Journal of Network Intelligence
Taiwan Ubiquitous Information
©2024 ISSN 2414-8105
Volume **9**, Number **4**, November **2024**

MSTA-GNN: A Graph Neural Network with Multi-view Spatio-Temporal Attention for Traffic Flow Prediction

Rong Xiong*, Lyuchao Liao, Zihao Wang, Chunbo Wang, Hankun Xiao

Fujian Provincial Key Laboratory of Automotive Electronics and Electric Drive
Fujian Provincial Universities Engineering Research Center for Intelligent Driving Technology
Fujian University of Technology, Fuzhou 350118, China
xiongr@smail.fjut.edu.cn, fjachao@gmail.com, 906599055@qq.com, 17335833563@163.com, 847419825@qq.com

*Corresponding author: Rong Xiong

Received October 18, 2023, revised January 6, 2024, accepted May 14, 2024.

Abstract. *With the rapidly growing number of vehicles and traffic roads, predicting traffic flow presents a challenge due to the intricate nature of the traffic network topology, the diverse driving behaviors, and the potential impact of unforeseen emergencies on road conditions. To address the complexity of traffic flow forecasting concerning the spatial structure of the road network. In this work, In this paper, we proposed a new model that addresses the problem that current research is too homogeneous for the construction of spatial correlations through multiple spatial views Firstly, recognizing that historical data inherently carries dynamic information about the spatial structure of road networks, we propose a dynamic spatiotemporal similarity graph to replace the conventionally predefined static graph used in traditional graph convolution approaches. Secondly, we devise an enhanced gated graph attention module incorporating multi-scale gated graph attention mechanisms to capture temporal features from multiple perspectives, thereby bolstering the model's ability to perceive the dynamic time dependencies within the road network. The proposed method substantially improves state-of-the-art techniques through extensive experimentation on real-world datasets.*
Keywords: Traffic flow prediction, Graph convolution network, Multi-view, correlation matrix

1. **Introduction.** Traffic flow prediction plays a vital role in traffic management and planning [1]. It refers to predicting future traffic conditions in road networks, including traffic flow, speed, and congestion levels. With the deployment of large-scale transportation facilities on highways, a large amount of traffic data is generated. These generated traffic data contain the evolution rules of traffic flow, and each road network node presents a complex space-time relationship and dependency pattern. How to fully utilize the relationship between these historical flow data and road network nodes to mine various forms and spatiotemporal dependence patterns of traffic flow is of great significance to future traffic signal optimization, path planning, and real-time traffic management [2]. However, traffic flow data often have obvious spatio-temporal correlation and multi-scale nature. The changes in traffic flow at different time and space are related to each other, and the distribution and changes of traffic flow at different time and space scales are also different. How to effectively model and exploit these correlations and multi-scale properties is a challenge.

Deep learning research on traffic flow prediction has made impressive progress in recent years. Early research mainly focused on using traditional deep learning models such as recurrent neural networks (RNN) and convolutional neural networks (CNN) for traffic flow prediction. Traditional deep learning models still have certain limitations in capturing traffic flow data's nonlinear and dynamic characteristics. Secondly, traditional models usually ignore the interaction between the topology of the transportation network and the traffic flow.

To improve the shortcomings of current research, more complex deep learning models, such as graph neural networks (GNN) and self-attention mechanisms, are introduced to better capture traffic data's nonlinear and dynamic characteristics. Secondly, combine the traffic network topology information to design a model considering the interaction between traffic flows. For example, Ma et al. [3] proposed a traffic flow prediction method based on a space-time graph diffusion network. They utilize graph convolutional neural network (GCN) and temporal convolutional neural network (TCN) to capture traffic flow data's spatial and temporal dependencies, and graph diffusion to capture the propagation effect of traffic flow. Lv et al. [4] proposed a traffic flow prediction method based on a graph convolutional recurrent neural network. They combine graph convolutions and recurrent neural networks to achieve accurate traffic flow predictions by modeling traffic flow data's spatial and temporal dependencies.

To further improve the performance of the model, researchers began to explore the application of attention mechanism in traffic flow prediction. Guo et al. [5] proposed a traffic flow prediction method ASTGCN based on an attention network and spatiotemporal graph convolution network (ST-GCN). They introduced an attention mechanism to capture the correlation between different nodes and extracted spatial and temporal features through graph convolution to achieve more accurate traffic flow prediction. However, most existing GCN methods use predefined static adjacency matrices to describe the spatial correlation in the road network, which cannot truly reflect the dynamic changes in spatial dependence between road networks.

