

# Local-global dynamic correlations based spatial-temporal convolutional network for traffic flow forecasting<sup>①</sup>

ZHANG Hong<sup>②</sup>(张红), GONG Lei, ZHAO Tianxin, ZHANG Xijun, WANG Hongyan  
(College of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, P. R. China)

## Abstract

Traffic flow forecasting plays a crucial role and is the key technology to realize dynamic traffic guidance and active traffic control in intelligent traffic systems (ITS). Aiming at the complex local and global spatial-temporal dynamic characteristics of traffic flow, this paper proposes a new traffic flow forecasting model spatial-temporal attention graph neural network (STA-GNN) by combining attention mechanism (AM) and spatial-temporal convolutional network. The model learns the hidden dynamic local spatial correlations of the traffic network by combining the dynamic adjacency matrix constructed by the graph learning layer with the graph convolutional network (GCN). The local temporal correlations of traffic flow at different scales are extracted by stacking multiple convolutional kernels in temporal convolutional network (TCN). And the global spatial-temporal dependencies of long-time sequences of traffic flow are captured by the spatial-temporal attention mechanism (STAtt), which enhances the global spatial-temporal modeling and the representational ability of model. The experimental results on two datasets, METR-LA and PEMS-BAY, show the proposed STA-GNN model outperforms the common baseline models in forecasting accuracy.

**Key words:** traffic flow forecasting, graph convolutional network (GCN), temporal convolutional network (TCN), attention mechanism (AM)

## 0 Introduction

Economic improvement and increased urbanization have led to a growing problem of traffic congestion in cities<sup>[1]</sup>. In order to alleviate this problem, governments and related organizations urgently need to develop intelligent transportation systems, of which traffic flow prediction is a key part. Accurate and real-time prediction is crucial for traffic management departments, which can help them rationally allocate road resources, reduce the risk of congestion, and provide residents with the best travel routes<sup>[2]</sup>. Early prediction methods include statistical<sup>[3]</sup> and machine learning<sup>[4]</sup> methods, but the former has limited accuracy and the latter is difficult to capture the spatial and temporal characteristics of large-scale data.

With the rapid development of big data and deep learning, more accurate traffic prediction models have emerged. In deep learning, spatial and temporal correlations of traffic flow are mainly extracted using convo-

lutional neural network (CNN), graph convolutional network (GCN), recurrent neural network (RNN), temporal convolutional network (TCN), and attention mechanism (AM). For example, Zhang et al.<sup>[5]</sup> and Hema et al.<sup>[6]</sup> integrated CNN, long-short term memory (LSTM), and gated recurrent unit (GRU) into spatial-temporal dependencies mining for traffic flow. Zhao et al.<sup>[7]</sup> designed the temporal graph convolutional network (T-GCN) model, which utilizes GCN to extract spatial features and then captures temporal features through GRU. The spatial-temporal graph convolution network bi-directional long short-term memory (STGCN-BiLSTM) model proposed by Wu et al.<sup>[8]</sup> extracts traffic flow's spatial and temporal correlations using GCN and a bi-directional long short-term memory (BiLSTM) neural network. The spatial-temporal complex graph convolution network (ST-CGCN) model designed by Bao et al.<sup>[9]</sup> uses stacked GCN layers to extract spatial features, and 3D-CNN combined with LSTM to extract temporal features, to effectively capture the spatial-temporal correlation of traffic flow. The

① Supported by the Key R&D Program of Gansu Province (No. 23YFGA0063), the National Natural Science Foundation of China (No. 62363022, 61663021), the Natural Science Foundation of Gansu Province (No. 22JR5RA226, 23JRRA886) and the Gansu Provincial Department of Education; Industrial Support Plan Project (No. 2023CYZC-35).

② To whom correspondence should be addressed. E-mail: zhanghong@lut.edu.cn.  
Received on Dec. 18, 2023

above studies are all based on CNN, GCN, and RNN to analyze the spatial-temporal correlations of traffic flow. However, the RNN cycle structure has many training parameters and slow convergence. Therefore, the new study abandons the RNN method of extracting temporal correlation and instead uses a network with convolutional operations to handle temporal correlation. For example, Cao et al. [10] utilized the enhanced diffusion convolutional network (EDCN) to capture the spatial correlation of nodes in the spatial-temporal sequence-to-sequence network (STSSN) model and employed the expanded causal convolution in TCN to mine the local temporal correlation. The multi-scale temporal dual graph convolution network (MD-GCN) model designed by Huang et al. [11] used gated temporal convolution and dual-graph convolution to extract temporal and spatial dependence, respectively.

AM has been widely applied in traffic flow forecasting tasks and has achieved good forecasting performance. For example, Wang et al. [12] developed an attention-based spatial-temporal graph attention network (ASTGAT) network, which introduced AM into TCN to capture dynamic temporal correlation and introduced a graph attention layer to enhance dynamic spatial correlation in the network directly. Zhang et al. [13] proposed a graph-based temporal attention (GTA) framework, which introduces AM to identify the relationships among temporal sub-modules adaptively and utilizes graph embedding technology on sensor networks to capture spatial dependencies better. The hierarchical spatial-temporal neural networks with attention mechanism (HSTAN) model proposed by Lian et al. [14] utilizes a multi-headed self-attention mechanism to extract dynamic spatial-temporal correlations simultaneously.

Although the relevant research work in recent years has achieved good prediction results, the following problems still need to be solved. GCN usually uses a fixed adjacency matrix to model the spatial relationship. Since the spatial correlation of the traffic network changes dynamically with time, the fixed adjacency matrix cannot capture this dynamic feature. Therefore, it is necessary to dynamically learn the hidden spatial structure relationship between road network nodes based on data to capture the spatial correlation within traffic flow. In addition, the traffic flow forecasting methods based on TCN and GCN extract the spatial-temporal characteristics through the local field of perception and lack the context state update of the global change of traffic flow, resulting in the failure of the model to capture the dependency relationship between distant nodes in the traffic network topology.

