Adaptive Spatial-Temporal Graph Convolution Networks for Collaborative Local-Global Learning in Traffic Prediction

Yibi Chen, Yunchuan Qin, Kenli Li, *Senior Member, IEEE*, Chai Kiat Yeo, and Keqin Li, *Fellow, IEEE*

***Abstract*—The rapid growth of vehicles as countries become more developed has brought great challenges to traffic prediction. Recent works model only local or global spatial-temporal features via graph neural networks (GNNs). Furthermore, the explicit graph structure information may contain bias, in particular, the lack of connections among multiple nodes when in fact, they are interdependent. This results in the inability to accommodate information interaction and the underutilization of high-quality information. In this article, we design an adaptive spatial-temporal graph convolution networks (ASTGCNs) to collaboratively learn local-global spatial-temporal information for traffic prediction. Specifically, we obtain different local spatial-temporal information (i.e. spatial-temporal information of each temporal point) by dividing the global spatial-temporal information along the temporal dimension. For local spatial-temporal information, we establish an adaptive graph convolution to enhance the ability of graph convolution networks (GCNs) in managing bias in the explicit graph structure. We then employ an attention mechanism to learn the local summarization of dynamic node neighborhoods to obtain high-quality information. For global spatial-temporal information, a temporal convolution network (TCN) block and the ordinary differential equation (ODE) are utilized in our model. In essence, our proposed ASTGCNs integrates adaptive graph convolution, attention mechanism, TCN block and ODE to collaboratively learn local-global spatial-temporal information. Experimental results show that our ASTGCNs is superior to state-of-art (SOTA) methods when applied to four real-world datasets.**

***Index Terms*—Adaptive graph convolution, attention mechanism, local-global spatial-temporal information, ordinary differential equation, temporal convolution network, traffic prediction.**

Manuscript received 10 May 2022; revised 25 July 2022 and 9 May 2023; accepted 12 May 2023. Date of publication 16 May 2023; date of current version 17 October 2023. The work was supported in part by the Key Program of the National Natural Science Foundation of China under Grants 61133005 and 61432005, in part by China Scholarship Council under Grant 202106130053, and in part by the Postgraduate Scientific Research Innovation Project of Hunan Province under Grant CX20220413. The review of this article was coordinated by Prof. Zhanyu Ma. *(Corresponding author: Kenli Li.)*

Yibi Chen is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China, and also with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: chenyibi@hnu.edu.cn).

Yunchuan Qin and Kenli Li are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: qinyunchuan@hnu.edu.cn; lkl@hnu.edu.cn).

Keqin Li is with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China, and also with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA (e-mail: lik@newpaltz.edu).

Chai Kiat Yeo is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: asckyeo@ntu.edu.sg).

Digital Object Identifier 10.1109/TVT.2023.3276752

I. Introduction

A. Motivation

INTELLIGENT transportation system (ITS) is one important barometer in evaluating the modernization and smart level of a city or country, especially for a region with a huge population and high traffic flow. In the context of building smart cities, traffic prediction [1], [2] can effectively promote the sustainable development of ITS, including safe travel [3], smooth road traffic [4], etc. Despite the great progress made in the traffic prediction methods applied to ITS, traffic forecasting still faces tremendous challenges [5], [6]. According to a report by Beijing Traffic Management Bureau, the number of motor vehicles in Beijing rose 279000 in 2021 compared to 2020, and the peak road congestion index reached 2.048 (i.e the highest congested city in China). Therefore, designing an efficient traffic prediction method is very necessary to reduce the difficulty of traffic management, especially for ITS in smart cities.

With the development of convolutional neural networks (CNNs), many methods have been applied to traffic prediction and have achieved impressive results in processing traffic image [7] and video data [8]. For example, ST-ResNet [9] proposes 2D convolutional residual network to capture spatial features between regions. STDN [10] adopts 2D CNN and long short-term memory (LSTM) to learn spatial and temporal features, respectively. LMST3D-ResNet [11] proposes 3D CNN and resnet to full exploit spatial-temporal features of multiple local regions. STAM [12] constructs 3D CNN layers to learn dynamic spatial-temporal information for videos. However, all these methods focus on the modelling of Euclidean data.

With the rise of graph representation learning [13], [14], many researchers focus on the graph traffic data (i.e non-Euclidean data), and these data are usually irregular [15]. Face with complex and changing traffic situation, graph traffic data can be used to freely construct different nodes (i.e traffic observation station) and establish relationships with other nodes. Generally, every node has different characteristics such as core and non-core regions. The smaller the spatial distance among nodes, the greater is the impact on their relationship and vice versa. According to the above analysis, how to effective build a model of graph traffic

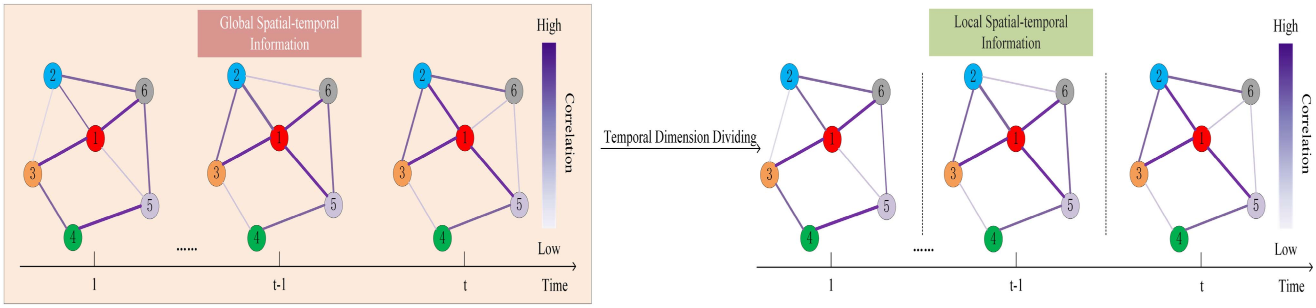


Fig. 1. Obtaining local spatial-temporal information.

prediction is a daunting challenge, especially for dynamic and real-time traffic environment.

Several existing traffic prediction methods have been applied to local spatial-temporal information. Li et.al [16] designed a unified neural network for local traffic subgraph in data sequences. Fang et.al [17] proposed to expand the receptive region of nodes by DAGC, and MS-Net integrates long-range traffic information and considers multiple-modal external information. FedTe [18] establishes an optimal regional traffic matrix and hierarchical GNNs to handle traffic information of multiple local regions and various traffic variation, respectively. STSGCN [19] constructs multiple spatial-temporal synchronous modules to obtain the heterogeneities in the localized features. However, these methods fail to consider the global spatial-temporal information.

Some global spatial-temporal methods have been put forward to achieve traffic prediction. STGCN [20] proposes complete graph convolutional structures to solve mid-and-long term traffic prediction. DCRNN [21] designs the bidirectional random walks and the encoder-decoder module based on RNN to capture spatial and temporal correlation information, respectively. GraphWaveNet [22] utilizes adaptive dependency matrix and deep 1D convolution layer to learn spatial-temporal features. STGODE [23] adopts two TCNs and an ODE to synchronously extract spatial-temporal features. However, the above-mentioned local and global traffic prediction methods fail to jointly consider the local-global spatial-temporal information. Moreover, the explicit graph structure may contain bias and some high-quality information is unutilized.

