Received July 22, 2020, accepted August 7, 2020, date of publication August 21, 2020, date of current version September 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3018452

A General Traffic Flow Prediction Approach Based on Spatial-Temporal Graph Attention

CONG TANG[1], (Student Member, IEEE), JINGRU SUN[1], (Member, IEEE), YICHUANG SUN[2], (Senior Member, IEEE), MU PENG[1], AND NIANFEI GAN[3]

[1]College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China
[2]School of Engineering and Computer Science, University of Hertfordshire, Hatfield AL10 9AB, U.K.
[3]College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China

Corresponding author: Jingru Sun (jt_sunjr@hnu.edu.cn)

This work was supported in part by the Science and Technology Project of Hunan Provincial Communications Department, China, under Grant 2018037, and in part by the National Nature Science Foundation of China under Grant 61674054.

ABSTRACT Accurate and reliable traffic flow prediction is critical to the safe and stable deployment of intelligent transportation systems. However, it is very challenging since the complex spatial and temporal dependence of traffic flows. Most existing works require the information of the traffic network structure and human intervention to model the spatial-temporal association of traffic data, resulting in low generality of the model and unsatisfactory prediction performance. In this paper, we propose a general spatial-temporal graph attention based dynamic graph convolutional network (GAGCN) model to predict traffic flow. GAGCN uses the graph attention networks to extract the spatial associations among nodes hidden in the traffic feature data automatically which can be dynamically adjusted over time. And then the graph convolution network is adjusted based on the spatial associations to extract the spatial features of the road network. Notably, the information of rode network structure and human intervention are not required in GAGCN. The forecasting accuracy and the generality are evaluated with two real-world traffic datasets. Experimental results indicate that our GAGCN surpasses the state-of-the-art baselines on one of two datasets.

INDEX TERMS Traffic flow forecasting, graph attention networks, graph convolutional network, dynamic spatial-temporal.

I. INTRODUCTION

The speedy growth of vehicles has brought tremendous pressure on urban traffic, which has seriously affected people's daily lives. Therefore, it is necessary to find an effective technical means to improve traffic management efficiency and ease traffic problems. As a critical part of Intelligent Transportation System (ITS) [1], short-term traffic flow prediction can predict the next 5-60 minutes' traffic conditions of the road section, and provides great helps in many areas, such as signal control, traffic guidance, path planning.

In the real world, traffic flow data is affected by many factors, with the properties of being highly complex and nonlinear, thus accurate traffic prediction is very challenging. After decades of research, traffic flow prediction methods were mainly classified into two approaches, model-driven and data-driven. Model-driven methods are also called parametric methods, such as time-series models, which have well-established theoretical background. However, such methods require plenty of parameters and assumptions to apply to the entire network, which makes their prediction performance unsatisfactory. Recently, with the improvement of transportation infrastructure, different data collection technologies such as monitoring points, detectors, have provide a mass of available data for traffic flow prediction. Data-driven approaches can be separated into two subclasses: machine learning and deep learning [2]–[4]. Common machine learning methods are inadequate when processing high-dimensional data and also rely on detailed feature engineering. Therefore, this type of methods have fragile generality. Deep learning models, for instance convolutional neural networks, long short-term memory neural (LSTM) networks and their combination, have achieved great success in traffic prediction [5]. Their success is mainly due to the good performance when dealing with highly nonlinear, dynamic arbitrary precision, and multidimensional problems. In traffic networks, the detector nodes

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

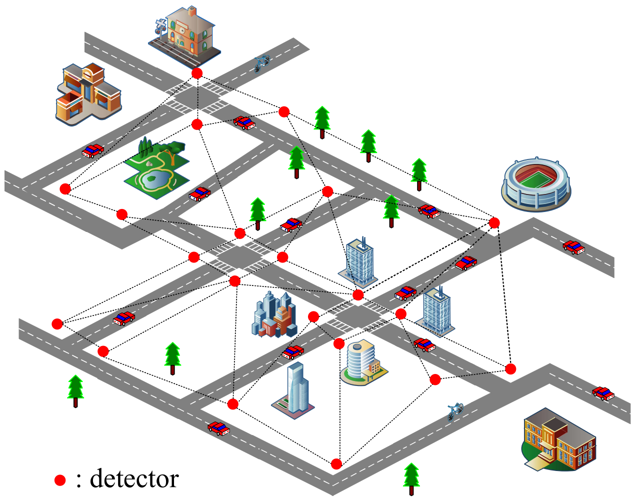


FIGURE 1. Topology graph of traffic network. The detector nodes deployed in the road network can be regarded as vertices on the topology graph. We contact the nodes at each location so that the road network can be abstracted into a topology graph. Then, we predict the vehicle speed of each detector in the road network of the next period of time.

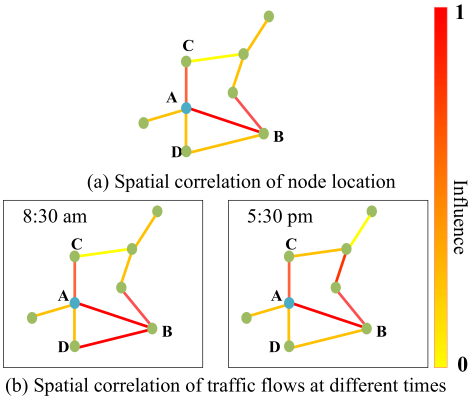

C
A
B
D
(a) Spatial correlation of node location
8:30 am
C
A
B
D
5:30 pm
C
A
B
D
(b) Spatial correlation of traffic flows at different times
1
Influence
0

FIGURE 2. The spatial-temporal association graph of traffic nodes. There are spatial associations among nodes, and this associations change over time.

are delpoyed on the traffic roads, which form a topological graph with a non-Euclidean structure, as shown in Fig. 1. Observations obtained at nearby locations influence each other, resulting in spatial local association. But traditional deep learning methods are not suitable for processing non-Euclidean data. An ideal way to process non-Euclidean structured data is to use graph convolutional network (GCN) [6], whose essential purpose is to collect the spatial features of the topological graph. Graph convolution includes vertex domain and spectral domain, but when using vertex domain to extract features, bacause the neighbors extracted from each vertex are different, the calculation process must be performed for each vertex. The spectral domain is the focus of GCN research, which regards the features of each node as signals on the graph, and studies the features of the graph through spectral analysis to realize the topological graph convolution operation. However, existing traffic flow prediction models based on graph convolutional network use fixed distance information between nodes, when constructing the Laplacian matrix of the graph, and ignore the dynamic changes in the association/weights among nodes. Even though some models considered that the association among nodes will change in time, the method of dynamically adjusting the spatial association/weights among nodes is not aimmed at topology graphs. Moreover, they all rely on the road network configuration, such as the position of the detectors deployed on the road, and the distance among the detectors. These methods can not reflect the true spatial-temporal properties of the tracific roads, and lack of generality.