To address the problems in the existing methods, the spatial-temporal attention graph neural network (STA-GNN) traffic flow forecasting model is proposed in this paper. The main contributions of this work are as follows:

(1) The hidden spatial associations are learned by constructing a dynamic adjacency matrix, which can mine the hidden dynamic graph network structure from the data without prior knowledge. GCN captures traffic flow's dynamic local spatial features according to the learned graph network structure.

(2) A multi-kernel temporal convolution layer (MKTCL) is designed to extract long-term, short-term, and different-scale local time series traffic flow features by stacking multiple convolutional kernels in each layer of TCN. Meanwhile, dynamic spatial and nonlinear temporal characteristics at different time steps can be effectively mined by stacking the TCN with GCN.

(3) A spatial-temporal attention layer (STAtt Layer) is introduced to model the global spatial and temporal correlation of long time series of traffic flow, and the spatial-temporal information is adaptively fused to enhance the characterization capability of the model for global spatial-temporal modeling.

## 1 Framework of STA-GNN

In this paper, given a graph  $G = (V, E, A)$  to represent the spatial relationships of the nodes in a traffic road network,  $V$  is the set of road nodes,  $E$  is the set of road network edges, and  $A$  is the graph's adjacency matrix. Assuming  $N$  sensors are in a particular area, the resulting data sequence is as follows:  $\mathbf{X}_t = (x_t^1, x_t^2, x_t^3, \dots, x_t^N) \in \mathbb{R}^{C \times N}$ ,  $x_t^i$  represents the data values recorded by the  $i$ th sensor at time  $t$ , and  $C$  represents the quantity of traffic sequence features (such as speed, traffic volume, etc.). The traffic flow forecasting problem is defined as follows: given a historical data sequence  $\mathbf{X}_{in} \in \mathbb{R}^{C \times N \times T}$ , with a length of  $T$  as the model input, the goal is to learn a prediction function  $f(\cdot)$  to generate the predicted values  $\hat{\mathbf{Y}}$  for the next  $T$  time steps. The calculation is as follows:

$$\mathbf{X}_{in} = [X^{t-T+1}, X^{t-T+2}, \dots, X^t] \xrightarrow{f(\cdot)} \hat{\mathbf{Y}} \quad (1)$$

where the predicted values  $\hat{\mathbf{Y}} = [Y^{t+1}, Y^{t+2}, \dots, Y^{t+T}] \in \mathbb{R}^{F \times N \times T}$  and  $F$  represent the output feature dimensions.

The STA-GNN modeling framework proposed in this paper is shown in Fig. 1. The model mainly consists of graph learning layer, graph convolutional layer (GC Layer), temporal convolutional layer (TC Layer)

er), and STAtt layer. The MKTCL stacks multiple convolutional kernels in each layer to learn long-term, short-term, and different-scale local temporal features. The graph learning layer computes a dynamic adjacency matrix, which the GC layer then utilizes to capture the dynamic spatial characteristics of the traffic flow. The STAtt layer uses spatial attention (SAtt), temporal attention (TAtt), and gated fusion to model global spatial-temporal features. The network layers consisting

of TCN, GCN, and STAtt extract local spatial features, local temporal features, and global spatial-temporal features from three different perspectives; spatial, temporal, and spatial-temporal, respectively, which enhances the characterization ability of spatial-temporal modeling of the model. Meanwhile, these network layers are stacked together and connected to the output layer by skip connections.

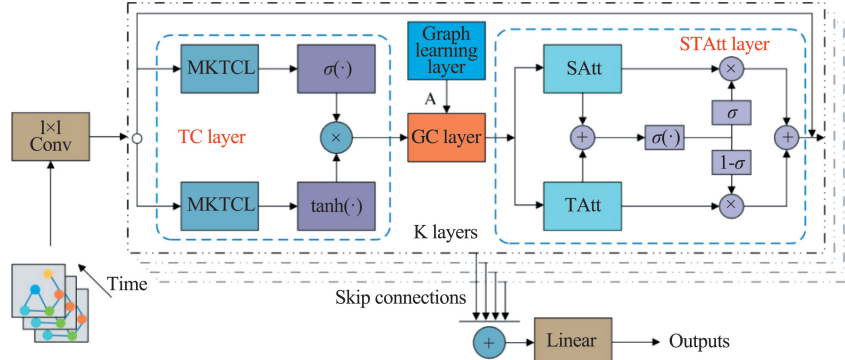


Fig. 1 The framework of STA-GNN

### 1.1 Graph learning layer

The traffic flow correlations existing between nodes in the METR-LA dataset and the PEMS-BAY dataset are weak in real traffic situations, and the learning relationship between nodes should be unidirectional, i. e. , traffic congestion occurring in the downstream will be transmitted to the upstream very quickly. In this paper, a dynamic graph learning layer specialized is proposed for extracting unidirectional relationships, which captures the complex multi-level hidden spatial associations among nodes of the road network by constructing a dynamic adjacency matrix.