B. Our Contributions

To tackle these problems, we construct a method called ASTGCNs that collaboratively learns the local-global spatial-temporal information for traffic prediction. Specifically, we obtain different local spatial-temporal information (i.e. spatial-temporal information of each temporal point) by dividing the global spatial-temporal information along the temporal dimension as shown in Fig. 1. The significance of local spatial-temporal information is to know the state of all nodes at each temporal dimension in detail thereby deeply exploiting the effective information. For local spatial-temporal information, adaptive graph convolution improves GCNs' ability to tackle bias in the explicit graph structure whereby the adaptive node parameter learns specific parameters for each node, and the adaptive graph generation automatically explores the potential dependencies among nodes. Then, we consider the spatial adjacency matrix in the attention mechanism to learn the local summarization of dynamic node neighborhoods for high-quality information. For global spatial-temporal information, we utilize a TCN block to obtain the long-term temporal correlation and an ODE to establish deeper networks. In essence, our constructed ASTGCNs integrates adaptive graph convolution, attention mechanism, TCN block and ODE to collaboratively learn the local-global spatial-temporal information. We applied our method on four real-world traffic graph datasets and the performance of ASTGCNs is superior to the SOTA methods.

We summarize the main contributions as follows.

- We propose a traffic prediction method called ASTGCNs. This method integrates adaptive graph convolution, attention mechanism, TCN block and ODE to collaboratively learn the local-global spatial-temporal information.
- Our approach is an end-to-end structure, in which adaptive graph convolution enhances GCNs ability to address bias in explicit graph structure.
- Our proposed ASTGCNs establishes an attention mechanism which can obtain high-quality information by computing the local summarization of node neighborhoods.
- Experimental results demonstrate that our method outperforms SOTA benchmark methods on four real-world traffic datasets.

C. Organization

The rest of this article is organized as follows. In Section II, we review the various technologies that have been applied to traffic prediction. Section III presents the definitions used and the problem statement to be accomplished. In Section IV, we propose the ASTGCNs framework and describe each module in detail. Section V documents the series of experiments performed as well as the results. Section VI is our conclusion of this work.

II. Related Work

With the building of smart cities and ITS, many deep learning methods have deployed in traffic forecasting. We review these methods under three categories: CNNs, GCNs and attention mechanism.

A. Convolutional Neural Networks

In recent years, CNNs have achieved remarkable results in process traffic image and video data. STDN [10] adopts CNNs and LSTM to capture the spatial-temporal information for multiple local regions. Ma et.al [24] propose CNNs and gated recurrent unit (GRU) to select hybrid spatial-temporal features for short-term traffic prediction. Jia et.al [25] design a multi-view CNN aimed at modelling adaptive time-varying control to compute the traffic flow speed. ST-ResNet [9] proposes a convolutional residual network to learn the spatial features between regions. LMST3D-ResNet [11] proposes 3D convolutional residual networks to fully exploit the spatial-temporal features of multiple local regions. Fu et.al [26] establish a Faster R-CNN-based model to learn the video data in traffic sign detection. Perafan-Villota et.al [27] systematically integrate CNNs, Hadoop and Spark frameworks to handle large-scale traffic videos. Sindhu et.al [28] devise a YOLOv4 based model to detect vehicles in traffic videos under different environmental conditions. STAM [12] constructs 3D CNNs layers to learn the dynamic spatial-temporal information for videos.

B. Graph Convolutional Networks

Recently, the emergence of GCNs in exploring graph spatial-temporal information for traffic prediction has achieved good results. STGCN [20] proposes complete graph convolutional structures to solve mid-and-long term traffic prediction. DeepSTN+ [29] aims to obtain long-range spatial correlation information via ConvPlus component and PoI prior knowledge. DCRNN [21] designs the bidirectional random walks and the encoder-decoder module based on RNN to capture spatial and temporal correlation information, respectively. LSGCN [30] aims to capture long short-term spatial-temporal features via spatial gated block and GCNs. GSTPRN [31] builds position graph convolution, approximate personalized propagation to enhance spatial position and neighborhood information. GraphWaveNet [22] utilizes adaptive dependency matrix and deep 1D convolution layer to learn spatial-temporal features. AGCRN [32] integrates GRU, node adaptive parameter learning and data adaptive graph generation to avoid pre-defined graph. HGCN [33] uses multiple GCNs with spectral pooling in a hierarchical manner for spatial-temporal learning. STGODE [23] adopts an ODE and two TCN blocks with residual structure to synchronously extract spatial-temporal features.

C. Attention Mechanism

The attention mechanism sets different values for different information to obtain high-quality information. ST-RGAN [34] designs graph attention networks and residual structure for fine-grained traffic forecasting. AST-GAT [35] establishes multi-head graph attention block and LSTM to obtain the spatial-temporal features and temporal features, respectively. DSANet [36] employs local-global temporal convolution and self-attention for time series prediction. IGAGCN [37] adopts causal convolution and attention mechanism for short time spans and to obtain dynamic information. AGNN [38] calculates an adaptive local summarization of node neighborhoods to obtain the attention in the homogeneous graph. ST-LBAGAN [39] proposes U-Net structure and attention map to handle missing traffic data imputation. TAGCN [40] employs temporal attention mechanism for different time granularity (e.g. hour, day and week-level). ASTGCN [41] integrates attention mechanism and GCNs for different time periods in traffic forecasting.

Overall, our proposed ASTGCNs is quite different from the existing literatures. We establish local spatial-temporal information to deeply explore the effective information by an adaptive graph convolution and an attention mechanism. Moreover, we adopt a temporal convolution network (TCN) block and the ordinary differential equation (ODE) to obtain global spatial-temporal information. The above-mentioned modules are integrated for collaborative local-global learning.

III. Preliminary

We describe some basic definitions and the problem statement for the graph spatial-temporal traffic network.

Definition 1 (Graph traffic network): A graph traffic network is described as $G = (V, E, A)$, where V presents the set of nodes; E is the set of edges; A is the adjacency matrix. Note that we also adopt spatial adjacency matrix A^{sp} [20] in this article.

Definition 2 (Graph traffic signal matrix): The historical data of each node at same temporal point t is denoted as $X_t = (X_{1,t}, \ldots, X_{n-1,t}, X_{n,t}) \in \mathbb{R}^{N\times F}$. For all nodes at various temporal points, the graph traffic signal matrix $X \in \mathbb{R}^{T\times N\times F}$ is represented as:

$$X = [X_1, \ldots, X_{t-1}, X_t]^T. \tag{1}$$

where T denotes whole time; N is the number of nodes; F is the dimension of each node.

