^l)] + b_{\tilde{h}}^l) \tag{11}$$

$$h_t^l = z_t^l * h_{t-1}^l + (1 - z_t^l) * \tilde{h}_t^l \tag{12}$$

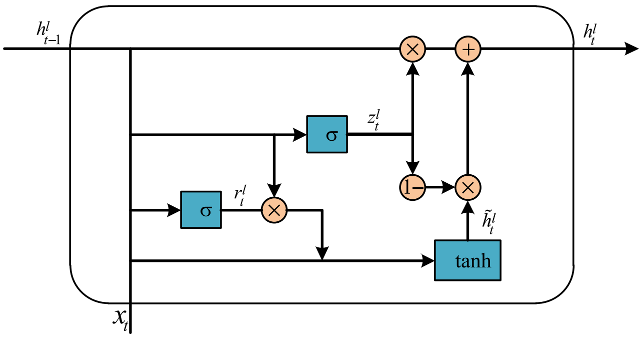


FIGURE 4. **Gated Recurrent Unit Network. The unit combines values at this moment and the output of previous moment to capture the temporal correlation.**

where r_t^l represents the reset gate of time t, W_r^l and b_r^l are the weight matrix and bias matrix of the reset gate, respectively. h_{t-1}^l is the output of the hidden layer at the previous moment.

For a given time slice, the unit first concatenates the output h_{t-1}^l of the hidden layer at the previous moment and the input $LGCN(X, A)$ at the current moment. And then, the data is transformed into [0,1] by the sigmoid function, which acts as the gate signals r_t^l and z_t^l. After that, the network uses the gate signal to selectively forget and save the information of h_{t-1}^l and $LGCN(X, A)$. In this way, GRU saves the traffic information of the previous moment and simultaneously combines the traffic context of the current moment, thus, achieves the temporal correlation.

E. T-LGGCN MODEL

To address the spatial-temporal correlation of traffic flow, especially the spatial correlation, we respectively construct the global spatial-temporal component and local spatial-temporal component to mine it, as shown in Fig. 1.

For the global spatial-temporal component, the global correlation matrix C is first calculated using the full amount of feature data. And then, we feed the input X_T^S and C into the global spatial correlation component to obtain $GGCN(\cdot)$, which is put into the GRU to extract the temporal correlation and get the output h_t^g of this component.

Next, the local spatial-temporal component is calculated. We employ the fully connected layer to extract the node features of input X_T^S and use the GCN to implement the aggregation and propagation of spatial features. The output $LGCN(\cdot)$ is directly input into GRU to obtain the output h_t^l of this component.

We use equation 13 to sum the output of the global spatial-temporal component and the local spatial-temporal component and input them into the Dense layer to output the prediction results.

$$y_{pre} = Dense(h_t^l + h_t^g) \tag{13}$$

where h_t^l is local spatial-temporal component output, h_t^g is global spatial-temporal component output.

In the training process of the model, we define the loss function of the model as follow:

$$Loss = ||y_{pre} - y_{true}|| + \beta L_{reg} \tag{14}$$

where we introduce L2 regularization to avoid the overfitting problems. And β is a hyperparameter, y_{pre} and y_{true} represent the predicted value of the model and the true value of the traffic flow, respectively.

Algorithm 1 outlines the training process of the T-LGGCN model.

Algorithm 1 The T-LGGCN Training Process

Input: Historical traffic parameters data X_T^S
Output: the predicted data y_{pre}
Parameters:
$W_L^{(1)}$, $b_L^{(1)}$, $W_G^{(1)}$, W_r^l, b_r^l, W_z^l, b_z^l, $W_{\tilde{h}}^l$, $b_{\tilde{h}}^l$, W_r^g, b_r^g, W_z^g, b_z^g, $W_{\tilde{h}}^g$, $b_{\tilde{h}}^g$
construct the global correlation matrix C by X_T^S
Initialize the parameters
for p-th epoch in total P training epochs do
  //local spatial-temporal component
  $Z^{(0)} = W_L^{(1)} X + b_L^{(1)}$
  $LGCN(X, A) = \sigma((1-\alpha)\hat{A}Z^{(0)} + \alpha Z^{(0)})$
  $r_t^l = \sigma(W_r^l[LGCN(X, A), h_{t-1}^l] + b_r^l)$
  $z_t^l = \sigma(W_z^l[LGCN(X, A), h_{t-1}^l] + b_z^l)$
  $\tilde{h}_t^l = tanh(W_{\tilde{h}}^l[LGCN(X, A), (r_t^l * h_{t-1}^l)] + b_{\tilde{h}}^l)$
  $h_t^l = z_t^l * h_{t-1}^l + (1 - z_t^l) * \tilde{h}_t^l$
  // local spatial-temporal component
  $GGCN(X, C) = \sigma(CXW_G^{(1)})$
  $r_t^g = \sigma(W_r^g[GGCN(X, C), h_{t-1}^g] + b_r^g)$
  $z_t^g = \sigma(W_z^g[GGCN(X, C), h_{t-1}^g] + b_z^g)$
  $\tilde{h}_t^g = tanh(W_{\tilde{h}}^g[GGCN(X, C), (r_t^g * h_{t-1}^g)] + b_{\tilde{h}}^g)$
  $h_t^g = z_t^g * h_{t-1}^g + (1 - z_t^g) * \tilde{h}_t^g$
  $y_{pre} = Dense(h_t^l + h_t^g)$ //component fusion
  Calculate the loss value in (14) and update the parameters by AdamOptimizer
End for