Recently, Chen et al. [6] proposed a new position graph convolutional network, which solves the problem of a predefined adjacency matrix by adding a learnable matrix and uses the absolute value of this matrix to represent the differences between different nodes and different levels of influence. Lan et al. [7] proposed a new dynamic spatio-temporal perceptual graph neural network and proposed a new data-driven strategy for dynamic spatio-temporal perceptual graphs to replace the predefined static graphs.

In addition, most time series traffic data have similar data patterns and similar functional relationships between road network sites. However, most of the existing graph-based spatiotemporal network models mine the temporal correlation of traffic flows from a single time series, lacking attention to short-term and long-term temporal correlations, and therefore have problems in capturing dynamic temporal correlations within the road network limitation.

To solve the problems mentioned above, we propose a new traffic flow prediction model named MSTA-GNN, which can effectively capture the spatiotemporal correlation of different periods of the road network. The main contributions of this paper can be summarized as follows:

(1) A correlation coefficient graph is constructed to capture the dynamic properties of spatial correlations between nodes by directly mining historical traffic flow data of nodes without using a predefined static adjacency matrix. The model is named Dynamic Spatio-Temporal Similarity correlation (DSTS), and it shows better capacity to capture the dynamic spatial correlation between sites.

(2) Spatio-temporal correlations for extracting traffic flow in three different historical periods are proposed. Three modules are used to learn the traffic relationship between sites in different periods, thereby obtaining the spatial correlation of the corresponding periods.

(3) An improved gated graph attention module is designed, which uses multi-scale gated graph attention to capture a variety of temporal features, further enhancing the model's perception of the dynamic time dependence of the road network. Extensive experiments on real road traffic datasets demonstrate the improved performance of our proposed algorithm compared to multiple baselines, including state-of-the-art algorithms.

The remainder of the paper is organized as follows. Section 2 presents the related works on Graph Convolution, Multi-view approach, and Spatio-Temporal prediction method; the proposed MSTA-GNN model is introduced in Section 3, and Section 4 presents the details of experiments and their results, including datasets, experimental setup, and the analysis of results. Finally, Section 5 concludes the paper.

2. **Related work.**

In this subsection, we first review and outline related work on Graph Convolution, followed by an overview of the Multi-view approach and spatiotemporal prediction model.

2.1. **Graph Convolution.**

Nowadays, graph convolutional neural networks are widely used in traffic flow prediction, usually including two methods. One is spectral GCN. Bruna et al. [8] used the Laplacian spectrum to extend the convolution operation on the graph in the spectral domain. However, calculating spectral domain convolution requires the calculation of all eigenvalues of the Laplacian matrix, which creates a computationally intensive problem. The ChebNet model proposed by Defferrard et al. [9] uses Chebyshev polynomials to expand the diagonal matrix based on eigenvalues to approximate graph convolution and reduce its computational complexity. In classic GCN [10], graph convolution is used in a CNN-like deep network framework to achieve effective embedding of graph structure and node attributes. The other is spatial GCN. Micheli and Alessio [11] performed graph convolution by directly summarizing the neighborhood information of nodes. Atwood et al. [12] regarded graph convolution as a diffusion process and introduced the probability of information propagation through different paths between any two nodes. Velickovic et al. [13] proposed a graph attention network in which an attention mechanism adjusts the weights between adjacent nodes.

2.2. **Multi-view based approaches.**

In recent years, more and more scholars have applied multi-view learning to traffic flow prediction. Multi-view methods for traffic flow prediction fall into two broad categories:

The first category is from the perspective of learning relationships between stations, and many methods capture various spatiotemporal dependencies in traffic flows by defining or learning different types of adjacency matrices.

The second category is from the perspective of learning temporal dependence. Some methods divide the data set into different subsets based on temporal attributes, such as periodic trends, closed flows, etc., or use clustering methods to divide historical data streams. Create multiple clusters for different attributes, and then learn different patterns of traffic characteristics from these clustered data.