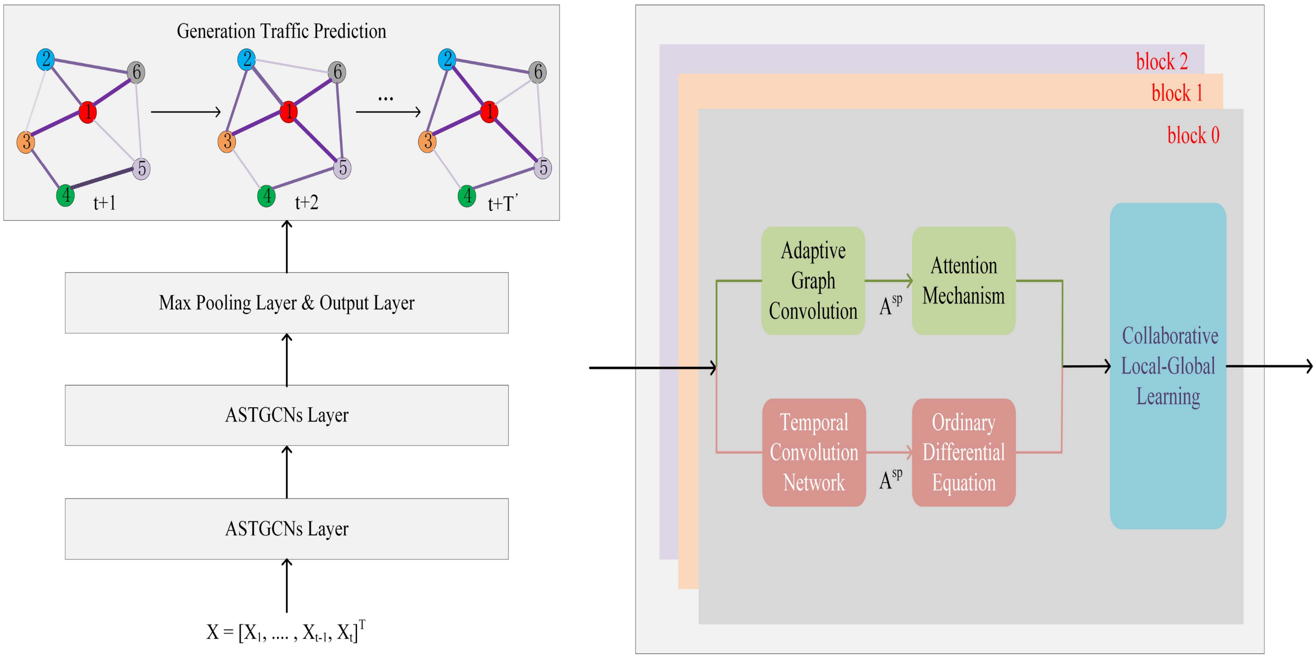
Problem statement: Given the graph traffic signal matrix X based on a graph traffic network G, the purpose of the traffic prediction is to learn a function f_θ based on historical observation data to predict the future T' situation and it is shown as follows:

$$\{X, G\} \xrightarrow{f_\theta} \{X_{t+1}, \ldots, X_{t+T'}\}. \tag{2}$$

IV. The Proposed ASTGCNs Architecture

STGODE [23] constructs two TCN blocks with residual structure and an ODE to synchronously learn global spatial-temporal features. This motivates us to introduce different local spatial-temporal information for collaborative local-global learning. The significance of the local spatial-temporal information is to understand the state of nodes at different temporal dimensions in detail so as to deeply exploit the effective information, especially for dynamic and real-time traffic environment. Our proposed ASTGCNs and STGODE have several notable differences:

1) Our method jointly considers local and global spatial-temporal information for collaborative local-global learning.
2) For local spatial-temporal information, we establish an adaptive graph convolution and an attention mechanism.



(a) (b)

Fig. 2. (a) Presents ASTGCNs architecture. (b) Shows the detail of ASTGCNs layer for collaborative local-global learning.

3) For global spatial-temporal information, we utilize only one TCN block without residual structure and an ODE of spatial adjacency matrix A^{sp}.

Fig. 2(a) presents our ASTGCNs architecture, including two ASTGCNs layers consisting of several ASTGCNs blocks, a max pooling layer and an output layer. In Fig. 2(b), the input spatial-temporal information (i.e. black line) is separated into two branches (i.e. the green line and the red line). The first branch (i.e. green line) shows the processing of the different local spatial-temporal information, $X_{:t} \in \mathbb{R}^{N\times F}$ (i.e. the spatial-temporal information of each temporal point) obtained by dividing the global spatial-temporal information along the temporal dimension. The second branch (i.e. red line) shows the processing of the global spatial-temporal information which is consistent with the input spatial-temporal information. For local spatial-temporal information (i.e. green line), we employ an adaptive graph convolution and an attention mechanism (i.e. green cuboid). For global spatial-temporal information (i.e. red line), we utilize a TCN block and an ODE (i.e. red cuboid). Finally, we achieve collaborative local-global learning by the aggregation of the outputs of the two branches.

A. Local Spatial-Temporal Information

Fig. 3 shows the process of local spatial-temporal information learning. Specifically, we input different local spatial-temporal information into an adaptive graph learning module where the outputs are then passed into an attention mechanism module. Thereafter, these local spatial-temporal information are aggregated.

1) Adaptive Graph Convolution: Adaptive graph convolution module includes the adaptive node parameter and adaptive graph generation. Its role is to enhance the ability of the GCNs to manage bias in the explicit graph structure.

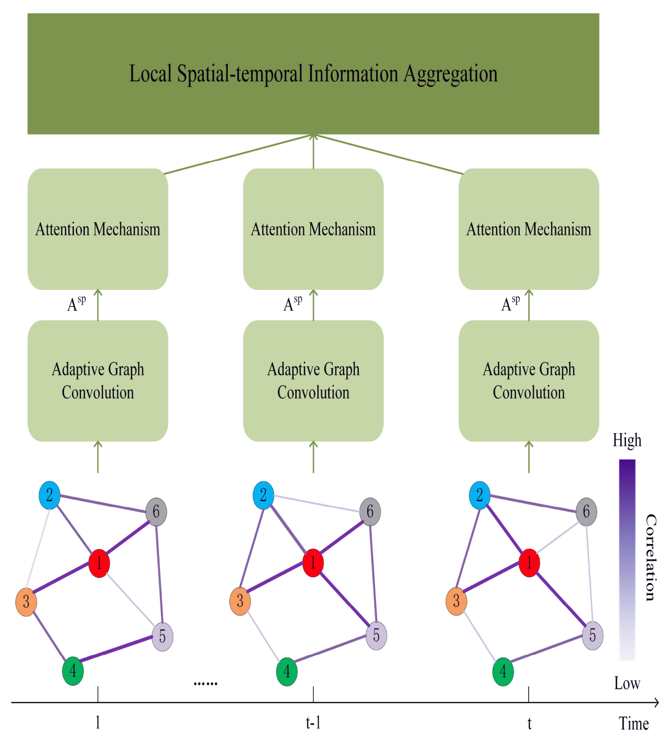


Fig. 3. Process of local spatial-temporal information learning.

The GCNs of the 1^{st} order Chebyshev adopts the parameters sharing mode to decrease the number of parameters for node embeddings as shown below:

$$Q_{:t} = \widehat{A}X_{:t}\Phi + b,$$
$$\widehat{A} = I_n + \overline{D}^{-\frac{1}{2}}\overline{AD}^{-\frac{1}{2}}, \tag{3}$$

where I_n indicates a unit matrix, $\overline{D}$ and $\overline{A}$ are the degree matrix and the adjacent matrix respectively. $X_{:t} \in \mathbb{R}^{N\times F}$ is the input and $Q_{:t} \in \mathbb{R}^{N\times d}$ is the output, $\Phi \in \mathbb{R}^{F\times d}$ denotes the weights and $b \in \mathbb{R}^d$ represents the bias.