IV. EXPERIMENTS

A. EXPERIMENTS SETTINGS

1) DATASETS

We evaluated the performance of T-LGGCN through two real highway datasets: the PEMS04 dataset and the PEMS08 dataset. Both datasets are from the Caltrans Performance Evaluation System. We took the traffic speed data as the traffic flow information. PEMS04 was collected from San Francisco Bay. It contains 307 sensors, and the time is from January 1, 2018, to February 28, 2018. PEMS08 was from San Bernardino. It contains 170 sensors, and the time is from July 1, 2016, to August 31, 2016. The traffic data in both datasets were aggregated every 5 minutes. There were 288 records per day and no missing data in both datasets. We used the threshold Gaussian kernel to calculate the

TABLE 1. Prediction results of different models on PEMS04 and PEMS08.

Model	PEMS04				PEMS08			
	RMSE	MAE	Accuracy	R^2	RMSE	MAE	Accuracy	R^2
HA [15]	3.8387	1.7139	0.9402	0.7829	3.1179	1.3641	0.9513	0.7982
ARIMA [16]	6.7931	4.8239	0.8921	-	5.0256	3.5817	0.9201	-
SVR [18]	2.5629	1.2432	0.9601	0.9032	2.1578	1.0109	0.9663	0.9033
GRU [6]	2.5668	1.5293	0.9601	0.9029	1.6257	0.8733	0.9746	0.9451
GCN [34]	2.9232	1.8483	0.9545	0.8741	2.1545	1.2631	0.9663	0.9036
T-GCN [33]	2.1204	1.2027	0.9670	0.9337	1.9876	1.1737	0.9697	0.9216
T-LGGCN	**1.8627**	**1.0088**	**0.9710**	**0.9489**	**1.5287**	**0.8382**	**0.9761**	**0.9515**

adjacency matrix of both datasets. During the experiments, we use 80% of the data as the training set and 20% for testing. The normalization operation is performed first when the data is input.

2) PARAMETER SETTINGS

The experiments were implemented based on the TensorFlow framework. We used the Adam optimizer to optimize the model. Adam is an optimization algorithm designed to find the global optimal points. It combines the advantages of Momentum and RMSProp to minimize the loss function [48]. Adam Optimizer is an optimizer that implements Adam algorithm in TensorFlow which has been validated on a large number of neural network experiments. After repeated experiments, when the model performance reached the optimum, the main parameters were set as follows: α is set to 0.8, the GRU dimension is set to 64, the input step is set to 12, the learning rate is set to 0.001, the batch size is set to 32, and the training epoch is set to 100.

3) EVALUATION METRICS

We use the root mean square error (RMSE), mean absolute error (MAE), accuracy, coefficient of determination (R^2) as the evaluation metrics, and the detailed definitions are as follows.

$$RMSE = \sqrt{\frac{1}{IS}\sum_{i=1}^{I}\sum_{s=1}^{S}(y_i^s - y_i^{s\prime})^2} \tag{15}$$

$$MAE = \frac{1}{IS}\sum_{i=1}^{I}\sum_{s=1}^{S}|y_i^s - y_i^{s\prime}| \tag{16}$$

$$Accuracy = 1 - \frac{||Y - Y'||_F}{||Y||_F} \tag{17}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{I}\sum_{s=1}^{S}(y_i^s - y_i^{s\prime})^2}{\sum_{i=1}^{I}\sum_{s=1}^{S}(y_i^s - \bar{y})^2} \tag{18}$$

where I represents the input time steps, S is the numbers of sensors, y_i^s and $y_i^{s\prime}$ denote the real traffic speed data and predicted data respectively, Y and Y' are the sets of y_i^s and $y_i^{s\prime}$, $\bar{y}$ is the mean value of Y.

B. BASELINES

To demonstrate the effectiveness of our model, we select parametric methods (i.e., HA and ARIMA), non-parametric methods (i.e., SVR), and deep learning methods (i.e., GCN, GRU, and T-GCN) for comparison.

(1) HA [15]: HA is the simplest parametric method. The average value of historical traffic parameters is used as the model prediction result in our experiments.

(2) ARIMA [16]: The basic idea of ARIMA is to treat the time-series data as a random series and describe this series approximately with a certain mathematical model. A smoothness test is performed on the data first, and an ARMA model is fitted for prediction. We set the value of the autoregressive coefficient to 1, and the values of the difference order and the moving average to 0.

(3) SVR [18]: SVR is one of the common nonparametric prediction methods. The data are mapped to a multidimensional space using a nonlinear function, and then linear regression is performed on them.