The limitations of existing traffic prediction models based on graph convolutional network encourage us to design a novel framework of traffic flow prediction. We have two observations regarding this problem. First, except distance, many other factors should also be considered when it comes to spatial associations/weights among nodes, and we should research the dynamic spatio-temporal associations of traffic flow data with Non-Euclidean structure from the perspective of topological graph, as shown in Fig. 2. Second, we should reduce the subjective participation of human and try to predict the traffic prevalence of the road network without knowing the road network structure in advance, if we want to make the model adapt to different road network structures and improve the generality of the prediction model.

In this paper, we propose a spatial-temporal graph attention based dynamic graph convolutional network (GAGCN), which is employed to predict the road network traffic flow based on spatial-temporal feature and has better generality and prediction accuracy than previous approaches. The main contributions of ours can be summarized:

- We develop a graph attention mechanism to dynamically adjust the spatial associations/weights among nodes over time. We identify the associations/weights among nodes hidden in the traffic data through the graph attention network, and the Laplacian matrix of the road network topology graph is dynamically adjusted in line with the spatio-temporal features of the traffic data.
- The structure information of the road network, such as the position of the detector, is not required and we only need traffic flow features data in our model. The proposed method can reduce the error of people's prior knowledge in previous models and also improve the generality of the model.
- Large-scale experiments are performed on two universal traffic datasets. The experimental results confirm the prediction accuracy and generality of our model on different datasets. The source code can be found at https://github.com/sam101340/GAGCN-BC-20200720.

The remainder of this paper includes: Section II introduces the research and development on traffic flow prediction. Section III introduces the spatial-temporal graph attention based dynamic graph convolutional network. Section IV exhibitions our experimental and results analysis and conclude our work in Section V.

II. BACKGROUND

A. TRAFFIC FLOW PREDICTION

In recent years, many excellent-performing prediction models have been proposed to assist signal control, traffic guidance, and path planning. Traffic data has the characteristic of flowing and is a typical time series, given traffic data $\mathcal{T}$ to predict the traffic parameters (such as speed, traffic flow or occupancy) Y of the next H time points with the traffic data of the past P time points of all the nodes in the road network, $Y = (y^1, y^2, \ldots, y^N)^T \in R^{N\times H}$, and $y^i = (y^i_{P+1}, y^i_{P+2}, \ldots, y^i_{P+H}) \in R^H$ denotes the future parameters of node i from $P + 1$.

The time-series analysis model uses mathematical formulas to model past behavior, and then uses the obtained model to predict future results. ARIMA [7] is a classic statistical model in time-series analysis, which is widely used in traffic prediction [8]. References [9] and [10],extended the spatial domain to the ARIMA time series model to obtain the spatio-temporal autoregressive integrated moving average (STARIMA) models. And STARIMA achieved good results in the field of traffic flow prediction [11]. However, the time-series analysis model is a purely inductive method, which requires some ideal prior assumptions. And it is difficult to satisfy these assumptions in the real world because of the inherent complexity of traffic data. Therefore, the above methods often perform poorly in practical applications.

Machine learning methods, for instance support vector regression (SVR) [12], [13], k-nearest neighbor algorithm (KNN) [14], K-means [15] have solid mathematical foundation and can help us handle more complex traffic data. To achieve the theoretical advantage of these methods, the premises are to choose the appropriate parameters and conduct detailed feature engineering.

Recently, deep learning models with good learning capabilities and deep network structures have developed rapidly. Based on its good performance, deep learning models have also made great progress in traffic flow prediction. Such as stacked autoencoder (SAEs) [16] and Deep belief network (DBN) [17], [18] are all based on deep learning models.

However, the fully connected networks are not sufficient to extract the spatial-temporal features of traffic flows, they only process one region every time, and the configuration of their neurons cannot meet this demand. Convolution neural networks (CNNs) [19] as a typical deep neural network, has made many breakthroughs in image processing. Recently, some researchers have used CNNs to capture spatial features in traffic prediction tasks [20]. Gated recurrent unit networks (GRUs) [21], [22] and long short-term memory neural networks (LSTM) [23] are both good at processing time-serie [24] and have also been used in traffic flow prediction. And then, researchers combined CNN and LSTM networks to propose a functional level fusion architecture CLTFP [25] for short-term traffic prediction by combining CNN with LSTM networks. Later, a convolutional LSTM was proposed by [26], which embedded an extended fully connected LSTM into convolutional layer LSTM (FC-LSTM) [27]. Compare with the above methods, CLTFP and FCLSTM can obtain better prediction performance. Convolutional neural networks can effectively capture the spatial features of grid data. However, convolutional neural networks cannot extract the spatial features of the road network, when we consider the traffic road network and the detectors deployed on the road network as a topology graph. Because the number of neighbors of each vertex in the topology graph is different, so the convolution operation cannot be performed with the same size convolution kernel.

However, it is hoped that the spatial features can be effectively extracted on the data structure such as the topology graph, so GCN has become the research focus. Existing graph convolutional network can be fall into two types according to the convolutional operator: vertex domain and spectral domain. Vertex domain finds the neighbors adjacent for each vertex to extract the spatial features on the topology graph [28], however the neighborhood of each vertex is different, each vertex needs to be processed alone. Spectral domain uses the theory of the spectral graph to convolve topological graph. Bruna *et al.* [29] proposes a general graph convolution framework, and then Defferrard, Bresson, and Vandergheynst [6] approximates it with Chebyshev polynomial to decrease the computational complexity of the model.

Recently, many researchers use GCN to predict traffic flow [30]–[33]. Spatial-Temporal Graph Convolutional Networks (STGCN) [34] was proposed, which constructs a fixed Laplacian matrix based on the spatial distance among the detector nodes and human experience. Further, Attention Based Spatial-Temporal Graph Convolutional Networks (ASTGCN) [35] uses an attention mechanism [36] to capture the dynamic associations among nodes. However, the construct of Laplacian matrix required by the GCN of the above methods are all dependent on the spatial distance among the,detector nodes in the road network and the human experience, which make the mo,del have great limitations.

B. ATTENTION MECHANISM

Attention mechanism, as a new technology, has developed rapidly in recent years, and is widely used in fields such as natural language processing, speech recognition and others. Attention mechanisms allow neural networks to focus on input data and provide helpful information for the current task. Then, an alignment model is proposed to evaluate the match between input and output [37]. After that, a neural network architecture consisting of two memory networks is proposed, which can model the semantic association and relationship between each word and two entities [36]. Based on the above research, Graph Attention Networks (GATs) [38] is proposed. GATs does not need to know the structure of the graph in advance and only focuses on the feature data of the nodes and uses the self-attention layer to specify the weights among the nodes. It has achieved the best level in the industry in three difficult benchmarks. In traffic flow prediction, to extract spatial features, Liu *et al.* perform 2D convolution on each feature map to obtain the corresponding attention matrix, perform maximum average pooling on each feature map, and use the result as the input of the feed-forward neural network to obtain the channel attention [39]. Recently, Zheng *et al.* use scaling point product attention mechanism to obtain spatial-temporal attention, and transformer attention from encoder to decoder [31].

To improve the generality of prediction model and decrease errors caused by human experience, we propose a spatial-temporal graph attention based dynamic graph convolutional network. Our framework employs the graph attention networks to find spatial dependencies hidden in traffic data and adjust the Laplacian matrix in time.