As mentioned, the GCNs utilizes parameter sharing among all the nodes to decrease the number of parameters. However, the shared parameter mode leads to the deviation of some important nodes information. Assigning parameters to each node $\Phi \in \mathbb{R}^{N\times F\times d}$ is an efficient method, but Φ is too large to may cause overfitting problem, especially if N is big enough.

To tackle the above problem, adaptive node parameters adopts the mode of matrix factorization to enhance GCNs ability. Specifically, the weights $\Phi \in \mathbb{R}^{N\times F\times d}$ are replaced by $\Phi = E_G \cdot W_G$, where $E_G \in \mathbb{R}^{N\times C}$ is the embedding matrix and $W_G \in \mathbb{R}^{C\times F\times d}$ represents a weight pool, and $C \ll N$. Taking a node i as an example, E_G^i extracts parameters Φ^i from the shared weight pool W_G, and this process is seen as finding the specific parameters for node i in the set of candidate parameters. Likewise for bias b. The equation for adaptive node parameter is:

$$Q_{:t} = \widehat{A}X_{:t}E_G W_G + E_G b_G. \tag{4}$$

In addition, from the perspective of distance factor, the spatial distance on the graph are defined by computing the physical distance between the nodes. However, some nodes establish the interaction by utilizing an intermediate node and the above process fails to consider some potential dependencies by the intermediate node. This causes bias in the explicit graph structure.

Adaptive graph generation aims to automatically explore potential dependencies in the data to solve bias in the explicit graph structure. The node embedding $E_A \in \mathbb{R}^{N\times d_e}$ is obtained by randomly initializing all nodes, where each row of E_A is one node embedding and d_e indicates the node embedding dimension. Then, we compute the multiplication of E_A and E_A^T to infer the potential dependencies between each pair of nodes as follows:

$$\widetilde{A} = \overline{D}^{-\frac{1}{2}}\overline{AD}^{-\frac{1}{2}} = \sigma(ReLU(E_A \cdot E_A^T)), \tag{5}$$

where σ is softmax function.

From (5), the matrix $\overline{D}^{-\frac{1}{2}}\overline{AD}^{-\frac{1}{2}}$ is replaced to reduce the repeated calculations during the training process. E_A can automatically explore the potential dependencies among nodes during training; thus the equation for adaptive graph generation is as follows:

$$Q_{:t} = (I_n + \widetilde{A})X_{:t}\Phi. \tag{6}$$

We then obtain the adaptive graph convolution which combines adaptive node parameter (i.e. (4)) and adaptive graph generation (i.e. (6)) and it is shown as follows:

$$Q_{:t} = (I_n + \widetilde{A})X_{:t}E_G W_G + E_G b_G. \tag{7}$$

2) Attention Mechanism: In graph traffic data, each node establishes various dependency relationships with its neighbors. However, these dependency relationships bring different levels of importance of information due to various factors (e,g. spatial distance, different time, core or non-core regions). How to find high-quality information from a large of dependency relationship is a daunting challenge, especially for real-time changes in the local spatial-temporal environment. Therefore, we design an attention mechanism module to learn the local summarization of dynamic node neighborhoods for high-quality information.

Based on the outcome of adaptive graph learning, attention propagation scheme is expressed as:

$$K_{:t} = PQ_{:t}, \tag{8}$$

where $P \in \mathbb{R}^{N\times N}$ denotes the propagation function.

For P in the propagation scheme, the attention between the node i and the node j is shown as:

$$P_{ij} = (1/C)e^{\beta \cos(Q_{:t,i}, Q_{:t,j})}, \tag{9}$$

where $C = \sum_{j\in N(i)\cup i} e^{\beta \cos(Q_{:t,i}, Q_{:t,j})}$ calculates the relation degree between nodes i and j, β is a trainable parameter, and $\cos(x, y) = x^T/y||x||\,||y||$ with the L_2 norm $||x||$.

We replace the static adjacent matrix $\overline{A}$ with the spatial adjacent matrix A^{sp}, and the output is shown as:

$$Z_{:t} = f(Q_{:t}, A^{sp}) = softmax(K_{:t}, W), \tag{10}$$

where $Z_{:t} \in \mathbb{R}^{N\times d}$ and $W \in \mathbb{R}^{d\times d}$ is the weights.

Finally, we aggregate different local spatial-temporal information $Z_{:t}$ to produce $Z_{local} \in \mathbb{R}^{T\times N\times d}$.

B. Global Spatial-Temporal Information

Fig. 4 shows the process of global spatial-temporal information learning. Specifically, we input the global spatial-temporal information into the TCN block module, and then go through an ODE module, and finally learn the global spatial-temporal information.

1) Temporal Convolution Network: Temporal convolution network (TCN) is a kind of one-dimensional dilated convolutional network to learn long-term temporal dependencies. TCN block contains three hidden layers.

$$H_{tcn}^{(m)} = \begin{cases} X, & \text{if } m = 0 \\ \mu(W^{(m)} *_b^{(m)} H_{tcn}^{(m-1)}), & \text{if } m = 1, 2, 3 \end{cases} \tag{11}$$

where $X \in \mathbb{R}^{T\times N\times F}$ denotes the initial input and $H_{tcn}^{(m)} \in \mathbb{R}^{T\times N\times d}$ is the m-th layer output, μ is an activate function (i.e. ReLU), $W^{(m)}$ is the m-th convolution kernel, $b^{(m)} = 2^{(m-1)}$ is the dilated rate.

2) Ordinary Differential Equation: Ordinary differential equation (ODE) addresses the over-smoothing problem with increasing CGNN [42] network depth. For spatial-temporal information, we illustrate the discrete expression of the ODE mathematical definition as:

$$H^{(l+1)} = H_{ode}^{(l)} \times_1 \frac{\alpha}{2}\widehat{A} \times_2 U \times_3 P + H^{(0)}, \tag{12}$$

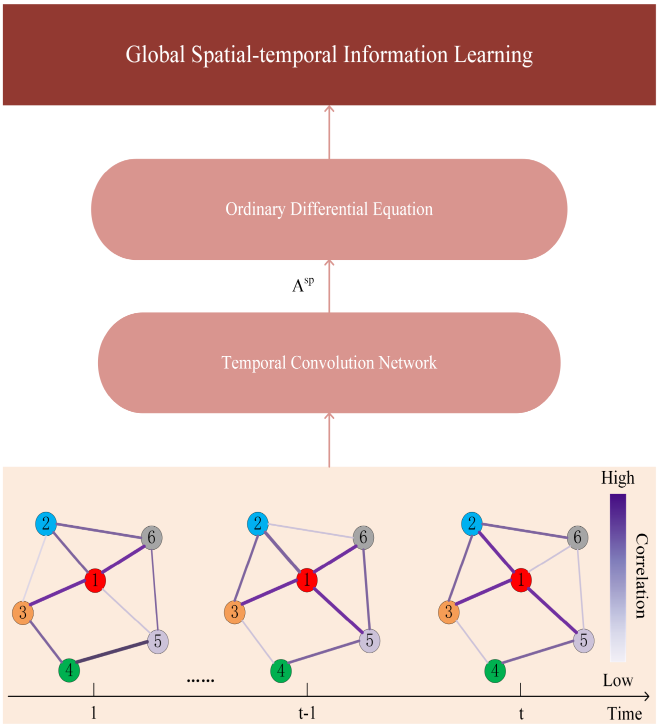


Fig. 4. Process of global spatial-temporal information learning.

where $H^{(l)} \in \mathbb{R}^{T\times N\times F}$ is the hidden information of l-th layer, $\alpha \in [0,1]$ represents a hyperparameter, $\times_i$ is matrix multiplication on mode i, U represents the transform matrix for temporal dimension, P denotes the characteristics transform matrix, and $H^{(0)}$ is the original input.