Wang et al. [14] proposed a new multi-view bidirectional spatiotemporal graph network (Multi-BiSTGN) to capture the spatiotemporal dependence of traffic flow by constructing three views: closeness, daily degree, and weekly degree. Jin et al. [15] introduced a multi-view spatiotemporal virtual graph neural network (DMVST-VGNN) to predict online ride-hailing demand. DMVST-VGNN captures the dynamic spatial dependencies of traffic

flows by constructing distance graphs, association graphs, and mobility graphs. Li et al. [16] proposed a method of extracting dynamic time correlation using multi-view spatio-temporal graph neural network (MVST-GNN) by constructing near, medium, and long-distance three-view traffic data. Liu et al. [17] proposed a dynamic multi-view coupled graph convolution model to predict urban travel demand. However, the multiple attribute graphs constructed by the above method are all based on the same period, and traffic characteristics cannot be learned from multiple historical periods, making it challenging to capture deep spatiotemporal dependencies.

Zhou et al. [18] proposed a data-driven method called MOHER to predict crowd flow, which uses cities' geographical proximity and functional similarity to identify adjacent flow areas and utilizes cross-modal GCN to learn different patterns and correlations. Li et al. [19] proposed a multi-task synchronized graph neural network (MTSGNN) to predict the transition between regions, which uses multi-task graph representation learning to capture multiple types of dynamic spatial dependencies simultaneously. Huang et al. [20] proposed an attention mechanism for traffic flow prediction based on the convolutional LSTM model. The model uses clustering to learn the macro and micro patterns of traffic flow and uses the attention mechanism to combine two different levels of features.

However, the multi-view learning methods used above usually use static relationship matrices to capture the spatial characteristics of sites while ignoring the dynamics of relationships between sites over time, making it difficult for the model to capture the deeper spatiotemporal dependencies of traffic flow and failing to reflect Dynamic spatial dependence characteristics of traffic network conditions. In contrast, the model in this paper responds to the dynamic spatial dependence of the traffic network by constructing different relationship matrices.

2.3. **Spatiotemporal prediction methods.** Researchers have recently proposed various deep-learning methods to capture the spatiotemporal correlation of traffic prediction. In the ASTGCN model proposed by Guo et al. [5], the attention mechanism is incorporated into standard convolution to update node information by fusing information from adjacent time slices. However, the spatial dependence of the ASTGCN model only comes from the static adjacency graph structure so this method may miss potential dynamic dependence information. The Graph WaveNet model proposed by Wu et al. [21] and the AGCRN model proposed by Bai et al. [22] discover hidden spatial dependencies by embedding learnable nodes. However, these models cannot stack spatiotemporal layers while expanding the perceptual domain. Park et al. [23] and Wang et al. [3] utilize the self-attention mechanism to model spatiotemporal correlation. However, due to the use of autoregressive mechanisms, these algorithms are prone to error accumulation during the inference phase.

There are also some scholars' methods that focus on designing new graph structures. The STFGCN model proposed by Li et al. [24] builds a spatio-temporal fusion graph for traffic prediction based on the research of Song et al. [25] and supplements historical sequence information based on the static adjacent graph. The STGODE model proposed by Fang et al. [26] is based on the combination of the semantic adjacency matrix and the static space adjacency matrix and introduces ordinary differential equations (ODE) into GCN, in which the semantic adjacency matrix is also calculated using DTW.

However, these models do not explicitly consider the dynamic spatiotemporal dependencies between road network nodes. The models mentioned above although their performance is good, the spatial dependence derived from these models cannot well reveal their dynamic nature due to the use of predefined static neighbor graphs.

3. **Preliminaries.**

3.1. **Problem Description.** We represent the road network graph as a graph $G = (V, E)$, where V represents the set of N nodes in the road network, and E represents the set of connectivity edges between nodes. The adjacency matrix of G is expressed as $A = A_{(i,j)} \in \mathbb{R}^{(N\times N)}$, if $V_i, V_j \in V$ and $(V_i, V_j) \in E$, then $A = A_{(i,j)}$ are equal to 1, then the traffic status at any time can be regarded as a graph signal $X_t \in \mathbb{R}^{(N\times C_n)}$ where is the number of parameter types in the traffic data. In this work, we predict only one parameter type: traffic flow (C_n hence 1). Given the recorded data $X_{t_0:t_0+\eta-1} \in \mathbb{R}^{(N\times C_n\times \eta)}$, the traffic volume on the road network G in the future T time steps $X_{t_0:t_0+T-1} \in \mathbb{R}^{(N\times C_n\times T)}$ can be predicted by training the model f. The formula is as Equation (1):