III. METHODOLOGY

In this part, before we present our model in detail, we will introduce some background and explain the definitions that appear in our article.

A. PROBLEM DEFINITION

For traffic data $\mathcal{T}$ on all nodes, suppose p represents a certain time point in the m^{th} time slice, $p \in (1, 2, \ldots, P)$ and $m \in (1, 2, \ldots, M)$. We use $x_p^{i,c} \in R^C$ to denote the values of all the features of node i at time p in m^{th} time slice, and $x_p^{i,c_j} \in R$ represents the value of the j^{th} feature of node i at time p in m^{th} time slice. So $T_p = \left(x_p^{1,c}, x_p^{2,c}, \ldots, x_p^{N,c}\right)^T \in R^{N\times C}$ (the traffic data of all nodes at p time), $T_m = (T_1, T_2, \ldots, T_P)^T \in R^{P\times N\times C}$ denotes the values of all the features of all the nodes at time slice m. $\mathcal{T} = (T_1, T_2, \ldots, T_M)^T \in R^{M\times P\times N\times C}$ (the traffic data of all the nodes over M time slices). We use $v_p^i \in R$ to denote the value of the velocity of node i at time p, and $v_p^i = T_p^{i,C_3}$.

B. GCN

Compare traffic network to a graph $G = (V, e, A)$, where V, e, A represent a set of vertices (detector positions), a set of edges, the adjacency matrix of G respectively. The vehicle speed V observed by the detector can be regarded as a graph signal that is defined on the graph, where v^i is the signal value at the i^{th} node. The theoretical core of the graph convolution is the feature decomposition of the Laplacian matrix of the graph. Graph Laplacian $L = D - A$, where D is the degree matrix of the vertices. Further, the Laplacian matrix can be normalized as: $L = I_n - D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$ (I_n is an identity matrix). In general, the performance of graph convolutional networks mainly depends on the quality of A [40]. Eigenvalue decomposition of the Laplacian matrix can obtain its eigenvector matrix U and eigenvalue matrix A. And the Laplacian matrix can be expressed as: $L = U\Lambda U^T$, where $\Lambda \in R^{N\times N}$ is a diagonal matrix, $U \in R^{N\times N}$ is Fourier basis. Graph convolution filter $g_\theta = diag(\theta)$ parameterized by $\theta = R^N$. Hence, the graph convolution of x defined in the Fourier domain is:

$$g_\theta *_G x = U g_\theta U^T x, \tag{1}$$

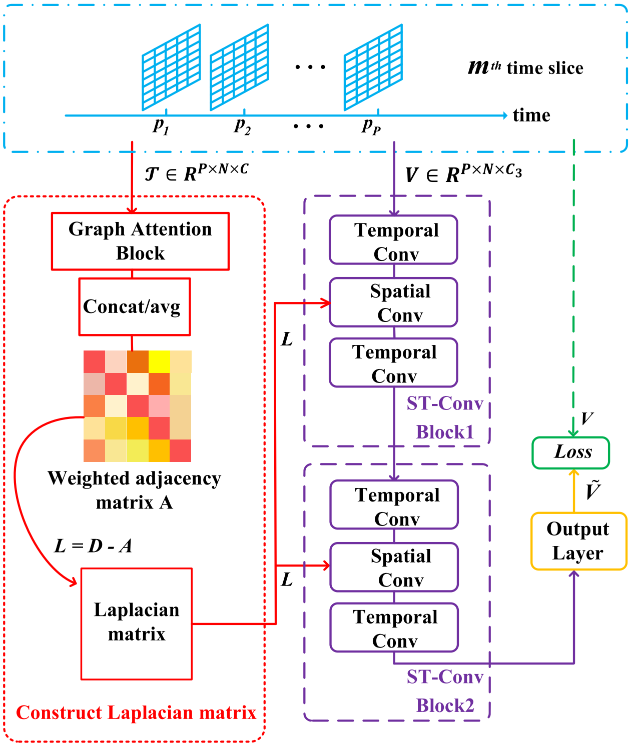


FIGURE 3. The framework of GAGCN.

where $*_G$ denotes a graph convolution operation and $U^T x$ is the graph Fourier transform of x.

However, it is expensive to use (1) to calculate the eigenvector matrix of L, when the structure of the graph is very complicated. To solve this problem, we use the Chebyshev polynomial approximation to reduce computational K^{th} complexity [41], and (1) can be further defined as:

$$g_{\theta'} *_G x \approx \sum_{k=0}^{K} \theta'_k T_k(\tilde{L})x, \tag{2}$$

where $\theta' \in R^K$ is a vector of Chebyshev coefficients. $\tilde{L} = \frac{2L}{\lambda\max} - I_n$, $\lambda_{\max}$ is the maximum eigenvalue of the L. $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$, where $T_0(x) = 1$, $T_1(x) = x$.

C. DYNAMIC SPATIAL-TEMPORAL GRAPH ATTENTION GRAPH CONVOLUTIONAL NETWORKS

The core of GCN is based on the spectral decomposition of Laplacian matrix. Building an accurate Laplacian matrix is very helpful in improving prediction accuracy. First we propose a method of constructing dynamic Laplacian matrix with traffic data of nodes, and then introduce a novel dynamic spatial-temporal GCN for traffic speed prediction.

As shown in Fig. 3, Our framework is compose of three modules: a Laplacian matrix module constructed from graph attention networks, two spatial-temporal convolution blocks and an output layer. Taking the m^{th} time series as an example, we use the constructed traffic data $\mathcal{T} \in R^{P\times N\times C}$ as the input of graph attention networks to obtain the weighted

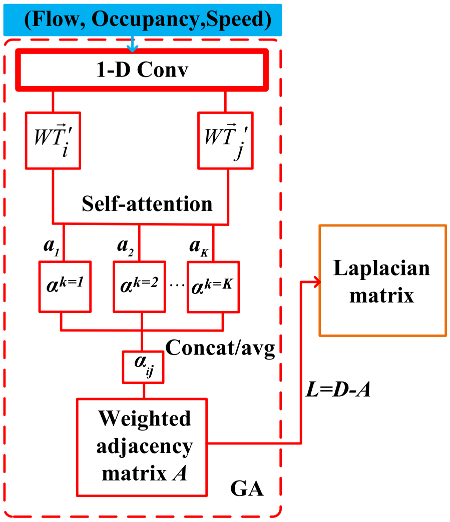


FIGURE 4. Get Laplacian matrix by Graph Attention Networks.

adjacency matrix A of the graph. Then we use $L = D - A$ to obtain the Laplacian matrix L, and spectrally decompose L to obtain the graph convolution kernel required for graph convolution. $V \in R^{P\times N\times C_3}$ is the only input data of the "ST-Conv block". There are two temporal convolution blocks and one spatial convolution block in "ST-Conv block". And the spatial convolution block is located in the middle of the two temporal convolution blocks. The output layer includes a temporal convolutional and a fully connected layer. Finally, the output layer integrates all features to get the final prediction result. $\tilde{V}$.