(4) GCN [34]: GCN is a neural network method for spatial correlation analysis. A one-layer graph convolutional network is used, and the specific calculation process is detailed in Equation (4).

(5) GRU [6]: GRU is a variant of RNN. It is usually used to analyze the time-series data. The data is directly input into GRU, and the calculation process is shown in 3.4. We set the number of hidden layer neurons to 64.

(6) Temporal Graph Convolutional Network(T-GCN) [33]: T-GCN is a hybrid neural network. It consists of one-layer GCN and GRU. The feature data is input into GCN for calculation, and the output of the GCN layer is input into GRU to get the prediction results.

C. RESULTS OF TRAFFIC PREDICTION

Table 1 shows the prediction performance of our model and other baseline models on the PEMS04 and PEMS08 datasets for 5 minutes prediction task. It can be seen that T-LGGCN obtains the best prediction performance under all metrics. Compared with the basic model T-GCN, the RMSE error of our model is reduced by 12.1% on PEMS04 and by 23.1% on PEMS08.

From the overall prediction effect, the neural network methods have the best performance, and the non-parametric method SVR is weaker than the neural network but better than the parametric methods ARIMA and HA. It is shown that the neural network methods can better learn the non-smoothness of traffic flow. It can deeply explore internal features of the

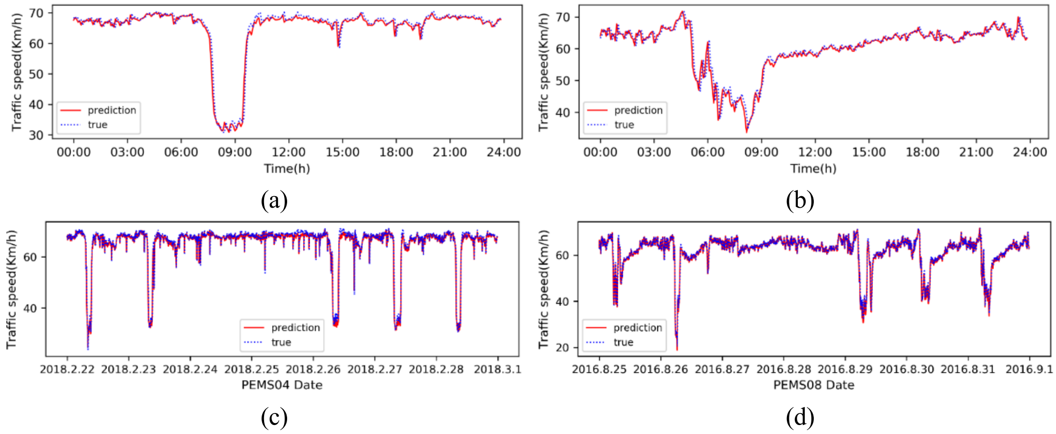


FIGURE 5. **The visualization results in test set. (a) T-LGGCN predictions for one day on PEMS04. (b) T-LGGCN predictions for one day on PEMS08. (c) T-LGGCN predictions for one week on PEMS04. (d) T-LGGCN predictions for one week on PEMS08.**

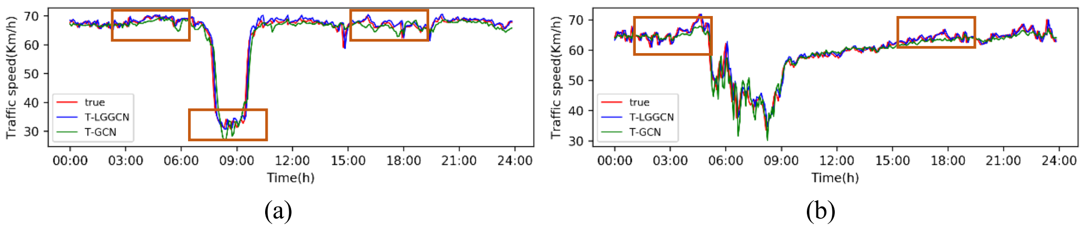


FIGURE 6. **The comparison of results in test set with T-GCN. (a) The comparison of results for one day on PEMS04. (b) The comparison of results for one day on PEMS08.**

traffic flow by training a large amount of data and effectively improve the prediction accuracy.

The RMSE of GRU on the PEMS04 and PEMS08 is 13.8% and 32.5% lower than the RMSE of GCN, respectively. GRU is used to capture temporal information. It can be found that temporal correlation is one of the important features of traffic flow. When we handle traffic prediction task, we need to take the temporal correlation into consideration.

We can also see that the hybrid neural networks outperform the single neural networks. Compared with GCN and GRU, the RMSE of T-LGGCN on the PEMS04 is improved by 36.2% and 27.4%. This indicates that the hybrid model takes into account both temporal and spatial features, which can more comprehensively explore the traffic flow characteristics.