$$X_{t_0:t_0+\eta-1} = f[X_{t-\eta+1:t}; G] \tag{1}$$

3.2. **Create a relationship matrix.** Based on the above graph structure, we established a relevant relationship matrix, which determines the path of information transmission in the graph, which is the key to performing graph convolution operations. We use the graph signal $X_{t_0:t_0+\eta-1}$ in the eta time interval and assign the similarity of $\hat{A^{(0)}}$ historical traffic between stations as a weight on each edge in the graph, thereby obtaining the traffic relationship matrix:

$$A_{rel} = F_{sim}[X_{t_0:t+\eta-1}] \tag{2}$$

$$\hat{A^{(0)}} = Norm(A_{rel}) \tag{3}$$

where t_0 represents the first time interval used to generate the relationship matrix, and η represents the number of time intervals of historical data. By performing a graph convolution operation on the graph G, represented by the relationship matrix $\hat{A^{(0)}}$, the spatial correlation between sites can be learned.

4. **Methodology.**

4.1. **Network architecture.** The proposed MSTA-GNN is shown in the Figure 1 and consists of stacked spatial-temporal blocks and prediction layers. Each ST block is concatenated and concatenated with the original input through residual connections and then sent to the prediction layer. The specific details of the model are as follows:

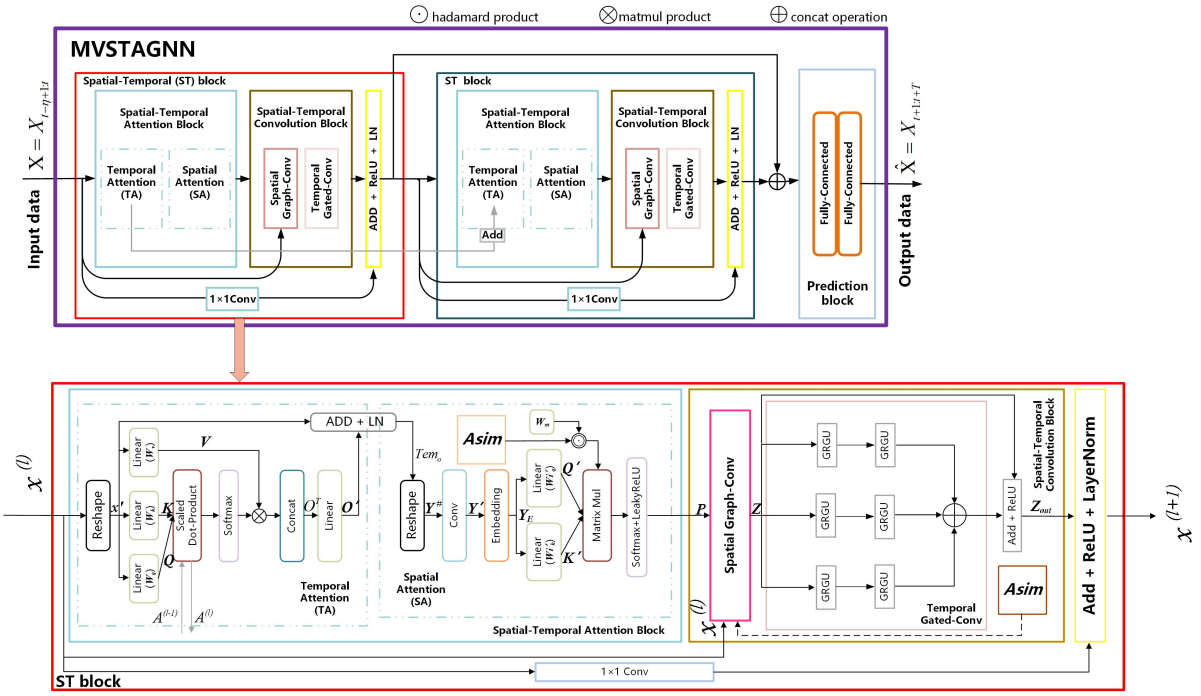


FIGURE 1. Architecture of MSTA-GNN

The overall framework is composed of several spatio-temporal blocks and a prediction block. The detailed map shows the detailed information of the ST block, which consists of a spatiotemporal attention block and a spatiotemporal convolution block. The spatiotemporal attention block includes temporal and spatial attention blocks. Spatiotemporal convolution blocks include temporal convolution blocks and spatial convolution blocks. The Dynamic Spatiotemporal Similarity Graph (DSTSG) is added to both the spatial attention block and the spatiotemporal convolution block to adjust the spatiotemporal attention further, and the predefined static adjacency graph used in traditional graph convolution is replaced with one with dynamic spatial dependence. Similar space-time diagram.

The similarity matrix part mainly replaces the traditional predefined adjacency matrix with a similarity matrix with dynamic spatial dependence, helping the Attention module adjust spatiotemporal attention. The Spatial-Temporal Multi-Attention Block part of this method mainly realizes the combination of spatio-temporal attention and further enhances the expression function of dynamic spatio-temporal correlation. The Spatial-Temporal Convolution Block part mainly realizes the further extraction of more meaningful traffic network features in the spatial dimension and captures the temporal dynamics of traffic flow data.