To mine the correlation relationship in the process of unidirectional feature transfer between nodes, a dynamic adjacency matrix  $\mathbf{A}_{Dy}$  with asymmetric properties is constructed in the dynamic graph learning layer, whose structure is a lower triangular matrix. The specific calculation is as

$$\begin{aligned} \mathbf{M}_1 &= \tanh(\alpha \mathbf{E}_1 \boldsymbol{\Theta}_1) \\ \mathbf{M}_2 &= \tanh(\alpha \mathbf{E}_2 \boldsymbol{\Theta}_2) \end{aligned} \quad (2)$$

$$\mathbf{A}_{Dy} = \text{ReLU}(\tanh(\alpha(\mathbf{M}_1 \mathbf{M}_2^T - \mathbf{M}_2 \mathbf{M}_1^T)))$$

where  $\mathbf{E}_1 \in \mathbb{R}^{N \times U}$  and  $\mathbf{E}_2 \in \mathbb{R}^{N \times U}$  are the embedding representation vectors of the nodes,  $\boldsymbol{\Theta}_1 \in \mathbb{R}^{U \times U}$  and  $\boldsymbol{\Theta}_2 \in \mathbb{R}^{U \times U}$  are the parameters of the embedding representation vector transformations, and  $\alpha$  is a hyperparameter used to control the overfitting of the tanh.  $\mathbf{M}_1 \in \mathbb{R}^{N \times U}$  and  $\mathbf{M}_2 \in \mathbb{R}^{N \times U}$  represent the variable matrices required for constructing  $\mathbf{A}_{Dy}$ .

### 1.2 GC layer

Spatial correlation exists in two aspects: directly interconnected nodes, referred to as local spatial correlations between nodes, and nodes that are not directly connected but reachable through a path, referred to as global spatial correlations between nodes. In this section, GCN utilizes the inter-node association relationships obtained from the graph learning layer to aggregate node information with its neighbors' information to extract the local spatial features. In this paper, two graph convolution modules are designed in the GC layer to process the inflow and outflow information through each node. Fig. 2 shows the framework of the GC layer and the graph convolution module.

The graph convolution module extracts the information flow on the spatially associated nodes through the input graph adjacency matrix. The graph convolution module consists of the information pass step and the information selection step. The information pass step passes the node information recursively in the vertical direction along with the given graph structure, defined as follows.

$$\mathbf{H}^{(l)} = \delta \mathbf{H}_{in} + (1 - \delta) \tilde{\mathbf{A}}_{Dy} \mathbf{H}^{(l-1)} \quad (3)$$

where  $\delta$  is the retention rate, which is used to retain a certain percentage of the node's original information. The information selection step horizontally filters out the critical information generated in each message pass and passes it on to the next level, defined as follows:

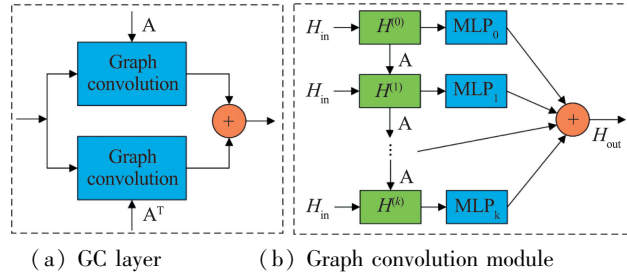


Fig. 2 The framework of the GC layer and the graph convolution module

$$\mathbf{H}_{out} = \sum_{i=0}^L \mathbf{H}^{(i)} \mathbf{J}^{(i)} \quad (4)$$

where  $L$  is the pass depth,  $\mathbf{H}_{in}$  and  $\mathbf{H}_{out}$  are the current layer's input and output hidden states.  $\tilde{\mathbf{A}}_{Dy}$  denotes the normalized adjacency matrix.  $\mathbf{J}^{(i)}$  is a parameter matrix used to filter important traffic flow information in the hidden layer.

### 1.3 TC layer

In the field of image processing, a widely used strategy called Inception takes the output of a 2D convolution with three different convolutional kernels ( $1 \times 1$ ,  $3 \times 3$ , and  $5 \times 5$ ) and stitches them together. Inspired by the Inception strategy, this paper propose the MKTCL. Fig. 3 illustrates the framework of the TC layer. MKTCL utilizes four differently sized convolutional kernels to extract the local temporal features at different scales, which are  $1 \times 2$ ,  $1 \times 3$ ,  $1 \times 6$ , and  $1 \times 7$ . Since the temporal patterns have several intrinsic periods, such as 7, 12, 24, 28, and 60, the  $1 \times 2$  convolutional kernel can extract the temporal features of the most recent moments, and a  $1 \times 7$  convolution kernel can represent cycle 7. To represent cycle 12, the input can be passed through a  $1 \times 7$  convolution kernel of the first MKTCL and then a  $1 \times 6$  convolution kernel of the second MKTCL. Thus, multiple periods are represented by stacking different convolutional kernels to concatenate convolutional receptive fields of various sizes. Therefore, by stacking different convolution kernels, convolution receptive fields of various sizes can be spliced to represent multiple time periods. Mathematically, given a one-dimensional input sequence, the convolutional kernels include  $k_{1 \times 2} \in \mathbb{R}^2$ ,  $k_{1 \times 3} \in \mathbb{R}^3$ ,  $k_{1 \times 6} \in \mathbb{R}^6$ ,  $k_{1 \times 7} \in \mathbb{R}^7$ . The output of each MKTCL is formed by concatenating the results of the four convolutional kernels, defined as follows:

$$m = \text{concat}(m \cdot k_{1 \times 2}, m \cdot k_{1 \times 3}, m \cdot k_{1 \times 6}, m \cdot k_{1 \times 7}) \quad (5)$$

where the outputs of the four convolutional kernels are split into equal lengths according to the giant convolutional kernel and connected across the channel dimensions,  $m \cdot k_{1 \times l}$  represents dilated convolution, which

can be defined as

$$m \cdot k_{1 \times l(t)} = \sum_{s=0}^{l-1} k_{1 \times l(s)m(t-d \times s)} \quad (6)$$

where  $d$  represents the dilation factor.