It is noted from the table that T-LGGCN has a good ability to mine global and local spatial features. Compared with the T-GCN, the RMSE of T-LGGCN decreased by 12.1% and 23.1% on the PEMS04 and PEMS08. T-GCN only aggregates the spatial information of adjacent first-order sensors. T-LGGCN combines the spatial features of the node itself, first-order neighboring sensor spatial information, and long-range highly correlated sensor spatial information so as to analyze the spatial correlation comprehensively.

D. PREDICTION PERFORMANCE OF T-LGGCN

To have a deeper understanding of the T-LGGCN performance, we visualize the prediction results of one sensor data on two test sets separately, as shown in Fig. 5. We visualize the prediction results for one day and one week on two datasets. It is observed that our model fits better.

Meanwhile, we compare the prediction results of T-GCN with T-LGGCN, as shown in Fig. 6. Obviously, the fit of T-GCN is poorer than that of T-LGGCN. Although the T-GCN prediction results are mostly close to the true values, we can find that the T-GCN does not accurately capture the traffic flow characteristics when the traffic flow state changes drastically. The T-GCN fit curve behaves relatively smoothly in adjacent times (shown in the orange box). This is because T-GCN uses GCN to perform spatial feature aggregation operations, which may cause an over-smoothing problem, resulting in smoother predicted values at the peak. We improved the GCN spatial aggregation method, and it can be seen that our model effectively avoids this drawback.

V. DISCUSSION

A. SPATIAL CORRELATION ANALYSIS

T-LGGCN considers the spatial correlation between sensors in the global network. We use Pearson correlation coefficients to analyze the correlation between sensors based on traffic speed. The correlation values of each sensor are shown in Fig. 7. Taking the PEMS04 dataset as an example, Fig. 7(a) shows the correlation of the true first-order topological correlation and Fig. 7(b) shows the sensor correlation used in our model. The darker the color, the higher the correlation. It can be found that the correlation in Fig. 7(a) is significantly

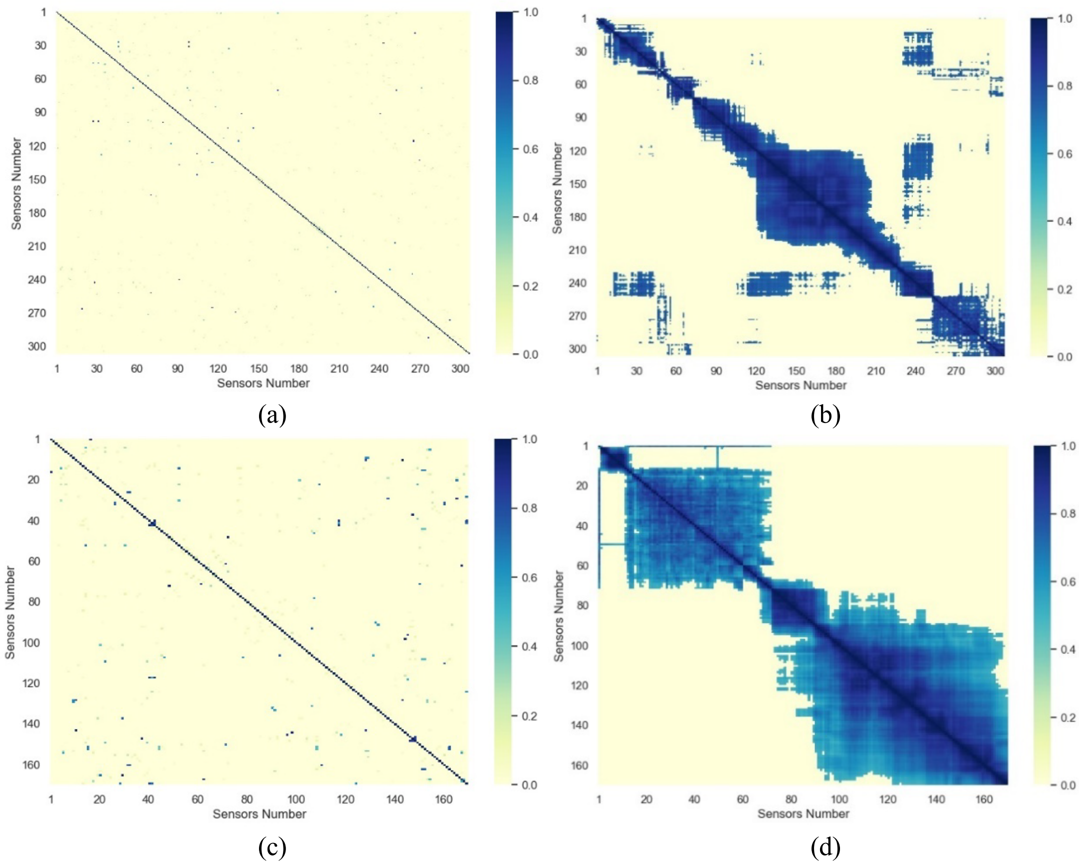


FIGURE 7. Sensor correlation analysis. (a) Correlation based on the adjacency on PEMS04. (b) Correlation based on traffic speed on PEMS04. (c) Correlation based on the adjacency on PEMS08. (d) Correlation based on traffic speed on PEMS08.