4.2. **Generation of correlation matrix.** The relationship matrix of a graph plays a crucial role in learning feature vectors between nodes. Previous methods usually define relationship matrices based on connectivity or distance between nodes, but this method cannot accurately capture the dynamic spatial correlation between sites. To solve this problem, this paper derives an initial relationship matrix based on the relationship matrix construction method in related work and captures the spatial correlation between nodes through dynamic learning.

The above statement emphasizes the importance of the relationship matrix for learning feature vectors between nodes, and the limitations of previous methods are pointed out.

By deriving an initial relationship matrix based on related work and introducing a new normalization method, this paper aims to more accurately capture the dynamic spatial correlation between sites. In addition, norm normalization is used to maintain the sparsity of node connections. The normalization formula is as follows:

$$A^{(0)} = \mathrm{ReLU}\left(E_1^{(0)} E_2^{(0)^T}\right) \tag{4}$$

$$D = \Sigma_j A^{(0)} \tag{5}$$

$$D_1 = \mathrm{diag}\left(\frac{1}{D}\right) \tag{6}$$

$$\widetilde{A}^{(0)} = D_1 A^{(0)} \tag{7}$$

Among them, E_1 and E_2 represent the two sub-embedding matrices obtained by SVD decomposition $A^{(0)}$, and $\widetilde{A}^{(0)}$ represents the normalized correlation matrix. We name the obtained correlation as Dynamic Spatiotemporal Similarity (DSTS) and its structure as Dynamic Spatiotemporal Similarity Graph (DSTSG).

4.3. **Spatiotemporal graph attention module.** Dynamic spatiotemporal similarity graph (DSTSG) can provide more accurate relationships between nodes, but the dynamic characteristics of these relationships need to be further refined to adapt to changes in real-time data. To this end, we propose a new spatiotemporal attention module that aims to enhance the representation of dynamic spatiotemporal dependencies further. This module enhances the expression of spatiotemporal dynamic correlations through sequential combination.

The above description emphasizes the accuracy of the dynamic spatiotemporal similarity graph for node relationships, and the need for further refinement in adapting to real-time data changes is pointed out. To meet this need, a new spatiotemporal attention module is introduced to enhance the representation of dynamic spatiotemporal dependencies. This module enhances the expression of spatiotemporal dynamic correlation through a sequential combination method.

4.3.1. *Time attention.* The multi-head self-attention mechanism can capture different focus and representation subspaces by applying multiple attention heads in parallel and then merging their results to obtain the final representation. This can better capture long-term correlations in the input time series. We exploit this mechanism to capture dynamic temporal dependencies between nodes.