The scheme extracts the spatial information and temporal information simultaneously based on the input of spatial-temporal information (i.e. tensor representation). According to different $\times_i$, the complex spatial-temporal dependencies is coupled. To address the problem of over-smoothing, $H^{(0)}$ is introduced. (12) is further expanded to:

$$H^{(l)} = \sum_{i=0}^{l} \left(H^{(0)} \times_1 \frac{\alpha}{2} \widehat{A}^i \times_2 U^i \times_3 P^i \right), \tag{13}$$

where $H^{(l)}$ aggregates the information of all layers as the output.

For the detailed explanation of the corollary process of ODE, please refer to STGODE [23]. The ODE is represented as follows:

$$H(t) = ODE \left(\frac{dH(t)}{dt}, H^{(0)}, t \right), \tag{14}$$

where $\frac{dH(t)}{dt} = H(t) \times_1 (\frac{\alpha}{2} \widehat{A} - I_n) + H(t) \times_2 (U - I_n) + H(t) \times_3 (W - I_n) + H^{(0)}$, and t denotes a continuous variable parameter.

To clearly illustrate the process of global spatial-temporal information, we jointly describe a TCN block and an ODE.

$$H(t)_{global} = ODE \left(\frac{dH(t)}{dt}, H_{tcn}^{(m)}, t \right), \tag{15}$$

where $H(t)_{global} \in \mathbb{R}^{T\times N\times d}$ is the output, and the static adjacent matrix $\overline{A}$ is replaced with spatial adjacent matrix A^{sp}, $\frac{dH(t)}{dt} = H(t) \times_1 (\frac{\alpha}{2}(I_n + \overline{D}^{-\frac{1}{2}} A^{sp} \overline{D}^{-\frac{1}{2}}) - I_n) + H(t) \times_2 (U - I_n) + H(t) \times_3 (W - I_n) + H^{(0)}$.

C. Collaborative Local-Global Learning

The first branch (i.e. different local spatial-temporal information) is processed by an adaptive graph learning and an attention mechanism, and then we aggregate these information to capture $Z_{local} \in \mathbb{R}^{T\times N\times d}$. The second branch (i.e. global spatial-temporal information) is learned through a TCN block and an ODE, and we obtain $H(t)_{global} \in \mathbb{R}^{T\times N\times d}$. Therefore, the collaborative local-global learning is expressed as:

$$S = Z_{local} + H(t)_{global}. \tag{16}$$

For multiple ASTGCNs layers, the above-mentioned learning process of local-global spatial-temporal information is repeated, in which $S^{(l)}$ is the output of l-th layer and the input of $(l+1)$-th layer. Moreover, several ASTGCNs blocks are adopted in a parallel structure to explore the complex spatial-temporal information and different dependencies.

D. Pooling Layer and Output Layer

Following the ASTGCNs layers, a max-pooling layer can choose different ASTGCNs blocks to aggregate information, and then the output layer employs two MLP to generate the prediction.

The parameters of the model are optimized by the Adam optimizer. Huber loss function is used to our proposed ASTGCNs:

$$Loss(Y, \widetilde{Y}) = \begin{cases} \frac{1}{2}(Y - \widetilde{Y})^2, & \text{if } |Y - \widetilde{Y}| \leq \delta; \\ \delta|Y - \widetilde{Y}| - \frac{1}{2}\delta^2, & \text{if } |Y - \widetilde{Y}| > \delta, \end{cases} \tag{17}$$

where δ denotes a hyperparameter to adjust the sensitivity from squared error loss.

V. Experiments

Using four real-world traffic flow datasets, we design different experiments to showcase the performance of our proposed ASTGCNs.

A. Datasets

Caltrans Performance Measurement System (PEMS) has more than 39,000 sensors on California highways to construct real-time traffic datasets such as PeMS03, PeMS04, PeMS07, PeMS08. These real-time highway data are captured every half-minute and comprise average occupancy, average speed, average and total flow. Flow refers to the number of vehicles that pass over the detector in a 30-second period and occupancy is the fraction of time that a vehicle is over the detector. The data thus summarize average vehicle flow as a function of occupancy providing insight into how traffic will flow in a given stretch of roads.

TABLE I
BASELINE RESULT COMPARISON ON TRAFFIC PREDICTION

Baseline	PeMS03			PeMS04			PeMS07			PeMS08		
	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE
ARIMA[43]	47.16	35.29	33.83	48.55	33.63	24.15	59.31	38.32	19.50	44.01	30.98	22.64
STGCN[20]	30.76	17.71	17.59	36.28	22.87	14.59	39.83	25.57	11.49	28.21	18.42	11.61
DCRNN[21]	30.46	18.25	18.71	33.98	21.57	15.14	38.97	25.51	12.12	26.91	16.98	10.98
GraphWaveNet[22]	33.45	19.62	19.31	40.17	25.20	17.57	41.06	26.78	12.29	30.43	18.58	12.52
ASTGCN[41]	29.42	17.19	17.02	34.09	21.91	15.36	38.56	24.28	10.86	27.24	17.55	12.01
STSGCN[19]	29.34	17.08	16.95	33.72	21.43	14.83	39.21	24.56	10.58	27.03	17.31	11.51
STODE[23]	28.02	16.67	16.83	32.98	21.17	14.74	38.45	23.39	10.43	25.94	16.81	10.66
ASTGCNs	**27.25**	**16.03**	**16.24**	**31.60**	**20.14**	**13.87**	**34.52**	**21.92**	**9.82**	**24.90**	**15.99**	**10.21**

PeMS03: The dataset collects data from 358 sensors with 547 edges and 26208 time steps. The collection time span is from 9/2018 to 11/2018.

PeMS04: The dataset collects data from 307 sensors with 340 edges and 16992 time steps. The collection time span is from 1/2018 to 2/2018.

PeMS07: The dataset collects data from 883 sensors with 866 edges and 28224 time steps. The collection time span is from 5/2012 to 6/2012.

PeMS08: The dataset collects data from 170 sensors with 295 edges and 17856 time steps. The collection time span is from 7/2016 to 8/2016.

B. Settings

We adopt a ratio of 6:2:2 to split the dataset into a training set, validation set and test set. The hyper-parameters σ and ϵ of the spatial adjacency matrix are 10 and 0.5. The learning rate is 0.01 and the regularized factor set to 0.8. The node embedding dimension (i.e. d_e) is set to 5. The trainable parameter β is set to 1. The hidden units are 64, 32, 64 in the TCN block, and each layer consists of 3 ASTGCNs blocks. 200 epochs are used in the training. Three classical evaluation metrics are used to evaluate the performance of the experiments, namely, Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE). We employ 60 minutes of historical traffic data to predict the next 60 minutes of future traffic.