TABLE 2. Prediction results of different output steps on PEMS04 and PEMS08.

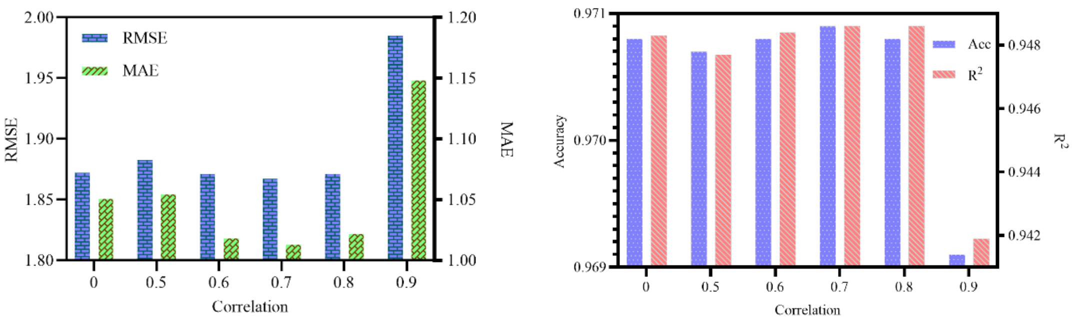
Steps	PEMS04				PEMS08			
	RMSE	MAE	Accuracy	R^2	RMSE	MAE	Accuracy	R^2
15min	2.6374	1.4202	0.9589	0.8975	2.1966	1.1448	0.9657	0.8997
30min	3.5305	2.0247	0.9450	0.8165	2.9303	1.6418	0.9543	0.8214
45min	4.1566	2.1695	0.9353	0.7459	3.4309	1.9336	0.9465	0.7548
60min	4.9374	3.0945	0.9233	0.6417	3.8942	2.2749	0.9392	0.6838

weaker than that in Fig. 7(b). This shows that the real topology does not adequately express the spatial correlations of traffic flow. The correlation matrix can further improve the ability of the model to capture global spatial features by reconstructing the correlation between sensors. In this manner, the model can have a better ability to capture global spatial features.

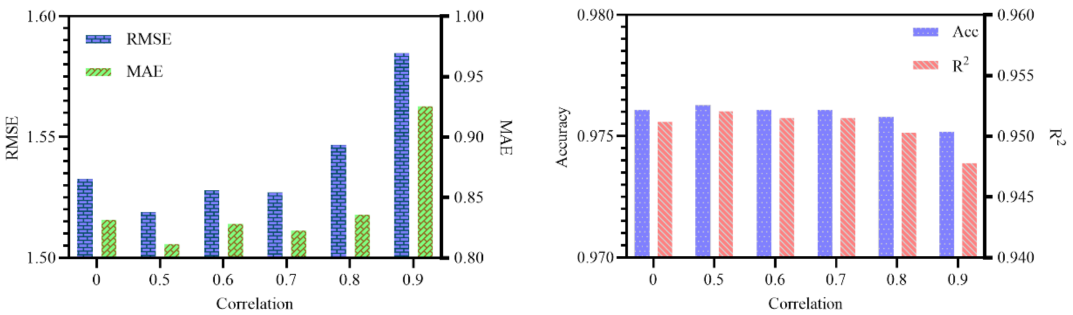
The sensors correlation k is an important factor affecting the prediction performance of the model. Fig. 8 shows the change of RMSE and MAE at different correlations. Model errors show an overall decreasing trend on the PEMS04 dataset. It indicates that discarding low correlation sensor information can effectively improve the model performance. When k is 0.7, the model error reaches a minimum and then increases, suggesting that 0.7 is the optimal correlation on the PEMS04. For the PEMS08, the model error tends to decrease at first with the increase of k. The model performance works best when k is 0.5. Subsequently, the error becomes larger when the correlation increases. It is noticed that the optimal k is different on different datasets. A possible reason for this is that the PEMS08 sensor number and the amount of data are small, and when the correlation increases, most sensors are abandoned, which results in the inability to capture the global spatial features effectively. Therefore, it is necessary to analyze the correlation of sensors specifically for different sizes of road networks.

B. LONG TERM PREDICTION ANALYSIS

The long-time prediction can provide appropriate guidance for those who are preparing to go out. To verify the long-term prediction capability of the model, we adjust the output time step of the model to validation. The results are shown in Table 2. As we can see from the table, the model performance decreases slightly when the prediction time step increases.



(a)



(b)

FIGURE 8. The impact of correlation on errors. (a) The impact of correlation on RMSE and MAE on PEMS04. (b) The impact of correlation on RMSE and MAE on PEMS08.

This indicates that T-LGGCN is relatively insensitive to the change of output step length. T-LGGCN can effectively predict traffic in the future to a certain extent.