For the multi-head attention with h heads, we define the variables Q, K, V as:

$$Q_i = XW_{Q_i} \tag{8}$$

$$K_i = XW_{K_i} \tag{9}$$

$$V_i = XW_{V_i} \tag{10}$$

among them, W_{Q_i}, W_{K_i} and W_{V_i} are parameter matrices for each attention head i, used for linear transformation. $X^{(l)} \in \mathbf{R}^{c^{(l-1)} \times M \times N}$ is derived from the lth ST block $X^{(l)}$ reshape, which represents the $c^{(l-1)}$-dimensional feature extracted from the N recording points output by the $l-1th$ layer at the time step of $t-\eta+1, t-\eta+2, ..., t$ dimensional features. The attention weight for each attention head is calculated as follows:

$$Att(Q^{(l)}, K^{(l)}, V^{(l)}) = Softmax(A^{(l)})V^{(l)} \tag{11}$$

$$A^{(l)} = \frac{Q^{(l)}(K^{(l)})^{\top}}{\sqrt{d_h}} + A^{(l-1)} \tag{12}$$

where $A^{(l-1)}$ is the output of the previous layer's attention.

At the same time, the residual attention idea is used to directly connect the output of each st block with the output of the next st block to enhance the connection between different levels of temporal attention. This allows the model to learn both shallow time dependence and deep time dependence simultaneously, which can alleviate the vanishing gradient problem while utilizing the dynamic time dependence in the traffic data stream.

Then Q, K, and V are projected H times using H different matrices and then spliced together.

$$O^{(head)} = Att(Q_i, K_i, V_i) \tag{13}$$

$$O_T = concat(O^1, O^2, ..., O^{head}) \tag{14}$$

Where head represents the $H-th$ attention head, and $O^{(head)}$ represents the output of the $h-th$ attention head.

It is then added to the input of the residual connection and passed through the normalization layer to get the output of the temporal attention layer, which is input to the spatial attention (SA) module. The formula is as follows:

$$Tem_O = LayerNorm(Linear(O_T) + X) \tag{15}$$

where Tem_O is the final output of temporal attention, and LayerNorm is layer normalization.

4.3.2. *Spatial attention.* The temporal attention module adaptively encodes time series data and obtains feature representations with global dynamic temporal dependence [27]. In terms of extracting spatial dependence, we designed an improved multi-head graph attention mechanism to obtain. The weight coefficients of the two branches (i.e., Query and Key) from the input embedding vector from the temporal attention module are calculated. However, the obtained weight coefficient is not used to weight the Value branch of the input embedding vector but is used to adjust the correlation coefficient map.

We first use the weight matrix to generate Q, K, V, whose dimensions are $W_q^{(h)}$, $W_k^{(h)}$, $W_v^{(h)} \in \mathbf{R}^{D\times H\times h}$. In forward propagation, we first linearly transform the node features and then calculate the weight of the multi-head attention. Finally, the values are weighted and aggregated using attention weights. The formula of Linear transformation is as follows:

$$q^h = Tem_O W_q^{(h)} \tag{16}$$

$$k^h = Tem_O W_k^{(h)} \tag{17}$$

$$v^h = Tem_O W_v^{(h)} \tag{18}$$

Among them, Tem_O is the output of the TA module $W_1^{(h)}$, $W_k^{(h)}$, $W_v^{(h)}$ are the weight matrices corresponding to each attention head h.

The input Tem_O is mapped to obtain $Tem_O^{'}$ two-dimensional matrix a, representing the set of embedding vector representations of each recording point. Then add the position information to $Tem_O^{'}$ through the embedding layer to get $Tem_E^{'}$. At the same time, when calculating the self-attention weight of the graph, we correct the calculation of the attention module through the DSTS graph. The calculation formula for self-attention weight and weight splicing is as follows:

$$P^{(h)} = \frac{exp(LeakyReLU(a^T[W_q^h h_i | W_k^h h_j | W_s^h h_i]))}{\sum_{j' \in \mathcal{N}_i} exp(LeakyReLU(a^T[W_q^h h_i | W_k^h h_j | W_s^h h_i]))} + W_m \odot A_{sim} \tag{19}$$

$$P = [P^{(1)}, P^{(2)}, ..., P^{(H)}] \tag{20}$$

Here, we introduce a new self-attention term, namely $W_s^h h_i$. It represents the self-attention item of node i, which is used to capture the characteristic relationship of the node itself. k is the number of attention heads; W_m is a learnable parameter to adjust the learning of attention by the similarity graph. a is the learnable parameter vector in the $h-th$ attention head;

4.4. **Spatiotemporal graph convolution module.**

4.4.1. *Spatial graph convolution.* Currently, many studies focus on the connectivity and globality of traffic networks and use predefined graph structures to perform graph convolution operations to obtain node features by aggregating information from adjacent nodes [28]. To fully utilize the topological characteristics of transportation networks, we adopt the above ideas and a graph convolution method based on Chebyshev polynomial approximation to learn structure-aware node features. However, unlike existing methods, we use correlation coefficient graphs instead of predefined graph structures. Furthermore, we dynamically adjust each term of the Chebyshev polynomial to extract more meaningful and broader traffic network features in the spatial dimension.