C. Comparision of the Baseline Methods

We select both classical methods as well as state-of-the-art (SOTA) methods in graph traffic prediction as baselines for comparison with our proposed ASTGCNs so as to showcase the effectiveness of our method.

ARIMA [43]: This statistical method adopts residual autocorrelations for time series.

STGCN [20]: This method constructs graph convolutional structures to solve mid-and-long term traffic prediction.

DCRNN [21]: This method designs the bidirectional random walks and the encoder-decoder module based on RNN to capture spatial and temporal correlation information, respectively.

GraphWaveNet [22]: This method utilizes adaptive dependency matrix and deep 1D convolution layer to learn spatial-temporal features.

ASTGCN [41]: This method integrates attention mechanism and GCNs for different time periods in traffic forecasting.

STSGCN [19]: This method employs multiple spatial-temporal synchronous modules to obtain the heterogeneities in the localized features.

STGODE [23]: This method adopts an ODE and two TCN blocks with residual structure to synchronously extract spatial-temporal features.

D. Result Comparison and Analysis

In Table I, we comprehensively compare the results of multiple baseline methods. It can be clearly seen that our ASTGCNs outperform the SOTA benchmarks in all four traffic datasets.

We conduct an in-depth analysis of the performance of the various baseline methods. ARIMA fails to consider spatial-dimension information. STGCN constructs a simple graph convolutional structure to capture the shallow information. DCRNN designs the RNN structure to limit its mid to long term prediction performance. GraphWaveNet loses a lot of graph structure information because GNNs is not considered. ASTGCN only adopts an attention mechanism to learn temporal dimension features. STSGCN does not consider global spatial-temporal features.

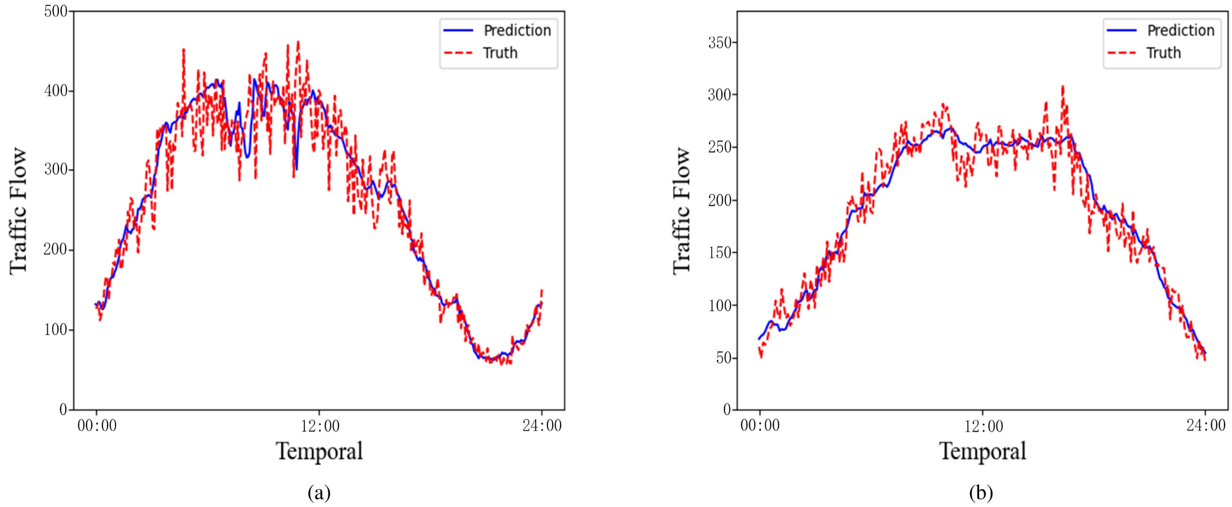


Fig. 5. Comparison of truth and prediction on ASTGCNs. (a) Node 80. (b) Node 256.

STGODE fails to model the different local spatial-temporal information.

Our ASTGCNs collaboratively learns local-global spatial-temporal information to perform traffic prediction. For local spatial-temporal information, we establish an adaptive graph convolution to enhance the ability of graph convolution networks (GCNs) to tackle bias in the explicit graph structure. Thereafter, we employ an attention mechanism to learn the local summarization of dynamic node neighborhoods to obtain high-quality information. For global spatial-temporal information, a temporal convolution network (TCN) block and the ordinary differential equation (ODE) are utilized in our model. Finally, our method deeply explores the spatial-temporal information fully by aggregating the two types of information. Therefore, ASTGCNs achieves better result.

E. Analysis of Truth and Prediction on ASTGCNs

In Fig. 5, we compare truth and prediction on ASTGCNs based on the relationship of traffic flow and temporal. We observe that the prediction results of ASTGCNs are close to the truth and reflect the trend of traffic flow changes along the temporal dimension.

From Fig. 5(a), we find that the high traffic flow of node 80 appeared before 12:00. Compared with the normal traffic flow period, the change between prediction and truth is obvious in the high traffic flow period. Fig. 5(b) shows traffic situation of node 256 and the high traffic flow occurs around 12:00. The change between prediction and truth is smoother than node 80 because the range of the traffic flow from 40 to 300.

F. Performance of the Proposed ASTGCNs With Different Node Embedding Dimension

We evaluate the performance of ASTGCNs on the PeMS07 dataset using different node embedding dimension d_e on the adaptive graph learning module. Table II shows the performance under the three metrics: RMSE, MAE and MAPE. We observe that $d_e = 5$ performs better than $d_e = 2$ in RMSE and MAE, but $d_e = 2$ performs better than $d_e = 5$ in MAPE. Compared to $d_e = 2$ and $d_e = 5$, $d_e = 10$ yields poor performance.

TABLE II
RESULT OF DIFFERENT NODE EMBEDDING DIMENSION ON PEMS07 DATASET

Dimension	PeMS07		
	RMSE	MAE	MAPE
d_e=2	34.70	21.98	9.75
d_e=5	34.52	21.92	9.82
d_e=10	35.28	22.59	10.09

TABLE III
RESULT OF DIFFERENT NODE EMBEDDING DIMENSION ON PEMS08 DATASET

Dimension	PeMS08		
	RMSE	MAE	MAPE
d_e=2	25.29	16.28	10.37
d_e=5	24.90	15.99	10.21
d_e=10	25.19	16.21	10.30

In additional, we also utilize different d_e to evaluate the performance of ASTGCNs on the PeMS08 dataset. Table III show that $d_e = 5$ obtains best result, and $d_e = 10$ performs slightly good than $d_e = 2$.

The results of different d_e on adaptive graph learning module show that our ASTGCNs obtains better result compared to the SOTA baselines.