1) OBTAINING LAPLACIAN MATRIX BY GRAPH ATTENTION NETWORKS

In the previously proposed method of predicting traffic flow by using GCN, in order to obtain a weighted adjacency matrix, they need specific road network information (such as detectors position information), which reduces the generality of the model. And they also need to use their prior knowledge (such as selecting a specific mathematical model to control the sparsity of the adjacency matrix) to construct the weighted adjacency matrix A.

In this paper, we implicitly assign different weights to nodes in the neighborhood by graph attention networks and do not rely on understanding the road network structure in advance (such as the spatial position information of the detector nodes). Fig. 4 shows the details of obtaining the Laplacian matrix through the graph attention networks. Taking into account the impact of changes of time and traffic conditions on the relationship among nodes, we construct the data into M time slice and batch input to the graph attention network. Take the m^{th} time slice as an example, data at the p^{th} time point $T_p = \{\vec{T}_1, \vec{T}_2, \vec{T}_3, \ldots, \vec{T}_N\}$, ($\vec{T}_i \in R^C$, N denotes the number of the detectors, C represents the feature number of traffic data) and the m^{th} time slice contains P time points. Each time point of data will get a association matrix by the graph attention network. Finally, we average the P relational matrices to obtain the association/weights matrix (weighted adjacency matrix) of the m^{th} time slice.

As shown in Fig. 4, for data T_p, a One-dimensional convolution layer is employed to transform the input features into higher level features on each node in the initial step to obtain $T_p' = \{\vec{T}_1', \vec{T}_2', \vec{T}_{3,\ldots,}' \vec{T}_N'\}$, $\vec{T}_p' \in R^{N\times C'}$, ($C'$ is the number of filters) and the time complexity is $O_1(N \cdot K \cdot C \cdot C')$ where K is the size of the convolution kernel. Before obtaining the attention coefficients among the nodes on the graph, the high-dimensional feature data T_p' is linearly transformed by the parameter matrix $W \in R^{C'\times 1}$. The attention coefficient can be expressed as:

$$e_{ij} = \vec{a}(W\vec{T}_i', W\vec{T}_j'), \tag{3}$$

where $\vec{a} \in R^{2C'}$ is a single feedforward neural network, $e_{ij} \in R^{P\times N\times N}$. Equation (3) indicates the degree of influence of the traffic condition of node j on node i and its time complexity is $O_2(2N \cdot K \cdot C')$. And this process is applied to each node on the graph. By implementing masked attention, the structural information of the graph is incorporated into the mechanism, and only compute e_{ij} for nodes $j \in N_i$ where N_i is some neighborhood of node i in the graph. In order to compare the attention coefficients e_{ij} between different nodes, we normalize them with a nonlinear function:

$$\alpha_{ij} = softmax(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k\in N_i} e_{ik}}. \tag{4}$$

Then applying the LeakyReLU nonlinearity, the coefficients computed by the attention mechanism can be expressed as:

$$\alpha_{ij} = \frac{\exp(LeakyReLU(\vec{a}^T[W\vec{T}_i' \oplus W\vec{T}_j']))}{\sum_{k\in N_i} \exp(LeakyReLU(\vec{a}^T[W\vec{T}_i' \oplus W\vec{T}_j']))}, \tag{5}$$

where $\oplus$ represents the concatenation operation.

Once the normalized attention coefficient α_{ij} obtained (ie. the weight relationship between node i and node j), then we can get each element of the weighted adjacency matrix:

$$\mathcal{W}_{ij} = \begin{cases} \alpha_{ij}, & i \neq j \\ 1, & otherwise, \end{cases} \tag{6}$$

We have only used one attention factor calculation mode as described above, (as shown in Fig. 5). But in order to stabilize the self-attention learning process, it is beneficial to use multi-head attention, as shown in Fig. 6. At this time, the time complexity of the calculation process in Fig. 6 is $O_3(\mathcal{K} \cdot (M^2 \cdot K \cdot C' + 2M \cdot K \cdot C'))$. We eventually get the $\mathcal{K}$ attention coefficient average:

$$\mathcal{W}_{ij} = \begin{cases} \sigma(\frac{1}{\mathcal{K}} \sum_{k=1}^{\mathcal{K}} \alpha_{ij}^k), & i \neq j \\ 1, & otherwise, \end{cases} \tag{7}$$

where $\mathcal{K}$ denotes the number of attention computations, σ is a nonlinear transformation. α_{ij}^k is the attention coefficient computed by the k^{th} attention mechanism $\vec{a}^k$. Finally, we get all

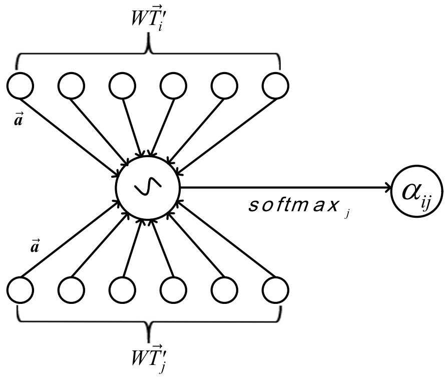


FIGURE 5. Constructing attention coefficients α_{ij}. Our model linearly transform the input traffic data T to obtain a higher-dimensional representation, and then employ the attention mechanism $\vec{a}(W\vec{T}_i', W\vec{T}_j')$ and the non-linear activation function $softmax_j$ to obtain the attention coefficients α_{ij}.

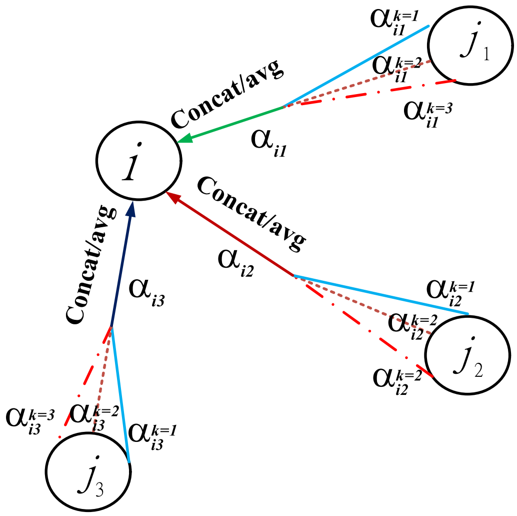


FIGURE 6. The explanation for multihead attention. For nodes i and j_1, multihead attention can be considered as using several different attention mechanisms $\vec{a}$ to obtain the attention coefficient between nodes i and j_1. Among them, each attention mechanism $\vec{a}$ will get a corresponding α_{ij_1}, and then average each α_{ij_1} as the final attention coefficients.

the $\mathcal{W}_{ij}$ and construct the final weighted adjacency matrix A.

$$A = \begin{bmatrix} 1 & \mathcal{W}_{12} & \cdots & \mathcal{W}_{1n} \\ \mathcal{W}_{21} & 1 & \cdots & \mathcal{W}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{W}_{n1} & \mathcal{W}_{n2} & \cdots & 1 \end{bmatrix} \tag{8}$$

Then we get the Laplacian matrix L according to $L = D - A$. In the end, we can get the time complexity $O(\mathcal{K} \cdot (N^2 \cdot K \cdot C' + 2N \cdot K \cdot C') + N \cdot K \cdot C' \cdot (C+2))$ of graph attention network. Treating $\mathcal{K}$, K, C and C' as constants, the time complexity of the graph attention networks becomes $O(N^2)$.