The above statement emphasizes the trend in current research to focus on the connectivity and globality of traffic networks and the method of using predefined graph structures for graph convolution. This paper adopts a similar idea, but unlike existing methods, we use correlation coefficient maps to capture the topological characteristics of transportation networks. Furthermore, by dynamically adjusting each term of the Chebyshev polynomial, we can extract more meaningful and broader traffic network features. In this paper, the scalar Laplacian matrix of the Chebyshev polynomials is defined as:

$$\widetilde{L} = \frac{2}{\lambda_{max}}(D - A^\star) - I_N \tag{21}$$

Where A^* is the correlation coefficient matrix graph DSTSG, I_N is the identity matrix, and D is the degree matrix. λ_{max} is the largest eigenvalue of the Laplacian matrix.

In graph convolution, the information at each node is derived from the nodes in its domain. To incorporate the dynamic properties of nodes, we use the K-order Chebyshev polynomial T_k to aggregate the information of the graph signal $x \in \mathbf{N}$ at each time step. The formula is as follows:

$$g_\theta * Gx = g_\theta(L)x = \sum_{k=0}^{K-1} \theta_k(T_k(\widetilde{L}) \odot P^{(k)})x \tag{22}$$

where g_θ is the approximate convolution kernel, $*G$ represents the graph convolution operation, and the learnable vector $\theta \in \mathbf{R}^k$ contains polynomial coefficients. $P^{(k)} \in \mathbf{R}^{N \times N}$ is the spatiotemporal attention matrix of the kth attention head. Finally, each node can aggregate information from adjacent nodes of order $0 \sim (K-1)$.

4.4.2. *Gated Recurrent Graph Attention Module(GRGU).* As a variant of RNN, GRU can better solve the vanishing gradient problem. Inspired by Li et al. [16], we use GAT to replace the linear transformation in GRU, and the output given by the spatial convolution is first passed through GAT to capture the temporal dynamic information of traffic flow data. In this paper, we use three gated recurrent graph convolution modules to extract the temporal characteristics of traffic flow data in three different periods. Gated recurrent graph convolution (GRGU) on traffic data for hourly view periods is defined as:

$$\begin{aligned} r_D^{(t)} &= \sigma\left(\Theta_r *_{GAT}\left[h_D^{(t)}, O_D^{(t-1)}\right] + b_r\right) \\ u_D^{(t)} &= \sigma\left(\Theta_u *_{GAT}\left[h_D^{(t)}, O_D^{(t-1)}\right] + b_u\right) \\ c_D^{(t)} &= \tan h\left(\Theta_c *_{GAT}\left[h_D^{(t)}, \left(r_D^{(t)} \odot O_D^{(t-1)}\right)\right] + b_c\right) \\ O_D^{(t)} &= u_H^{(t)} \odot O_H^{(t-1)} + \left(1 - u_D^{(t)}\right) \odot c_D^{(t)} \end{aligned} \tag{23}$$

where $h_D^{(t)}$ and $O_D^{(t)}$ represent the output of GAT and GRU respectively at the t time interval in each day, and $\odot$ represents the Hadema product. σ is the activation function. $\Theta_r, \Theta_u, \Theta_c$ are the corresponding filtering parameters. You can also obtain the recent and weekly cycle views through these operations.

5. Experiments and result analysis.

To evaluate the performance of the model, comparative experiments were conducted on two real traffic datasets. In addition, we further conducted ablation experiments to demonstrate the effectiveness of the different modules.

5.1. Datasets.

To evaluate the performance of the model, we conducted comparative experiments on four sets of real road traffic data sets PEMS04, PEMS03, PEMS07, and PEMS08 released by California, USA. These data sets are provided by [25]. Raw traffic data are aggregated into 5-minute intervals and normalized to zero mean. And construct the spatial adjacency graph of each data set based on the actual road network in Table 1.