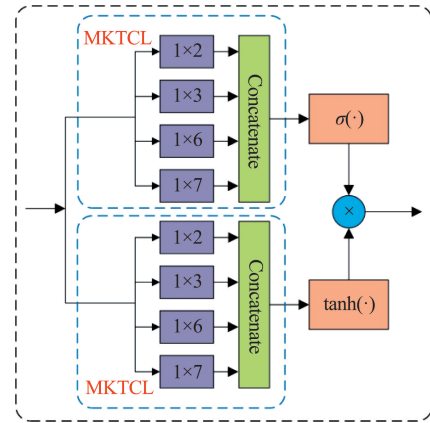


Fig. 3 The framework of the TC layer

### 1.4 STAtt layer

The framework of the STAtt layer is shown in Fig. 4, which consists of SAtt, TAtt, and gated fusion. The input data of the STAtt layer firstly get two outputs by SAtt and TAtt, and then these two outputs are fused by the gated fusion to get the final output.

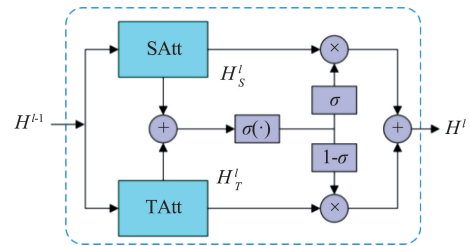


Fig. 4 The framework of STAtt layer

#### 1.4.1 SAtt

To model the global spatial features in the traffic data, the multi-head self-attention (MSA) is used to learn weights among different nodes and focus on the most critical node information. In this paper, denoting the input of the SAtt as  $\mathbf{H}^{(l-1)}$ , where the hidden state

of node  $v_i$  at time step  $t_j$  is represented by  $h_{v_i,t_j}^{(l-1)}$ ; and the output of the SAtt is denoted as  $\mathbf{H}_S^{(l)}$ , where the hidden state of node  $v_i$  at time step  $t_j$  is designated as  $hs_{v_i,t_j}^{(l)}$ .

MSA establishes different projection information in multiple projection spaces, projects the input matrix into different projection spaces, obtains multiple output matrices, and concatenates them together. Specifically, in this paper,  $K$  parallel attentions with different learnable projections are concatenated. At time step  $t_j$ , the correlation between nodes  $v_i$  and  $v$  can be represented as

$$s_{v_i,v}^{(k)} = \frac{[f_{s,1}^{(k)}(h_{v_i,t_j}^{(l-1)}) \cdot f_{s,2}^{(k)}(h_{v_i,t_j}^{(l-1)})]}{\sqrt{d}} \quad (7)$$

$$\alpha_{v_i,v}^{(k)} = \frac{\exp(s_{v_i,v}^{(k)})}{\sum_{v \in V} \exp(s_{v_i,v}^{(k)})} \quad (8)$$

where  $s_{v_i,v}^{(k)}$  is the correlation between nodes  $v_i$  and  $v$ ,  $\alpha_{v_i,v}^{(k)}$  is the attention score of the  $k$ th head, illustrating the importance of node  $v$  to  $v_i$ , and  $\sum_{v \in V} \alpha_{v_i,v}^{(k)} = 1$ . This paper describes the non-linear transformation as  $f(x) = \text{ReLU}(xW + b)$ , where  $W$  and  $b$  are learnable parameters.  $f_{s,1}^{(k)}(\cdot)$  and  $f_{s,2}^{(k)}(\cdot)$  represent two non-linear mappings in the  $k$ th head attention mechanism.  $d = D/K$ , where  $d$  is the dimensions of the attention head, and  $D$  is the number of channels in the layer. Then, attention scores  $\alpha_{v_i,v}^{(k)}$  are obtained to update the hidden state of node  $v_i$  at time step  $t_j$ , calculated as follows.

$$hs_{v_i,t_j}^{(l)} = \parallel_{k=1}^K \{ \sum_{v \in V} \alpha_{v_i,v}^{(k)} \cdot f_{s,3}^{(k)}(\cdot) \} \quad (9)$$

where  $f_{s,3}^{(k)}(\cdot)$  represents the non-linear mapping, ultimately producing  $d$ -dimensional output.

#### 1.4.2 TAtt

MSA is used to calculate the weights for different time steps to model the global temporal dependence across time steps and denote the input of the TAtt as  $\mathbf{H}^{(l-1)}$ , where the hidden state of node  $v_i$  at time step  $t_j$  is represented by  $h_{v_i,t_j}^{(l-1)}$ . The output of the TAtt is denoted as  $\mathbf{H}_T^{(l)}$ , where the hidden state of node  $v_i$  at time step  $t_j$  is denoted as  $ht_{v_i,t_j}^{(l)}$ . For node  $v_i$ , the correlation between time step  $t_j$  and  $t$  can be expressed as

$$u_{t_j,t}^{(k)} = \frac{[f_{t,1}^{(k)}(h_{v_i,t_j}^{(l-1)}) \cdot f_{t,2}^{(k)}(h_{v_i,t}^{(l-1)})]}{\sqrt{d}} \quad (10)$$

$$\beta_{t_j,t}^{(k)} = \frac{\exp(u_{t_j,t}^{(k)})}{\sum_{t_r \in N_{t_j}} \exp(u_{t_j,t_r}^{(k)})} \quad (11)$$

where  $u_{t_j,t}^{(k)}$  is the correlation between time steps  $t_j$  and  $t$ ;  $\beta_{t_j,t}^{(k)}$  is the attention score of the  $k$ th head, representing the importance of time step  $t$  to  $t_j$ ;  $f_{t,1}^{(k)}(\cdot)$  and

$f_{t,2}^{(k)}(\cdot)$  represent two different learnable transformations;  $N_{t_j}$  represents a set of time steps before time  $t_j$ .

Then, attention scores  $\beta_{t_j,t}^{(k)}$  are obtained to update the hidden state of node  $v_i$  at time step  $t_j$ .

$$ht_{v_i,t_j}^{(l)} = \parallel_{k=1}^K \{ \sum_{t \in N_{t_j}} \beta_{t_j,t}^{(k)} \cdot f_{t,3}^{(k)}(\cdot) \} \quad (12)$$

where  $f_{t,3}^{(k)}(\cdot)$  denotes the non-linear projection.