G. Ablation Experiment

To clearly show the performance of the different modules, we conduct ablation studies of the different modules on PeMS07 dataset. The results are as follows:

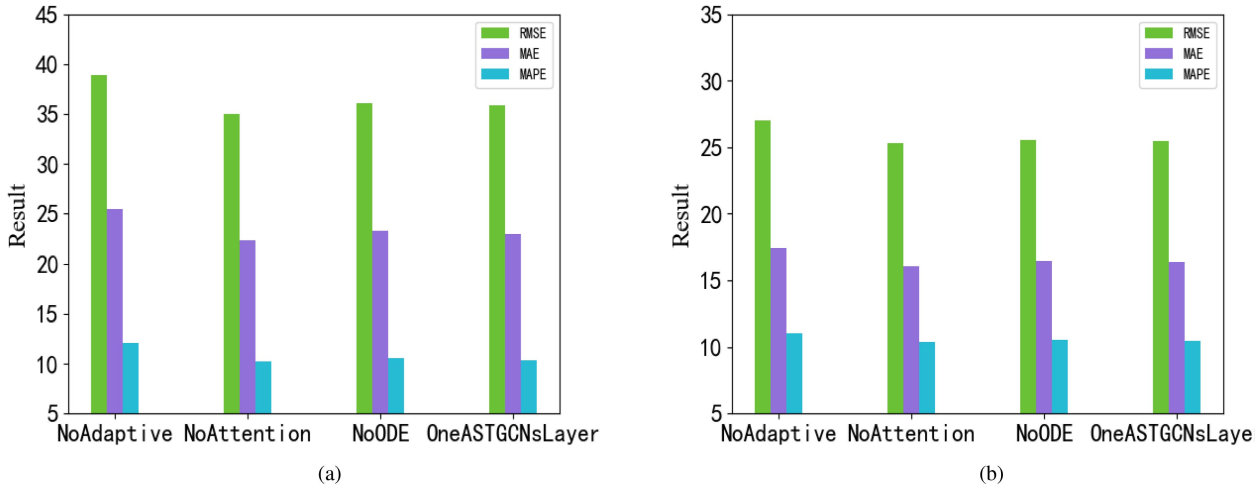


Fig. 6. Results from the different modules. (a) PeMS07. (b) PeMS08.

- *NoAdaptive:* the method fails to consider adaptive graph learning to illustrate the effect of bias in the explicit graph structure.
- *NoAttention:* the method removes the attention mechanism to show the different degree of influence of nodes relationship.
- *NoODE:* the method does not construct the ODE to show the benefits of a deeper network.
- OneASTGCNsLayer: the method only ulitizes one ASTGCNs layer to compare single layer and multiple layers learning.

It can be seen from Fig. 6(a) that all results (i.e. NoAdaptive, NoAttention, NoODE and OneASTGCNsLayer) on PeMS07 dataset are poorer than ASTGCNs. NoAdaptive yields the worst performance which illustrates that the bias in the explicit graph structure is more sensitive to the model, especially for dynamic and real-time traffic environment. The performance of NoAttention is improved compared to NoODE and OneASTGCNsLayer, which reflects that deep network and multiple layers can extract more spatial-temporal information. The result of NoODE and OneASTGCNsLayer shows that these two modules have a similar effect on the model.

Fig. 6(b) shows the results on PeMS08 dataset. By observing the performance of all modules, NoAdaptive produces the worst performance, and the result of NoAttention is improved compared to NoODE and OneASTGCNsLayer. This phenomenon is consistent with the PeMS07 dataset. However, the performance of the different modules is smoother (i.e. closer to ASTGCNs) on PeMS08 dataset.

VI. Conclusion

In this article, we address the limitation that existing methods only consider local or global spatial-temporal information through modeling different local spatial-temporal information and global spatial-temporal information. From the local spatial-temporal information aspect, we adopt an adaptive graph learning to solve the bias in the explicit graph structure. Moreover, we employ an attention mechanism which exploits the use of spatial adjacency matrix to obtain high-quality information. From the global spatial-temporal information aspect, we employ a TCN block and ODE to construct a deeper network for long-term temporal dependencies. Finally, we integrate adaptive graph learning, attention mechanism, TCN block and ODE into multiple ASTGCNs layers comprising several blocks. Experimental results show that our proposed method outperforms the SOTA baselines on four datasets.

For future research direction, we will explore electricity usage prediction based on regions of city.

References

[1] M. Li, S. Chen, Y. Shen, G. Liu, I. W. Tsang, and Y. Zhang, "Online multi-agent forecasting with interpretable collaborative graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, 2022.
[2] X. Song, Y. Guo, N. Li, and L. Zhang, "Online traffic flow prediction for edge computing-enhanced autonomous and connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2101–2111, Mar. 2021.
[3] J. Chen, K. Li, H. Rong, K. Bilal, K. Li, and S. Y. Philip, "A periodicity-based parallel time series prediction algorithm in cloud computing environments," *Inf. Sci.*, vol. 496, pp. 506–537, 2019.
[4] J. Chen, K. Li, and P. S. Yu, "Privacy-preserving deep learning model for decentralized VANETs using fully homomorphic encryption and blockchain," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11633–11642, Aug. 2022.
[5] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transp. Res. Part C: Emerg. Technol.*, vol. 19, no. 4, pp. 606–616, 2011.
[6] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 5668–5675.
[7] Y. Guo, R. Du, X. Li, J. Xie, Z. Ma, and Y. Dong, "Learning calibrated class centers for few-shot classification by pair-wise similarity," *IEEE Trans. Image Process.*, vol. 31, pp. 4543–4555, 2022.
[8] R. Du, J. Xie, Z. Ma, D. Chang, Y.-Z. Song, and J. Guo, "Progressive learning of category-consistent multi-granularity features for fine-grained visual classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9521–9535, Dec. 2022.
[9] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1655–1661.
[10] H. Yao, X. Tang, H. Wei, G. Zheng, Y. Yu, and Z. Li, "Modeling spatial-temporal dynamics for traffic prediction," vol. 1, p. 9, 2018, *arXiv:1803.01254*.

[11] Y. Chen, X. Zou, K. Li, K. Li, X. Yang, and C. Chen, "Multiple local 3D CNNs for region-based prediction in smart cities," *Inf. Sci.*, vol. 542, pp. 476–491, 2021.

[12] Z. Chang, X. Zhang, S. Wang, S. Ma, and W. Gao, "Stam: A spatiotemporal attention based memory for video prediction," *IEEE Trans. Multimedia*, 2022.

[13] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017.

[14] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*.

[15] Y. Chen, Y. Hu, K. Li, C. K. Yeo, and K. Li, "Approximate personalized propagation for unsupervised embedding in heterogeneous graphs," *Inf. Sci.*, vol. 600, pp. 287–300, 2022.

[16] Y.-F. Li, Y. Gao, Y. Lin, Z. Wang, and L. Khan, "Time series forecasting using a unified spatial-temporal graph convolutional network," in *Proc. Preregister Workshop 34th Conf. Neural Inf. Process. Syst.*, 2020.

[17] S. Fang et al., "MS-Net: Multi-source spatio-temporal network for traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 7142–7155, 2022.

[18] M. Ye, J. Zhang, Z. Guo, and H. J. Chao, "Federated traffic engineering with supervised learning in multi-region networks," in *Proc. IEEE 29th Int. Conf. Netw. Protoc.*, 2021, pp. 1–12.

[19] C. Song, Y. Lin, S. Guo, and H. Wan, "Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 914–921.