2) JOINT EXTRACTION OF SPATIAL FEATURES BY GRAPH ATTENTION NETWORKS AND GCN

Our purpose is to predict the vehicle speed Y of the next H time slices with the traffic data of the past P time points of all the nodes in the road network ($Y = (y^1, y^2, \ldots, y^N)^T \in R^{N\times H}$, and $y^i = (y^i_{P+1}, y^i_{P+2}, \ldots, y^i_{P+H}) \in R^H$ denotes the future velocity of node i from $P+1$). All the features are selected as the input of the graph attention networks, and only the velocity is selected as the input of the GCN. The input of the graph convolutional layer can be expressed as $x \in R^{N\times P\times C_{in}}$, where N, P, C_{in} represent the number of the node, time point and channel, respectively.

We create the Laplacian matrix L of the graph with weighted adjacency matrix A, and perform feature decomposition on L. We set $K = 1$, and use a layer-wise linear formulation to stack multiple localized graph convolutional layers. With the first-order approximation of graph Laplacian and in this linear formulation of a GCN we can further approximate $\lambda_{max} \approx 2$ [42]. Under these approximations (2) simplifies to:

$$\begin{aligned} g_{\theta'} *_G x &\approx \theta'(\frac{2L}{\lambda\max} - I_n)x \\ &\approx \theta_0' x - \theta_1'(D^{-\frac{1}{2}}AD^{-\frac{1}{2}})x, \end{aligned} \tag{9}$$

where θ_0', θ_1' are two shared parameters, θ_0' and θ_1' can be replaced by a single parameter θ' by letting $\theta' = \theta_0' = -\theta_1'$. To avoid repeated application of this operator, which may lead to numerical instability and explosion/disappearance gradients [42], A and D are renormalized by $\tilde{A} = A + I_n$ and $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ separately. Then (9) can be alternatively expressed as:

$$g_{\theta'} *_G x \approx \theta'(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}})x. \tag{10}$$

Finally, the graph convolution in (10) can be rewritten as:

$$y_i = \sum_{i=1}^{C_{in}} \theta_{ij}'(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}})x_i, \quad j = 1, 2, 3, \ldots, C_{out}, \tag{11}$$

where $y_i \in R^{N\times c_{out}}$, C_{in}, C_{out} represent the size of input feature map and output feature map, respectively (in this case, $C_{in} = 1$).

According to the reference [33], the time complexity of the graph convolution network is $O(K(N \cdot C_{in} + N \cdot C_{in} \cdot C_{out}))$, where K is the approximate order number of Chebyshev polynomial. Taking K, C_{in} and C_{out} as constants, the time complexity of the graph convolution module becomes $O(N)$.

3) GATED CNNs FOR EXTRACTING TEMPORAL FEATURES

As shown in Fig. 7, the traditional 2-D convolution operation of CNN is used to obtain short-term features of traffic flow. We use $\Upsilon \in R^{N\times P\times C_{in}}$ as the input to the temporal convolution layer (N, P represent the size of the spatial and temporal dimensions, respectively). The convolutional kernel $\Gamma \in R^{K\times C_{in}\times 2C_{out}}$ will map the input to an output element $[A, B] \in R^{N\times(P-k+1)\times(2C_{out})}$ (A, B have the same size of channels, which is half of the total size of channels). The temporal convolution can be defined as:

$$\Gamma *_{\mathcal{T}} \Upsilon = A \odot \sigma(B), \tag{12}$$

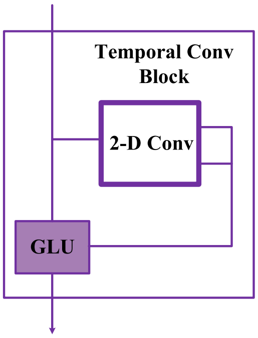


FIGURE 7. Temporal Convolution Block. The temporal convolutional block contains a 2-D causal convolution followed by gated linear units (GLU).

where A, B are input of gates in GLU, respectively, $\odot$ is Hadamard product. The sigmoid gate σ (B) is a gating mechanism that controls which information in A can be passed to the next layer.

4) SPATIAL-TEMPORAL CONVOLUTIONAL BLOCK

In order to improve the prediction accuracy, we fuse the spatial convolution block with the temporal convolution block, and jointly process the time series of the graph structure by using the space-time convolution block.

IV. EXPERIMENTS

A. DATASET DESCRIPTION

We validate our model on the real-world traffic dataset PeMS (collected by the California Department of Transportation). The dataset contains key attributes such as overall flow for each detector node, average lane occupancy, and average vehicle speed. It also contains the geometric information of the detector and the corresponding timestamp, as detailed below:

- **PeMSD4:** It refers to the traffic data in San Francisco Bay Area, containing 3848 detectors on 29 roads. We randomly select 50/100 detectors and select data for major traffic routes during the workdays from May 1, 2012 to June 30, 2012. The traffic data are aggregated and output by each detector every 5 minutes.
- **PeMSD7:** Its traffic data for the Los Angeles area includes 39,000 detectors. We randomly select 100/206 detectors and select data for major traffic routes during the workdays from May 1, 2012 to June 30, 2012. The traffic data are aggregated and output by each detector every 5 minutes.

We select three traffic features: traffic volume, average lane occupancy, and speed, and the time interval of the data is 5 minutes. Therefore, each node contains 288 data points per day. The missing values are filled by the linear interpolation. In addition, z-score normalization is performed on three different traffic attribute data, respectively.

B. EXPERIMENTAL SETTINGS

All experiments are performed and tested on the Window's operating system (CPU: Intel Core i7-8700 @ 3.20GHz, GPU: NVIDIA GeForce GTX 1070Ti). In order to get the best parameters on validation, grid search strategy is selected. In this paper, the historical time window of all experiments is 60 minutes, i.e. 12 observation data points (P = 12) for predicting the next one hour (H = 3; 6; 9; 12).

During the training phase, the RMSprop is used to optimize the mean square error. All baselines are also trained for 50 epochs with batch size as 30. The initial learning rate is 10^{-4} with a decay rate of 0.7 after every 5 epochs. We use the 1st-order approximation, both the spatial and the temporal convolution kernel size are set to 1.