#### 1.4.3 Gated fusion

The traffic condition of a road will be jointly influenced by historical traffic information and other road traffic conditions, and this influence exhibits a global spatial-temporal correlation of traffic flow data. In this paper, to model the global spatial-temporal correlation, a gated fusion mechanism is designed to dynamically fuse the outputs of SAtt and TAtt and extract the global spatial-temporal features of the traffic flow data at each roadway node and time step. The fusion method can be expressed as

$$\mathbf{H}^{(l)} = p \cdot \mathbf{H}_S^{(l)} + (1-p) \cdot \mathbf{H}_T^{(l)} \quad (13)$$

$$p = \sigma(\mathbf{H}_S^{(l)} \mathbf{W}_{p,1} + \mathbf{H}_T^{(l)} \mathbf{W}_{p,2} + b_p) \quad (14)$$

where  $\mathbf{W}_{p,1} \in \mathbb{R}^{D \times D}$ ;  $\mathbf{W}_{p,2} \in \mathbb{R}^{D \times D}$ ;  $b_p \in \mathbb{R}^D$  are learnable parameters;  $\sigma$  is the Sigmoid function;  $\mathbf{H}_S^{(l)}$  denotes the output of the SAtt;  $\mathbf{H}_T^{(l)}$  denotes the output of the TAtt;  $\mathbf{H}^{(l)}$  denotes the final production; and  $p$  denotes the gate, which is used to control the information of the spatial-temporal attention layer.

## 2 Experiment

### 2.1 Datasets

This paper uses the METR-LA dataset from Los Angeles and the PEMS-BAY dataset from California to validate the predictive performance of the proposed STA-GNN. The data is collected at 30 s intervals, and each record is aggregated into 5 min intervals, meaning that each hour contains 12 traffic data samples. The detailed information on the experimental datasets is shown in Table 1.

Table 1 Description of the experimental dataset

Datasets	METR-LA	PEMS-BAY
Data type	Traffic speed	Traffic speed
Position	Los Angeles	San Francisco
	Freeway	Bay Area Freeway
Interval	5 min	5 min
Time steps	34 272	52 116
Time range	2012/3/1 –	2017/1/1 –
	2012/6/30	2017/6/30
Number of sensors	207	325

## 2.2 Settings

The data is divided into training, testing, and validation sets in the ratio of 7 : 2 : 1. Based on the PyTorch deep learning framework, the experiments were compiled and run on a Linux server (CPU: Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10 GHz, GPU: NVIDIA GeForce RTX 3060).

This paper sets the history and prediction time steps to 12 (1 h), i. e.,  $T = T^* = 12$ . A 3-layer STA-GNN network is used, the model optimizer is Adam, the initial learning rate is 0.001, the dropout rate is  $p = 0.3$ , the dimension of the node embedding representation vector set to 40, the epoch is set to 100, the batch size is set to 64<sup>[15]</sup>, the number of attention heads  $K = 8$ , and the dimension of each head's attention  $d = 8$ <sup>[16]</sup>. In addition, mean absolute error (MAE), root mean square error (RMSE), and mean

absolute percentage error (MAPE) are used to assess the model performance, and the formula is shown as

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (15)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (16)$$

$$MAPE = \frac{100}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (17)$$

where,  $y_i$  denotes the actual value,  $\hat{y}_i$  represents the predicted value, and  $n$  indicates the length of the time series.

## 2.3 Experimental results

As shown in Table 2, the prediction performance of STA-GNN and the baseline model is compared on two datasets. STA-GNN achieved excellent results for all the evaluated metrics.

Table 2 Comparison of STA-GNN and baseline model performance

Datasets	Models	15 min			30 min			60 min		
		MAE	RMSE	MAPE/%	MAE	RMSE	MAPE/%	MAE	RMSE	MAPE/%
METR-LA	HA	4.16	7.80	13.00	4.16	7.80	13.00	4.16	7.80	13.00
	ARIMA	3.99	8.21	9.60	5.15	10.45	12.70	6.90	13.23	17.40
	VAR	4.37	7.78	10.08	5.40	9.37	12.75	6.50	10.68	15.84
	FNN	3.99	7.94	9.90	4.23	8.17	12.90	4.49	8.69	14.00
	FC-LSTM	3.44	6.30	9.60	3.77	7.23	10.90	4.37	8.69	13.20
	WaveNet	2.99	5.89	8.04	3.59	7.28	10.25	4.45	8.93	13.62
	DCRNN	2.77	5.38	7.30	3.15	6.45	8.80	3.60	7.60	<b>10.50</b>
	STGCN	2.88	5.74	7.62	3.47	7.24	9.57	4.59	9.40	12.70
	AGCRN	3.33	6.50	8.36	4.20	7.97	9.50	4.95	9.41	11.21
	ASTGCN	4.86	9.27	9.21	5.43	10.61	10.13	6.51	12.52	11.64
PEMS-BAY	STSGCN	3.31	7.62	8.06	4.13	9.77	10.29	5.06	11.66	12.91
	STA-GNN	<b>2.68</b>	<b>5.21</b>	<b>7.00</b>	<b>3.04</b>	<b>6.21</b>	<b>8.54</b>	<b>3.53</b>	<b>7.36</b>	<b>10.63</b>
	HA	2.88	5.59	6.80	2.88	5.59	6.80	2.88	5.59	6.80
	ARIMA	1.62	3.30	3.50	2.33	4.76	5.40	3.38	6.50	8.30
	VAR	1.74	3.09	3.59	2.33	4.15	5.02	2.92	5.11	6.46
	FNN	2.20	4.42	5.19	2.30	4.63	5.43	2.46	4.98	5.89
	FC-LSTM	2.05	4.19	4.80	2.20	4.55	5.20	2.37	4.96	5.70
	WaveNet	1.39	3.01	2.91	1.83	4.21	4.16	2.35	5.43	5.87
	DCRNN	1.38	2.95	2.90	1.74	3.97	3.90	2.07	4.74	4.90
	STGCN	1.36	2.96	2.90	1.81	4.27	4.17	2.49	5.69	5.79
AGCRN	1.39	2.96	2.87	1.75	3.94	3.84	2.11	4.79	5.03	
ASTGCN	1.52	3.13	3.22	2.01	4.27	4.48	2.61	5.42	6.00	
STSGCN	1.44	3.01	3.04	1.83	4.18	4.17	2.26	5.21	5.40	
STA-GNN	<b>1.34</b>	<b>2.83</b>	<b>2.81</b>	<b>1.67</b>	<b>3.81</b>	<b>3.78</b>	<b>1.98</b>	<b>4.56</b>	<b>4.68</b>	