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

[21] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *Proc. Int. Conf. Learn. Representations*, 2018.

[22] 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. Art. Intell.*, 2019, pp. 1907–1913.

[23] Z. Fang, Q. Long, G. Song, and K. Xie, "Spatial-temporal graph ode networks for traffic flow forecasting," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2021, pp. 364–373.

[24] C. Ma, Y. Zhao, G. Dai, X. Xu, and S.-C. Wong, "A novel STFSA-CNN-GRU hybrid model for short-term traffic speed prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 4, pp. 3728–3737, Apr. 2023.

[25] D. Jia et al., "An enhanced predictive cruise control system design with data-driven traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8170–8183, Jul. 2022.

[26] X. Fu, B. Fang, J. Qian, Z. Wu, J. Zhu, and T. Du, "Roadside traffic sign detection based on faster R-CNN," in *Proc. 11th Int. Conf. Mach. Learn. Comput.*, 2019, pp. 439–444.

[27] J. C. Perafan-Villota, O. H. Mondragon, and W. M. Mayor-Toro, "Fast and precise: Parallel processing of vehicle traffic videos using Big Data analytics," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 12064–12073, Aug. 2022.

[28] V. S. Sindhu, "Vehicle identification from traffic video surveillance using yolov4," in *Proc. IEEE 5th Int. Conf. Intell. Comput. Control Syst.*, 2021, pp. 1768–1775.

[29] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin, "DeepSTN : Context-aware spatial-temporal neural network for crowd flow prediction in metropolis," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 1020–1027.

[30] R. Huang, C. Huang, Y. Liu, G. Dai, and W. Kong, "LSGCN: Long short-term traffic prediction with graph convolutional networks," in *Proc. Int. Joint Conf. Artif. Intell.*, 2020, pp. 2355–2361.

[31] Y. Chen, K. Li, C. K. Yeo, and K. Li, "Traffic forecasting with graph spatial-temporal position recurrent network," *Neural Netw.*, vol. 162, pp. 340–349, 2023.

[32] 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.*, 2020, vol. 33, pp. 17804–17815.

[33] K. Guo, Y. Hu, Y. Sun, S. Qian, J. Gao, and B. Yin, "Hierarchical graph convolution networks for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 151–159.

[34] M. Fang, L. Tang, X. Yang, Y. Chen, C. Li, and Q. Li, "FTPG: A fine-grained traffic prediction method with graph attention network using big trace data," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5163–5175, Jun. 2022.

[35] D. Li and J. Lasenby, "Spatiotemporal attention-based graph convolution network for segment-level traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8337–8345, Jul. 2022.

[36] S. Huang, D. Wang, X. Wu, and A. Tang, "DSANet: Dual self-attention network for multivariate time series forecasting," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, 2019, pp. 2129–2132.

[37] J. An et al., "IGAGCN: Information geometry and attention-based spatiotemporal graph convolutional networks for traffic flow prediction," *Neural Netw.*, vol. 143, pp. 355–367, 2021.

[38] K. K. Thekumparampil, C. Wang, S. Oh, and L.-J. Li, "Attention-based graph neural network for semi-supervised learning," 2018, *arXiv:1803.03735*.

[39] B. Yang, Y. Kang, Y. Yuan, X. Huang, and H. Li, "ST-LBAGAN: Spatio-temporal learnable bidirectional attention generative adversarial networks for missing traffic data imputation," *Knowl.-Based Syst.*, vol. 215, 2021, Art. no. 106705.

[40] W. Zi, W. Xiong, H. Chen, and L. Chen, "TAGCN: Station-level demand prediction for bike-sharing system via a temporal attention graph convolution network," *Inf. Sci.*, vol. 561, pp. 274–285, 2021.

[41] 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.*, 2019, pp. 922–929.

[42] L.-P. Xhonneux, M. Qu, and J. Tang, "Continuous graph neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 10432–10441.

[43] G. E. Box and D. A. Pierce, "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models," *J. Amer. Stat. Assoc.*, vol. 65, no. 332, pp. 1509–1526, 1970.

Yibi Chen is currently working toward the Ph.D. degree in computer science with Hunan University, Changsha, China. He is also a Student with Nanyang Technological University, Singapore. His research interests include deep learning, data mining, graph representation learning, spatial-temporal prediction.

Yunchuan Qin received the Ph.D. degree in computer science and technology from Hunan University, Changsha, China, in 2019. He is currently the Director of the Experiment Center for Information Technology, College of Computer Science and Electronic Engineering, Hunan University. His research interests include autonomous unmanned systems and high-performance edge computing.

Kenli Li (Senior Member, IEEE) received the Ph.D. degree in computer science from the Huazhong University of Science and Technology, Wuhan, China, in 2003. He is currently the Dean and a Full Professor of computer science and technology with Hunan University, Changsha, China, and Director of National Supercomputing Center, Changsha. He has authored or coauthored more than 300 research papers in international conferences and journals, such as the Association for the Advance of Artificial Intelligence, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, Design Automation Conference, and IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS. His major research interests include parallel computing, deep learning and distributed computing. He is currently an Associate Editor for IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS and IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING.

Chai Kiat Yeo received the B.Eng. (1st class Hons.) and M.Sc. degrees in electrical engineering, from the National University of Singapore, Singapore, and the Ph.D. degree from the School of Electrical and Electronics Engineering, Nanyang Technological University (NTU), Singapore. She was an Assistant Principal Engineer with Singapore Technologies Electronics and Engineering Limited prior to joining NTU in 1993. She is an Associate Professor and was the Deputy Director of Centre for Multimedia and Network Technology (CeMNet) and the Associate Chair (Academic) with the School of Computer Science and Engineering, NTU. She is currently the Deputy Director and Programme Lead of Singtel Cognitive and Artificial Intelligence Lab for Enterprises@NTU. Her research interests include anomaly detection, machine learning, artificial intelligence, predictive operational analytics, ad hoc and mobile networks.

Keqin Li (Fellow, IEEE) is currently a SUNY Distinguished Professor of computer science with the State University of New York, New Paltz, NY, USA. He is also a National Distinguished Professor with Hunan University, Changsha, China. He has authored or coauthored more than 900 journal articles, book chapters, and refereed conference papers. He holds nearly 70 patents announced or authorized by the Chinese National Intellectual Property Administration. His research interests include cloud computing, fog computing and mobile edge computing, energy-efficient computing and communication, embedded systems and cyber-physical systems, heterogeneous computing systems, Big Data computing, highperformance computing, CPU-GPU hybrid and cooperative computing, computer architectures and systems, computer networking, machine learning, intelligent and soft computing. He is among the world's top five most influential scientists in parallel and distributed computing in terms of both single-year impact and career-long impact based on a composite indicator of Scopus citation database. He was the recipient of the several best paper awards. He has chaired many international conferences. He is currently an Associate Editor for the *ACM Computing Surveys*, and *CCF Transactions on High Performance Computing*. He has served on the Editorial Boards of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON CLOUD COMPUTING, IEEE TRANSACTIONS ON SERVICES COMPUTING, and IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING. He is an AAAS Fellow and an AAIA Fellow. He is also a Member of Academia Europaea (Academician of the Academy of Europe).