1) EVALUATION INDICATORS AND BASELINES

a: EVALUATION INDICATORS

We use Mean Absolute Error (MAE), Mean Absolute Error Percentage (MAPE) and Root Mean Square Error (RMSE) to evaluate the performance of different models. They are defined as:

$$MAE = \frac{1}{n}\sum_{t=1}^{n}|v_t - \tilde{v}_t| \tag{13}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{v_t - \tilde{v}_t}{v_t}\right| \times 100 \tag{14}$$

$$RMSE = \left[\frac{1}{n}\sum_{t=1}^{n}(v_t - \tilde{v}_t)^2\right]^{\frac{1}{2}} \tag{15}$$

where v_t is the detected vehicle speed, and $\tilde{v}_t$ is the predicted vehicle speed.

b: BASELINES

We compare our GAGCN with the following seven baselines:

- **HA:** Historical Average method. Here, we use the average value of the last one hour to predict the next value.
- **VAR** [43]: Vector Auto-Regression is a time series model, which easy to analyze multiple time series.
- **CNN-LSTM** [44]: 2-D CNN with LSTM is a traditional spatial-temporal model.
- **DCRNN** [30]: A diffusion convolutional recurrent neural network, which captures the spatial dependence between nodes on the graph by bidirectional random walk.
- **STGCN** [34]: A spatial-temporal graph convolutional model, which base on fixed Laplacian matrix.
- **MSTGCN** [35]: Multi-Component spatial-temporal graph convolution model, which gets rid of the spatial-temporal attention.
- **ASTGCN** [35]: Attention Based Spatial-Temporal Graph Convolutional Networks, which can learn the dynamic spatial-temporal associations of traffic data.
- **Graph WaveNet** [32]: Combining graph convolution and diffusion convolution.
- **GMAN** [31]: A graph multi-attention network with Encoder-Decoder structure.

TABLE 1. Forecasting error given on PeMSD7 and PeMSD4.

	PeMSD7-100			PeMSD7-206			PeMSD4-50			PeMSD4-100		
Model	**MAE**	**MAPE (%)**	**RMSE**	**MAE**	**MAPE (%)**	**RMSE**	**MAE**	**MAPE (%)**	**RMSE**	**MAE**	**MAPE (%)**	**RMSE**
HA	6.24	14.44	12.87	6.49	15.89	11.42	2.78	6.28	5.88	4.02	6.35	6.01
VAR	6.11	13.20	11.03	5.92	14.32	10.69	2.57	5.31	4.75	3.87	5.13	4.99
CNN-LSTM	5.35	11.74	9.49	4.67	12.59	8.96	2.51	4.35	3.83	3.87	5.13	4.99
DCRNN	4.11	10.03	8.89	3.63	11.38	8.04	1.65	3.55	3.90	**1.62**	**3.50**	3.72
STGCN	3.60	8.63	6.52	3.63	11.38	8.04	2.07	4.41	4.20	2.09	4.25	4.15
MSTGCN	4.24	10.27	7.82	4.69	11.56	8.91	2.73	4.85	4.61	2.31	4.98	4.51
ASTGCN	4.02	10.02	7.62	4.38	11.32	8.35	2.73	4.85	4.61	1.99	4.32	4.39
Graph WaveNet	3.75	10.42	7.26	4.38	11.32	8.35	1.66	3.85	3.85	1.75	3.92	3.88
GMAN	3.69	9.95	7.21	3.73	10.60	7.55	**1.53**	**3.33**	**3.47**	1.64	3.60	**3.70**
MTGNN	3.84	9.70	6.78	3.73	10.60	7.55	3.20	7.78	5.90	3.33	7.76	5.97
NGAGCN(ours)	3.60	8.59	7.02	3.72	8.12	6.97	1.98	4.01	4.17	1.97	4.10	4.01
GAGCN(ours)	**3.55**	**8.47**	**6.38**	**3.37**	**6.12**	**6.37**	1.88	3.83	4.05	1.91	4.03	3.90

TABLE 2. Mean and standard deviation of the two datasets.

Dataset	**Mean**	**STD**
PeMSD7-100	59.83	13.79
PeMSD7-206	59.82	14.11
PeMSD4-50	62.95	8.42
PeMSD4-100	62.83	8.22

- **MTGNN** [33]: A general graph network for multi-time series forecasting.

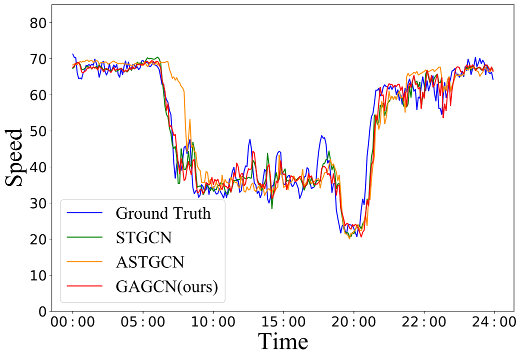


FIGURE 8. Speed prediction in PeMSD7-100.

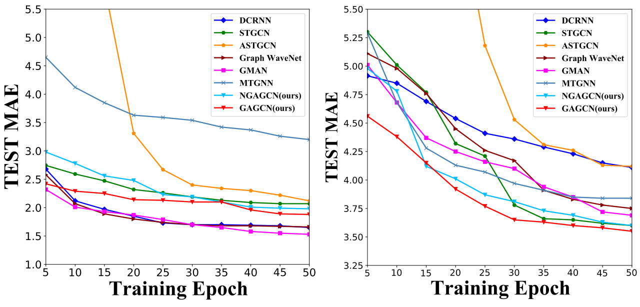
C. COMPARISON AND ANALYSIS OF RESULTS

1) FORECASTING ACCURACY

We validated our model and ten baselines on the datasets PeMSD7 and PeMSD4 and the mean and standard deviation of these two datasets are showed in Table 2. Table 1 shows the average speed of the four time steps (15 minutes, 30 minutes, 45 minutes and 1 hour) of each verification baseline in the next hour. It can be seen from Table 1 that the prediction results of the traditional time series analysis methods (HA, VAR) are usually not ideal, demonstrating those methods' limited abilities of modeling nonlinear and complex traffic data. The comparison results show that traditional deep learning models like CNN-LSTM have better prediction performance than traditional time series analysis methods. The prediction results of graph convolutional network models such as STGCN and MSTGCN without injecting the attention mechanism in the model are better than traditional spatial-temporal deep learning models such as CNN-LSTM. When the road network structure is complicated and the positions of the detectors are relatively random, the prediction accuracy of ASTGCN is lower than traditional spatial-temporal models without attention mechanism, such as STGCN. Compare with DCRNN, Graph WaveNet and GMAN on PeMSD7, GAGCN does not inject road network graph structure information, however it can still achieve the best prediction performance. As shown in Table 1, on two different datasets, the prediction performance of GAGCN is better than MTGNN, which shows that the method of using graph attention network to extract the correlation among nodes hidden in various traffic feature data is more effective. The prediction results of STGCN, ASTGCN, and GAGCN on a certain day are shown in Fig. 8. And they are all based on the graph convolution network method.we can easy to observe that GAGCN is closest to the ground truth, compared with other two methods. Fig. 9 shows the change of the prediction performance of each method with the increase of the training epoch. In general, as the training cycle increases, the prediction error also gradually decreases and eventually stabilizes.