Table 2 shows that the prediction accuracy of deep learning-based methods (STA-GNN, STSGCN, ASTGCN, AGCRN, STGCN, and DCRNN) is higher than that of the other methods, which is attributed to the more powerful ability of deep learning methods to model the nonlinear and complex spatial-temporal nature of traffic data. Specifically, models that consider spatial and temporal correlations achieve better results than those that only think of temporal dependencies and ignore spatial correlations of traffic nodes. For example, at 60 min, compared with the VAR and FC-LSTM, the MAE of STA-GNN in dataset METR-LA was improved by about 45.69% and 19.22%, and the RMSE was improved by approximately 31.09% and 15.30%, respectively.

Meanwhile, models based on GCN (such as DCRNN) rely entirely on predefined graphs to represent the non-Euclidean distances of the traffic network. On the contrary, the STA-GNN proposed in this paper does not use predefined graphs but captures the complex multi-level spatial-temporal correlations between the nodes of the traffic network through the dynamic

adjacency matrix constructed by the graph learning layer. At 15 min, the MAE and RMSE of STA-GNN in the PMS-Bay dataset are improved by about 2.90% and 4.07% compared with the DCRNN method, respectively; at 30 min, MAE and RMSE improved by 4.02% and 4.03% respectively; at 60 min, MAE and RMSE improved by about 4.35% and 3.80% respectively.

In addition, STGCN and STSGCN use TCN or GCN to simulate temporal correlations. Due to their smaller receptive fields, they can only capture a small range of dependency characteristics, making it difficult to focus on long-term temporal information. In contrast, the proposed STA-GNN utilizes the advantages of TCN and GCN combined with AM to capture complex dynamic spatial-temporal characteristics of traffic flow effectively and has apparent advantages in long-term prediction. At 60 min, compared with STGCN and STSGCN, the MAE of STA-GNN in the dataset METR-LA was improved by about 23.09% and 30.24%, and the RMSE was improved by approximately 21.70% and 36.88%, respectively.

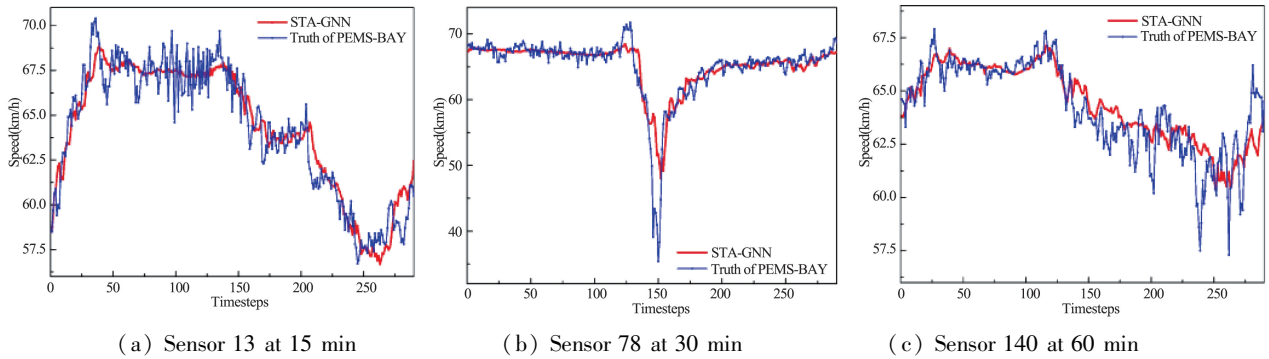


Fig. 5 Comparison of predicted results in PEMS-BAY

On the PEMS-Bay dataset, three sensors and any 288 consecutive time steps were randomly selected to compare the actual and predicted values of the STA-GNN, as shown in Fig. 5 (a), (b) and (c), and the prediction results of 15 min, 30 min, and 60 min were compared respectively. From the comprehensive view of the three figures, the STA-GNN model's predicted results can better simulate the changing trend of traffic speed. Although the gap between the actual and predicted values gradually increases as the prediction time window expands, the predicted value is still close to the actual value. It does not deviate from the fluctuation range of the actual value. From Fig. 5 (b), it can be observed that when there is a significant change

in speed, STA-GNN can also effectively simulate the change.

## 2.4 Ablation study

In this paper, ablation experiments were conducted on the METR-LA and PMS-Bay datasets to verify the validity of different modules in the STA-GNN model, and the variant models are represented as follows.

- (1) no-GCL: GC layer is removed.
- (2) no-mix-hop: GC module has no information selection step.
- (3) no-inception: there is no Inception in the MKTCL; only a  $1 \times 7$  convolution kernel is used.
- (4) no-STAtt: STAtt layer is removed.