VI. CONCLUSION

In this article, we propose a hybrid model integrating local and global spatial correlation (T-LGGCN) for traffic prediction. On the one hand, our model can consider the spatial correlation among distant sensors under the global road network. On the other hand, by adopting the strategy of separating the node feature analysis and the GCN local aggregation, we can improve the ability of local spatial correlation analysis. Compared with the baselines, the results show the importance of fully considering global spatial correlation and local spatial correlation. In conclusion, the T-LGGCN model successfully captures the spatial-temporal correlation of traffic flow and can be well applied to traffic flow prediction tasks.

REFERENCES

[1] Y. Yang, Z. Z. Yuan, D. Y. Sun, and X. L. Wen, "Analysis of the factors influencing highway crash risk in different regional types based on improved Apriori algorithm," *Adv. Transp. Stud.*, vol. 49, pp. 165–178, Nov. 2019, doi: 10.4399/97888255280911 3.

[2] Z. Lv, S. Zhang, and W. Xiu, "Solving the security problem of intelligent transportation system with deep learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4281–4290, Jul. 2021, doi: 10.1109/TITS.2020.2980864.

[3] J. Ye, J. Zhao, K. Ye, and C. Xu, "How to build a graph-based deep learning architecture in traffic domain: A survey," *IEEE Trans. Intell. Transp. Syst.*, early access, Dec. 29, 2021, doi: 10.1109/TITS.2020.3043250.

[4] D. Pavlyuk, "Feature selection and extraction in spatiotemporal traffic forecasting: A systematic literature review," *Eur. Transp. Res. Rev.*, vol. 11, no. 1, pp. 1–19, Dec. 2019, doi: 10.1186/s12544-019-0345-9.

[5] Q.-Q. Shen, Y. Cao, L.-Q. Yao, and Z.-K. Zhu, "An optimized discrete grey multi-variable convolution model and its applications," *Comput. Appl. Math.*, vol. 40, no. 2, pp. 1–26, Mar. 2021, doi: 10.1007/s40314-021-01448-z.

[6] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.

[7] H. Bohan and B. Yun, "Traffic flow prediction based on BRNN," in *Proc. IEEE 9th Int. Conf. Electron. Inf. Emergency Commun. (ICEIEC)*, Jul. 2019, pp. 320–323, doi: 10.1109/ICEIEC.2019.8784513.

[8] H. Lu, Z. Ge, Y. Song, D. Jiang, T. Zhou, and J. Qin, "A temporal-aware LSTM enhanced by loss-switch mechanism for traffic flow forecasting," *Neurocomputing*, vol. 427, pp. 169–178, Feb. 2021, doi: 10.1016/j.neucom.2020.11.026.

[9] W. Shu, K. Cai, and N. N. Xiong, "A short-term traffic flow prediction model based on an improved gate recurrent unit neural network," *IEEE Trans. Intell. Transp. Syst.*, early access, Jul. 14, 2021, doi: 10.1109/TITS.2021.3094659.

[10] J. Zhu, Q. Wang, C. Tao, H. Deng, L. Zhao, and H. Li, "AST-GCN: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting," *IEEE Access*, vol. 9, pp. 35973–35983, 2021, doi: 10.1109/ACCESS.2021.3062114.

[11] X. Kong, J. Zhang, X. Wei, W. Xing, and W. Lu, "Adaptive spatial-temporal graph attention networks for traffic flow forecasting," *Int. J. Speech Technol.*, pp. 3–6, Jul. 2021, doi: 10.1007/s10489-021-02648-0.

[12] D. Feng, Z. Wu, J. Zhang, and Z. Wu, "Dynamic global-local spatial-temporal network for traffic speed prediction," *IEEE Access*, vol. 8, pp. 209296–209307, 2020, doi: 10.1109/ACCESS.2020.3038380.

[13] Y. Ren, D. Zhao, D. Luo, H. Ma, and P. Duan, "Global-local temporal convolutional network for traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, early access, Oct. 7, 2020, doi: 10.1109/tits.2020.3025076.

[14] K. He, X. Chen, Q. Wu, S. Yu, and Z. Zhou, "Graph attention spatial-temporal network with collaborative global-local learning for citywide mobile traffic prediction," *IEEE Trans. Mobile Comput.*, early access, Sep. 1, 2020, doi: 10.1109/tmc.2020.3020582.