2) EFFECT OF GRAPH ATTENTION MECHANISM

To investigate the effect of graph attention mechanism in our model, we designed a version of GAGCN that contains road information, named Non-universal spatial temporal graph attention based dynamic graph convolutional network (NGAGCN). In this part, we compare and analyze three models: GAGCN, NGAGCN, STGCN. The main difference between the three models is the way in which the associations/ weights among road network nodes is extracted. Among them, STGCN only uses the structural information of the road network (the distance between the detection points), NGAGCN combines the graph attention mechanism and the



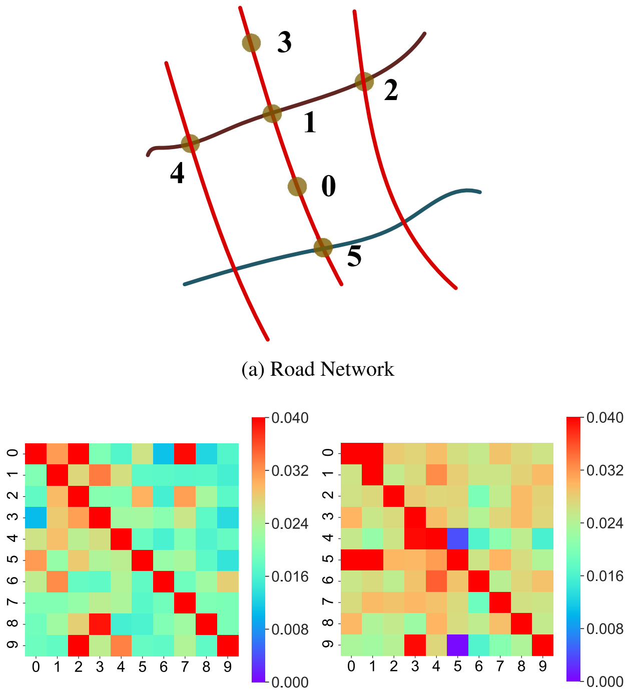
(a) PeMSD4-50 (b) PeMSD7-100

FIGURE 9. Performance changes of different methods as the training epochs increases.

road network structure information, and GAGCN only uses the graph attention mechanism to extract the associations/weights relationship among the nodes hidden in the traffic feature data, and the data does not contain any road network structure information. It can be seen from Table 1 that the prediction accuracy of STGCN which only using road network structure information is the worst, and there is no generality match with other road networks. NGAGCN, which combines graph attention mechanism and road network structure information, reduces the generality of the model, but achieves better prediction results than STGCN. This proves that it is effective to inject a graph attention mechanism into the model to extract the spatial features of traffic flow. Finally, compared with NGAGCN and STGCN, GAGCN that does not contain the information of road network structure not only improves the versatility of the model, but also achieves the best prediction performance. This further proves that graph attention network can effectively mine the dynamic association features between hidden nodes in traffic feature data.

3) SPATIAL WEIGHT MATRIX

Based on the change of traffic conditions caused by time changes, our model can pay attention to and adjust to the association/weights among nodes in the road network according to the changing traffic conditions at the nodes. Fig. 10 (b-c) shows a part of the spatial association/weights matrix among nodes obtained by our graph attention networks. The i^{th} row represents the association between each detector and the i^{th} detector. Taking 0^{th} and 1^{th} detectors as an example, we can find from Fig. 10 (b-c) that the intensity of the influence of 1^{th} on 0^{th} is greater than 0^{th} on 1^{th} detector. This result is consistent with the real situation, because the traffic condition of detector 1 in the traffic network is affected by detectors 0, 2, 3, and 4, while detector 0 is only affected by detectors 1 and 5, as shown in Fig. 10 (a). Therefore, our model not only obtains the best prediction performance and the highest generality, but also shows an interpretability advantage.



(a) Road Network

(b) Time Step–50 (c) Finally

FIGURE 10. The attention matrix obtained by graph attention network.

4) BENEFITS OF GATs BUILDING LAPLACIAN MATRIX

Obtaining the weighted adjacency matrix is the key to constructing the graph Laplacian matrix. The existing model is mainly based on the positional distance among the detectors in the road network, and then uses its own prior-knowledge explicitly to assign a weight relationship to each detector node, which may cause a lot of error, and lower the prediction accuracy. We pay attention to the traffic data of each detector, and let the graph attention networks to find out the hidden relationship among each detector, that is, implicitly specify the weight relationship among nodes, and construct a real weighted adjacency matrix. Our method (GAGCN) reduces the error caused by human experience and does not require the spatial position information of the detector. The results in Table 1 indicates that the error of our model is the smallest under the same conditions. This shows that our model captures the influence relationship among detectors more accurately than STGCN, in other words, it better captures the spatial features of the road network. Although MTGNN does not inject graph structure into the model, and uses traffic characteristic data to find the correlation among nodes, however the experimental results on tasks with a large time span are not very ideal, and the experimental process is relatively complicated. It can be concluded that the Laplacian matrix using GATs to construct not only improves the generality, but also ensures the accuracy of the model.

V. CONCLUSION

In this paper, we propose a novel traffic flow prediction model called GAGCN. GAGCN employs graph attention networks to dynamically obtain weighted adjacency matrix of road network graph. Its ''spatial-temporal convolution'' uses graph convolutional network to extract the spatial features of the road network, and employs gated temporal convolution to excavate temporal features. The information of rode network structure and human intervention are not required in GAGCN, and it can flexibly face various complex road networks. Lots of experiments and comparisons were performed on two actual datasets, compared with traditional methods that use the distance between nodes and human experience, GAGCN has better accuracy and versatility, and can also achieve better prediction performance when compared with models without injected graph structure.

In real life, there are many factors that can affect the traffic conditions of the road network. These include natural and unnatural factors, such as weather, social events, air quality, and many other factors. In the future, we will consider some external influence factors to further improve the prediction accuracy.

ACKNOWLEDGMENT

The authors would like to thank Prof. Z. Xiao from Hunan University and Dr. B. Xia from Huazhong University of Science and Technology for their suggestions to modify the article.

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] X. Peng, H. Zhu, J. Feng, C. Shen, H. Zhang, and J. T. Zhou, "Deep clustering with sample-assignment invariance prior," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Dec. 31, 2020, doi: 10.1109/TNNLS.2019.2958324.

[3] P. Hu, D. Peng, X. Wang, and Y. Xiang, "Multimodal adversarial network for cross-modal retrieval," *Knowl.-Based Syst.*, vol. 180, pp. 38–50, Sep. 2019.

[4] P. Hu, D. Peng, Y. Sang, and Y. Xiang, "Multi-view linear discriminant analysis network," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5352–5365, Nov. 2019.

[5] J. Zhang, F. Chen, and Q. Shen, "Cluster-based LSTM network for short-term passenger flow forecasting in urban rail transit," *IEEE Access*, vol. 7, pp. 147653–147671, 2019.

[6] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.

[7] G. Comert and A. Bezuglov, "An online change-point-based model for traffic parameter prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 3, pp. 1360–1369, Sep. 2013.

[8] J. Zheng and M. Huang, "Traffic flow forecast through time series analysis based on deep learning," *IEEE Access*, vol. 8, pp. 82562–82570, 2020.

[9] P. E. Pfeifer and S. J. Deutrch, "A three-stage iterative procedure for space-time modeling Phillip," *Technometrics*, vol. 22, no. 1, pp. 35–47, 1980.

[10] P. E. Pfeifer and S. J. Deutsch, "Variance of the sample space-time correlation function of contemporaneously correlated variables," *SIAM J. Appl. Math.*, vol. 40, no. 1, pp. 133–136, Feb. 1981.

[11] P. Duan, G. Mao, W. Liang, and D. Zhang, "A unified spatio-temporal model for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3212–3223, Sep. 2019.