Table 3 shows the traffic flow prediction results for the STA-GNN model at 15 min, 30 min, and 60 min

on the two datasets, respectively. All variant models have the same parameters and settings as STA-GNN.

Table 3 Performance comparison between STA-GNN and variant models

Datasets	Models	15 min			30 min			60 min		
		MAE	RMSE	MAPE/%	MAE	RMSE	MAPE/%	MAE	RMSE	MAPE/%
METR-LA	no-GCL	2.84	5.61	7.59	3.22	6.63	9.19	3.64	7.56	10.72
	no-Mix-hop	2.76	5.37	7.30	3.11	6.28	8.64	3.54	7.32	10.41
	no-Inception	2.73	5.27	<b>6.79</b>	3.08	6.21	<b>8.27</b>	<b>3.51</b>	<b>7.21</b>	<b>9.94</b>
	no-STAtt	2.76	5.34	6.86	3.11	6.32	8.47	3.54	7.38	10.05
	<b>STA-GNN</b>	<b>2.68</b>	<b>5.21</b>	7.00	<b>3.04</b>	<b>6.21</b>	8.54	3.53	7.36	10.63
PEMS-BAY	no-GCL	1.35	2.87	2.86	1.68	3.80	3.78	1.99	4.56	4.80
	no-Mix-hop	1.52	3.01	3.55	1.73	3.76	4.05	2.00	4.48	4.57
	no-Inception	1.35	2.84	2.87	<b>1.66</b>	<b>3.76</b>	<b>3.77</b>	<b>1.95</b>	<b>4.47</b>	<b>4.55</b>
	no-STAtt	1.34	2.83	2.83	1.67	3.79	3.78	1.95	4.49	4.62
	<b>STA-GNN</b>	<b>1.34</b>	<b>2.83</b>	<b>2.81</b>	1.67	3.81	3.78	1.98	4.56	4.68

The METR-LA dataset is used as an example for comparative analysis. It can be concluded that STA-GNN improves the MAE by about 5.63%, 2.90%, 1.83%, and 2.90%, and the RMSE by about 7.13%, 2.98%, 1.14%, and 2.43% for STA-GNN compared with no-GCL, no-Mix-hop, no-Inception, and no-STAtt, respectively, at 15 min.

Meanwhile, the experimental results of STA-GNN and variant models for the next 1 h prediction on the METR-LA dataset are visualized as shown in Fig. 6, with each time step representing 5 min. STA-GNN outperforms the variant model overall regarding MAE and RMSE, and STA-GNN outperforms the no-GCL and no-Mix-hop models overall in terms of MAPE. It shows that the GC Layer can capture the hidden spatial correlations in the traffic flow, the information selection method in the GC module can extract the more essen-

tial node features, MKTCL can portray the long, short, and multi-scale temporal patterns, and STAtt can simulate the global spatial-temporal correlations of the traffic flow. These modules play an essential role in the enhancement of the prediction performance.

In contrast, at 30 min and 60 min, the STA-GNN model does not perform as well as the variant model no-Inception in terms of prediction, because traffic flow data is usually highly time-dependent, i. e., recent data has a strong influence on future predictions. At shorter time scales (e. g., 15 min), the Inception model captures this time dependence better because it has more layers and parameters to extract more complex features. However, at longer time scales (e. g., 30 min and 60 min), the temporal dependence becomes relatively weaker, and thus a simpler model would be more effective.

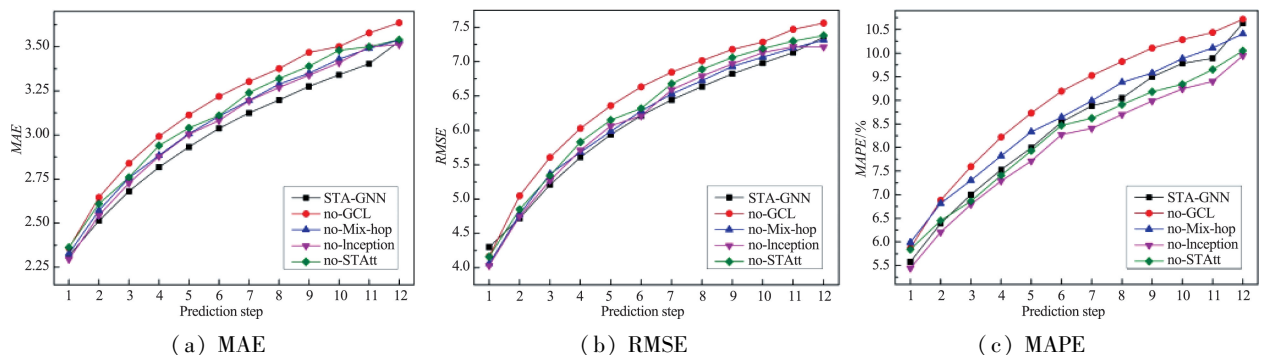


Fig. 6 Comparison of prediction results of STA-GNN and variant models on METR-LA

In order to verify the effect of different numbers of heads in spatial-temporal attention (STAtt) on the pre-

diction performance of the model, this paper has done a comparison experiment on the PEMS-BAY dataset as

shown in Fig. 7. It can be found that an appropriate increase in the number of heads in attention can improve the performance of the model. On the PEMS-BAY dataset, the model performance is best when  $K$  is 8 and  $d$  is 8, at which time the model can capture a wider range of spatial-temporal patterns and dependen-

cies in the traffic data; when it is less than 8, the model cannot adequately learn different data representations; when it is greater than 8, the complexity of the model increases and the increase in the number of parameters leads to a decrease in the computational efficiency and performance of the model.