[15] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin, ''Deep learning on traffic prediction: Methods, analysis and future directions,'' *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 10, 2021, doi: 10.1109/TITS.2021.3054840.
[16] Z. Zhang, Y. Li, H. Song, and H. Dong, ''Multiple dynamic graph based traffic speed prediction method,'' *Neurocomputing*, vol. 461, pp. 109–117, Oct. 2021, doi: 10.1016/j.neucom.2021.07.052.
[17] Z. Li, G. Xiong, Y. Tian, Y. Lv, Y. Chen, P. Hui, and X. Su, ''A multi-stream feature fusion approach for traffic prediction,'' *IEEE Trans. Intell. Transp. Syst.*, early access, Oct. 7, 2020, doi: 10.1109/tits.2020.3026836.
[18] Q. Li, Z. Han, and X. M. Wu, ''Deeper insights into graph convolutional networks for semi-supervised learning,'' in *Proc. 32nd AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 3538–3545.
[19] L. Zhao, Y. Song, C. Zhang, and Y. Liu, ''T-GCN: A temporal graph convolutional network for traffic prediction,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020, doi: 10.1109/TITS.2019.2935152.
[20] A. Ermagun and D. Levinson, ''Spatiotemporal traffic forecasting: Review and proposed directions,'' *Transp. Rev.*, vol. 38, no. 6, pp. 786–814, Nov. 2018, doi: 10.1080/01441647.2018.1442887.
[21] J. Liu and W. Guan, ''A summary of traffic flow forecasting methods,'' *J. Highway Transp. Res. Develop.*, vol. 21, no. 3, pp. 82–85, Mar. 2004.
[22] S. V. Kumar and L. Vanajakshi, ''Short-term traffic flow prediction using seasonal ARIMA model with limited input data,'' *Eur. Transp. Res. Rev.*, vol. 7, no. 3, pp. 1–9, Sep. 2015, doi: 10.1007/s12544-015-0170-8.
[23] H. Yang, X. Li, W. Qiang, Y. Zhao, W. Zhang, and C. Tang, ''A network traffic forecasting method based on SA optimized ARIMA–BP neural network,'' *Comput. Netw.*, vol. 193, Jul. 2021, Art. no. 108102, doi: 10.1016/j.comnet.2021.108102.
[24] S. Liu, Y. Lin, C. Luo, and W. Shi, ''A novel learning method for traffic flow forecasting by seasonal SVR with chaotic simulated annealing algorithm,'' in *Proc. IEEE 6th Int. Conf. Comput. Commun. Syst. (ICCCS)*, Apr. 2021, pp. 205–210, doi: 10.1109/ICCCS52626.2021.9449161.
[25] F. Su, H. Dong, L. Jia, Y. Qin, and Z. Tian, ''Short-term traffic forecasting using self-adjusting k-nearest neighbours,'' *IET Intell. Transp. Syst.*, vol. 12, no. 1, pp. 41–48, Feb. 2018, doi: 10.1049/IET-its.2016.0263.
[26] Y. Gu, W. Lu, X. Xu, L. Qin, Z. Shao, and H. Zhang, ''An improved Bayesian combination model for short-term traffic prediction with deep learning,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1332–1342, Mar. 2019, doi: 10.1109/TITS.2019.2939290.
[27] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, ''Traffic flow prediction with big data: A deep learning approach,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2014, doi: 10.1109/tits.2014.2345663.
[28] Z. He, C. Y. Chow, and J. D. Zhang, ''STCNN: A spatio-temporal convolutional neural network for long-term traffic prediction,'' in *Proc. IEEE Int. Conf. Mob. Data Manag.*, Jun. 2019, pp. 226–233, doi: 10.1109/MDM.2019.00-53.
[29] T. Bogaerts, A. D. Masegosa, J. S. Angarita-Zapata, E. Onieva, and P. Hellinckx, ''A graph CNN-LSTM neural network for short and long-term traffic forecasting based on trajectory data,'' *Transp. Res. C, Emerg. Technol.*, vol. 112, pp. 62–77, Mar. 2020, doi: 10.1016/j.trc.2020.01.010.
[30] H. Zhu, Y. Xie, W. He, C. Sun, K. Zhu, G. Zhou, and N. Ma, ''A novel traffic flow forecasting method based on RNN-GCN and BRB,'' *J. Adv. Transp.*, vol. 2020, pp. 1–11, Oct. 2020, doi: 10.1155/2020/7586154.
[31] Z. Wang, X. Su, and Z. Ding, ''Long-term traffic prediction based on LSTM encoder-decoder architecture,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6561–6571, Oct. 2021, doi: 10.1109/TITS.2020.2995546.
[32] D. Han, X. Yang, G. Li, S. Wang, Z. Wang, and J. Zhao, ''Highway traffic speed prediction in rainy environment based on APSO-GRU,'' *J. Adv. Transp.*, vol. 2021, pp. 1–11, Aug. 2021, doi: 10.1155/2021/4060740.
[33] X. Chen, H. Chen, Y. Yang, H. Wu, W. Zhang, J. Zhao, and Y. Xiong, ''Traffic flow prediction by an ensemble framework with data denoising and deep learning model,'' *Phys. A, Stat. Mech. Appl.*, vol. 565, Mar. 2021, Art. no. 125574, doi: 10.1016/j.physa.2020.125574.
[34] L. Yuting, Z. Ming, M. Chicheng, B. Bo, Z. Zhiheng, Y. Kai, W. Guanghui, and Y. Guiying, ''Graph neural network,'' *SCIENTIA SINICA Math.*, vol. 50, no. 3, p. 367, Mar. 2020, doi: 10.1360/n012019-00133.
[35] T. N. Kipf and M. Welling, ''Semi-supervised classification with graph convolutional networks,'' in *Proc. 5th Int. Conf. Learn. Represent. (ICLR) Conf. Track*, 2017, pp. 1–14.
[36] C. Zheng, X. Fan, C. Wang, and J. Qi, ''GMAN: A graph multi-attention network for traffic prediction,'' in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, Apr. 2020, pp. 1234–1241, doi: 10.1609/aaai.v34i01.5477.
[37] W. Li, X. Wang, Y. Zhang, and Q. Wu, ''Traffic flow prediction over muti-sensor data correlation with graph convolution network,'' *Neurocomputing*, vol. 427, pp. 50–63, Feb. 2021, doi: 10.1016/j.neucom.2020.11.032.
[38] X. Shi, H. Qi, Y. Shen, G. Wu, and B. Yin, ''A spatial-temporal attention approach for traffic prediction,'' *IEEE Trans. Intell. Transport. Syst.*, vol. 22, no. 8, pp. 4909–4918, Apr. 2020, doi: 10.1109/TITS.2020.2983651.
[39] H. Zheng, F. Lin, X. Feng, and Y. Chen, ''A hybrid deep learning model with attention-based conv-LSTM networks for short-term traffic flow prediction,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 11, pp. 6910–6920, Nov. 2021, doi: 10.1109/tits.2020.2997352.
[40] T. Zhang, W. Ding, T. Chen, Z. Wang, and J. Chen, ''A graph convolutional method for traffic flow prediction in highway network,'' *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–8, Jul. 2021, doi: 10.1155/2021/1997212.
[41] Y. Huang, Y. Weng, S. Yu, and X. Chen, ''Diffusion convolutional recurrent neural network with rank influence learning for traffic forecasting,'' in *Proc. 18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun./13th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2019, pp. 1–16.
[42] K. Guo, Y. Hu, Z. Qian, H. Liu, K. Zhang, Y. Sun, J. Gao, and B. Yin, ''Optimized graph convolution recurrent neural network for traffic prediction,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 2, pp. 1138–1149, Feb. 2021, doi: 10.1109/TITS.2019.2963722.
[43] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, ''Graph wavenet for deep spatial-temporal graph modeling,'' in *Proc. Int. Jt. Conf. Artif. Intell. (IJCAI)*, Aug. 2019, pp. 1907–1913, doi: 10.24963/ijcai.2019/264.
[44] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, ''Adaptive graph convolutional recurrent network for traffic forecasting,'' in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2020, pp. 1–16.
[45] M. Lv, Z. Hong, L. Chen, T. Chen, and S. Ji, ''Temporal multi-graph convolutional network for traffic flow prediction,'' *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3337–3348, Jun. 2020, doi: 10.1109/TITS.2020.2983763.
[46] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, ''The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains,'' *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013, doi: 10.1109/MSP.2012.2235192.
[47] J. Klicpera, A. Bojchevski, and S. Günnemann, ''Predict then propagate: Graph neural networks meet personalized PageRank,'' in *Proc. 7th Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–15.
[48] D. P. Kingma and J. L. Ba, ''Adam: A method for stochastic optimization,'' in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR) Conf. Track*, 2015, pp. 1–15.