[12] Y.-S. Jeong, Y.-J. Byon, M. M. Castro-Neto, and S. M. Easa, "Supervised weighting-online learning algorithm for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1700–1707, Dec. 2013.

[13] J. Xiao, Z. Xiao, D. Wang, J. Bai, V. Havyarimana, and F. Zeng, "Short-term traffic volume prediction by ensemble learning in concept drifting environments," *Knowl.-Based Syst.*, vol. 164, pp. 213–225, Jan. 2019.

[14] F. G. Habtemichael and M. Cetin, "Short-term traffic flow rate forecasting based on identifying similar traffic patterns," *Transp. Res. C, Emerg. Technol.*, vol. 66, pp. 61–78, May 2016.

[15] R. García-Ródenas, M. L. López-García, and M. T. Sánchez-Rico, "An approach to dynamical classification of daily traffic patterns," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 3, pp. 191–212, Mar. 2017.

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

[17] Y. Jia, J. Wu, and Y. Du, "Traffic speed prediction using deep learning method," in *Proc. IEEE 19th Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2016, pp. 1217–1222.

[18] 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.

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

[20] D. Yu, Y. Liu, and X. Yu, "A data grouping CNN algorithm for short-term traffic flow forecasting," in *Proc. Asia–Pacific Web Conf.* Springer, 2016, pp. 92–103.

[21] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*. [Online]. Available: http://arxiv.org/abs/1412.3555

[22] 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.

[23] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000.

[24] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: A deep learning approach for short-term traffic forecast," *IET Intell. Transp. Syst.*, vol. 11, no. 2, pp. 68–75, Mar. 2017.

[25] Y. Wu and H. Tan, "Short-term traffic flow forecasting with spatial-temporal correlation in a hybrid deep learning framework," 2016, *arXiv:1612.01022*. [Online]. Available: http://arxiv.org/abs/1612.01022

[26] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[27] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018.

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

[29] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," 2013, *arXiv:1312.6203*. [Online]. Available: http://arxiv.org/abs/1312.6203

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

[31] C. Zheng, X. Fan, C. Wang, and J. Qi, "Gman: A graph multi-attention network for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.* 2020, pp. 1234–1241.

[32] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 1907–1913.

[33] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," 2020, *arXiv:2005.11650*. [Online]. Available: https://arxiv.org/abs/2005.11650

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

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

[36] X. Feng, J. Guo, B. Qin, T. Liu, and Y. Liu, "Effective deep memory networks for distant supervised relation extraction," in *Proc. IJCAI*, Aug. 2017, pp. 4002–4008.

[37] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: http://arxiv.org/abs/1409.0473

[38] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*. [Online]. Available: http://arxiv.org/abs/1710.10903

[39] Q. Liu, B. Wang, and Y. Zhu, "Short-term traffic speed forecasting based on attention convolutional neural network for arterials," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 33, no. 11, pp. 999–1016, Nov. 2018.

[40] X. Peng, J. Feng, S. Xiao, W.-Y. Yau, J. T. Zhou, and S. Yang, "Structured AutoEncoders for subspace clustering," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 5076–5086, Oct. 2018.

[41] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Appl. Comput. Harmon. Anal.*, vol. 30, no. 2, pp. 129–150, Mar. 2011.

[42] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*. [Online]. Available: http://arxiv.org/abs/1609.02907

[43] E. Zivot and J. Wang, "Vector autoregressive models for multivariate time series," *Modeling Financial Time Series With S-PLUS*. 2006, pp. 385–429.

[44] Y. Liu, H. Zheng, X. Feng, and Z. Chen, "Short-term traffic flow prediction with Conv-LSTM," in *Proc. 9th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2017, pp. 1–6.

CONG TANG (Student Member, IEEE) is currently pursuing the master's degree with the College of Computer Science and Electronic Engineering, Hunan University.

His research interests include data mining and intelligent transportation technology.

JINGRU SUN (Member, IEEE) received the Ph.D. degree in computer science and technology from Hunan University, China, in 2014.

She is currently an Assistant Professor with the College of Computer Science and Electronic Engineering, Hunan University. She has published more than ten articles. Her research interests include intelligent transportation, memristors, and its application to storage and neural networks.

YICHUANG SUN (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees from Dalian Maritime University, Dalian, China, in 1982 and 1985, respectively, and the Ph.D. degree from the University of York, York, U.K., in 1996, all in communications and electronics engineering.

He is currently a Professor of communications and electronics, the Head of the Communications and Intelligent Systems Research Group, and Head of the Electronic, Communication and Electrical Engineering Division, School of Engineering and Computer Science, University of Hertfordshire, U.K. He has published over 350 articles and contributed ten chapters in edited books. He has also published four text and research books: *Continuous-Time Active Filter Design* (CRC Press, USA, 1999), *Design of High Frequency Integrated Analogue Filters* (IEE Press, U.K., 2002), *Wireless Communication Circuits and Systems* (IET Press, 2004), and *Test and Diagnosis of Analogue, Mixed-signal and RF Integrated Circuits: The Systems on Chip Approach* (IET Press, 2008). His research interests include wireless and mobile communications, RF and analogue circuits, microelectronic devices and systems, machine learning, and deep learning.

Dr. Sun was a Series Editor of IEE Circuits, Devices and Systems Book Series from 2003 to 2008. He had been Associate Editor of the IEEE Transactions on Circuits and Systems—I: Regular Papers from 2010 to 2011, from 2016 to 2017, and from 2018 to 2019. He is also Editor of *ETRI Journal*, the *Journal of Semiconductors*, and the *Journal of Sensor and Actuator Networks*. He was a Guest Editor of eight IEEE and IEE/IET journal special issues: High-frequency Integrated Analogue Filters in IEE Proceedings on Circuits, Devices and Systems in 2000, RF Circuits and Systems for Wireless Communications in IEE Proceeding on Circuits, Devices and Systems in 2002, Analogue and Mixed-Signal Test for Systems on Chip in IEE Proceeding on Circuits, Devices and Systems in 2004, MIMO Wireless and Mobile Communications in IEE Proceeding on Communications in 2006, Advanced Signal Processing for Wireless and Mobile Communications in IET Signal Processing in 2009, Cooperative Wireless and Mobile Communications in IET Communications in 2013, Software-Defined Radio Transceivers and Circuits for 5G Wireless Communications in the IEEE Transactions on Circuits and Systems—II: Express Briefs in 2016, and the 2016 IEEE International Symposium on Circuits and Systems in the IEEE Transactions on Circuits and Systems—I: Regular Papers in 2016. He has also been widely involved in various IEEE technical committee and international conference activities.

MU PENG is currently pursuing the master's degree with the College of Computer Science and Electronic Engineering, Hunan University.

His research interests include data mining and intelligent transportation technology.

NIANFEI GAN received the Ph.D. degree in mechanical engineering from Hunan University, China, in 2007.

She is currently an Associate Professor with the College of Mechanical and Vehicle Engineering, Hunan University. She has published more than ten articles. Her research interests include driving intention recognition, human-like motion planning, and its application to intelligent vehicle.