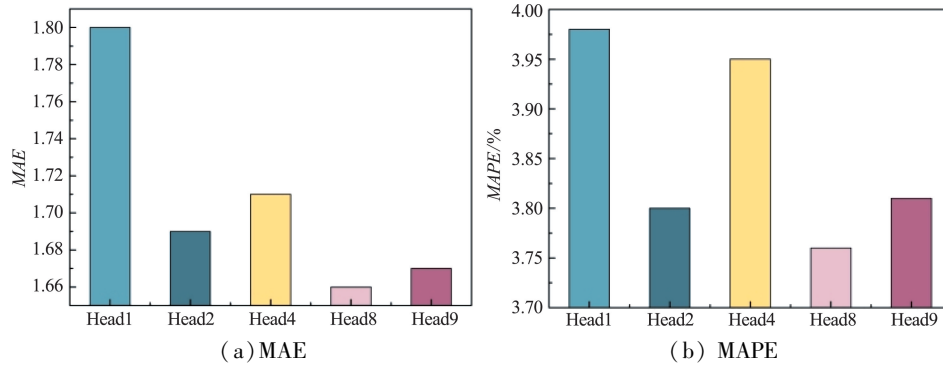


Fig. 7 Performance comparison of different head counts in METR-LA

### 3 Conclusion

To accurately predict future traffic road conditions, it is necessary to model the global spatial-temporal characteristics of traffic flow accurately. Therefore, this paper proposes the STA-GNN traffic flow prediction model based on AM and spatial-temporal convolutional networks. The model constructs a dynamic adjacency matrix to learn the hidden spatial correlations, which can discover the hidden dynamic graph network structure from the data without any a priori knowledge guidance. The GCN is used to capture the dynamic local spatial features of the traffic flow based on the learned graph network structure. A multi-core temporal convolutional network is designed to capture the long-time, short-time, and different-scale local time series traffic flow features by integrating multiple convolutional kernels, enabling the convolutional operation to obtain flexible and variable receptive fields. A spatial-temporal attention network is also proposed to model the global temporal dependence and spatial correlation of traffic flow and dynamically fuse the two to achieve traffic flow prediction incorporating global spatial-temporal features. Many experiments are conducted on real traffic datasets and compared with related studies. STA-GNN achieved better predictive performance under different datasets and different periods. Ablation experiments are also conducted on each module in the model to verify its effectiveness.

In future work, the external factors affecting traffic flow changes (e. g., holidays, weather, traffic accidents, etc.) will be further considered to establish a prediction model to improve the accuracy of traffic flow forecasting in complex environments.

### References

- [1] ZHANG Q Y, LI C W, SU F W, et al. Spatio-temporal residual graph attention network for traffic flow forecasting [J]. *IEEE Internet of Things Journal*, 2023, 10 (13): 11518-11532.
- [2] RAHMANI S, BAGHBANI A, BOUGUILA N, et al. Graph neural networks for intelligent transportation systems: a survey [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2023, 24(8): 8846-8885.
- [3] WANG K, MA C X, HUANG X T. Research on traffic speed prediction based on wavelet transform and ARIMA-GRU hybrid model [J]. *International Journal of Modern Physics C*, 2023, 34(10): 2350127.
- [4] LIN G C, LIN A J, GU D L. Using support vector regression and K-nearest neighbors for short-term traffic flow prediction based on maximal information coefficient [J]. *Information Sciences*, 2022, 608: 517-531.
- [5] ZHANG Y, XIN D R. A diverse ensemble deep learning method for short-term traffic flow prediction based on spatiotemporal correlations [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2022, 23 (9): 16715-16727.
- [6] HEMA D D, KUMAR K A. Optimized deep neural network based intelligent decision support system for traffic state prediction [J]. *International Journal of Intelligent Transportation Systems Research*, 2023, 21(1): 26-35.
- [7] ZHAO L, SONG Y J, ZHANG C, et al. T-GCN: a tempo-

- ral graph convolutional network for traffic prediction[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(9): 3848-3858.
- [8] WU S F. Spatio-temporal dynamic forecasting and analysis of regional traffic flow in urban road networks using deep learning convolutional neural network[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(2): 1607-1615.
- [9] BAO Y X, HUANG J S, SHEN Q Q, et al. Spatial-temporal complex graph convolution network for traffic flow prediction[J]. Engineering Applications of Artificial Intelligence, 2023, 121: 106044.
- [10] CAO S Q, WU L B, WU J, et al. A spatial-temporal sequence-to-sequence network for traffic flow prediction [J]. Information Sciences, 2022, 610: 185-203.
- [11] HUANG X H, WANG J Y, LAN Y C, et al. MD-GCN: a multi-scale temporal dual graph convolution network for traffic flow prediction[J]. Sensors, 2023, 23(2): 841.
- [12] WANG Y, JING C F, XU S S, et al. Attention based spatio-temporal graph attention networks for traffic flow forecasting[J]. Information Sciences, 2022, 607: 869-883.
- [13] ZHANG S K, GUO Y, ZHAO P Z, et al. A graph-based temporal attention framework for multi-sensor traffic flow forecasting[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(7): 7743-7758.
- [14] LIAN Q Y, SUN W, DONG W. Hierarchical spatial-temporal neural network with attention mechanism for traffic flow forecasting[J]. Applied Sciences, 2023, 13(17): 9729.
- [15] ZHANG X J, HAO J, NIE S Y, et al. MEEMD-DBA-based short term traffic flow prediction[J]. High Technology Letters, 2023, 29(1): 41-49.
- [16] ZHENG C, FAN X, WANG C, et al. GMAN: a graph multi-attention network for traffic prediction [C]//Proceedings of the AAAI Conference on Artificial Intelligence. New York, USA: AAAI Press, 2020: 1234-1241.

**ZHANG Hong**, born in 1977. She received her Ph. D, M. S. and B. E. degrees in Lanzhou University of Technology in 2018, 2004 and 2001 respectively. Her research interests include big data processing, data mining and analysis, machine learning and intelligent transportation system.