SIYUN FENG is currently pursuing the M.S. degree with the School of Information Science and Technology, Nantong University, Nantong, China. Her main research interests include intelligent transportation technology, graph convolutional networks, and deep learning.

JIASHUANG HUANG received the Ph.D. degree from the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, in 2020. From 2018 to 2019, he was a Visiting Scholar with the University of Wollongong (UOW), Wollongong, NSW, Australia. He is currently a Lecturer with the School of Information Science and Technology, Nantong University. He has published more than 20 research peer-reviewed journal and conference papers, including the IEEE Transactions on Medical Imaging, the IEEE Journal of Biomedical and Health Informatics, *Medical Image Analysis*, *Bioinformatics*, AAAI, and MICCAI. His recent research is to analyze the brain network by using machine learning methods.

QINQIN SHEN received the Ph.D. degree from the School of Rail Transportation, Soochow University, in 2021. She is currently a Senior Experimentalist with the School of Transportation and Civil Engineering, Nantong University. She has published over ten articles in high-level journals, including *Computational and Applied Mathematics*, *Computers and Mathematics with Applications*, and *Numerical Algorithms*. Her research interests include intelligence transportation and numerical computation.

QUAN SHI (Member, IEEE) received the M.S. and Ph.D. degrees in management information systems from the University of Shanghai for Science and Technology, Shanghai, China, in 2005 and 2011, respectively. He is currently a Professor with the School of Transportation, Nantong University. His research interests include the development of signal and image processing, intelligent information processing, and big data techniques for computer.

ZHENQUAN SHI received the master's degree from the School of Computer Science and Technology, University of Shanghai for Science and Technology, in 2009, and the Ph.D. degree in management information systems from the School of Management, University of Shanghai for Science and Technology, in 2021. He is currently working with the School of Transportation, Nantong University. He has published eight relevant articles in high-level journals. His main research interests include intelligent transportation and deep learning.