Table 1. Description and statistics of the datasets

Datasets	Node	Edges	Timesteps	Missing Rate
PeMS03	358	547	26208	0.672%
PeMS04	307	340	16992	3.182%
PeMS07	883	866	28224	0.452%
PeMS08	170	295	17856	0.696%

5.2. Experimental environment and parameters.

For the sake of fairness, we divide the data into a training set and a validation set and then test the method in the same way as the baseline, that is, 6:2:2 on the PEMS dataset. We use one hour of historical data to predict the next hour's Stream flow traffic. All experiments were run on the same platform, NVIDIA 3090, 24GB card. The training process is implemented using PyTorch 1.10.1 in the Python 3.8.10 environment for all deep learning models. We set the following hyperparameters: The number of terms of the Chebyshev polynomial (equal to the number of spatial attention heads) K=3. The pooling layer window size W is set to 2. The number of attention heads in the spatiotemporal attention module is 32. All graph convolutional layers and temporal convolutional layers use 32 convolution kernels. All experiments use two spatiotemporal module stacks. In this paper, we use MSE as the loss function. We adopt the Adam optimizer to train our model with epoch 150, learning

rate 0.0001, and batch size 64. Mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE) are used to measure the performance of the model.

5.3. **Evaluation Metric and Baselines.** In our experiments, we use root mean square error ($RMSE$), mean absolute error (MAE), and mean absolute percentage error ($MAPE$) as metrics to evaluate the quality of the model:

$$MAE = \frac{1}{n}\sum_{i=1}|\hat{y}_i - y_i| \tag{24}$$

$$RMSE = \sqrt{\frac{1}{z}\sum_{i}(y_i - \hat{y}_i)^2} \tag{25}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{\hat{y}_i} \tag{26}$$

Among them, y_i and $\hat{y}_i$ represent the site's actual value and predicted value, respectively. n represents the number of all predicted values;

The baselines compared to the proposed model include both traditional and state-of-the-art methods.

i) FC-LSTM [29]: It is a special RNN model, a recurrent neural network with a fully connected network.
ii) TCN [30]: It is a time series modeling method based on convolutional neural networks that uses one-dimensional convolution operations to capture long-term dependencies in time series data.
iii) DCRNN [31]: Integrating graph convolution into gated recurrent units, graph convolution, and LSTM are used to capture traffic's spatial and temporal dependence, respectively.
iv) STGCN [28]: Integrate graph convolution into a one-dimensional convolution unit and use graph convolution and gated CNN to extract spatiotemporal features of traffic data.
v) ASTGCN [5]: Attention-based spatio-temporal graph convolutional network, which utilizes spatio-temporal attention mechanism to model spatio-temporal correlation.
vi) STSGCN [25]: Includes a local spatiotemporal subgraph module that considers spatial and temporal information.
vii) AGCRN [22]: The learnable embedding of nodes is utilized in graph convolution, and an attention mechanism is introduced to strengthen the correlation between nodes.
viii) STGODE [26]: In multivariate time series forecasting, the concepts of graph convolutional neural network (GCN) and ordinary differential equations (ODE) are combined to apply continuous graph neural network to traffic forecasting.

5.4. **Experimental Results and Analysis.** The Table 2 shows the comparison results of MSTA-GNN and eight baseline methods. It can be seen that our model achieves the best results on all three indicators in the four data sets. The dynamic spatio-temporal similarity graph proposed by us can help the model better capture the dynamic spatial dependence between nodes, which shows that our model can achieve better results than models using pre-defined graphs without a pre-defined adjacency matrix.

TABLE 2. Performance comparison of MSTA-GNN

Datasets	Metric	FC-LSTM	TCN	DCRNN	STGCN	ASTGCN	STSGCN	AGCRN	STGODE	MSTA-GNN
PEMS03	MAE	21.41	19.37	18.38	17.60	17.13	17.03	15.94	16.61	**15.92**
	MAPE(%)	22.48	19.85	18.97	17.32	19.11	16.90	15.43	16.50	**14.86**
	RMSE	35.20	34.03	30.54	30.06	29.10	28.93	28.39	27.91	**27.02**
PEMS04	MAE	26.44	23.37	24.70	22.84	22.58	21.20	19.89	20.41	**19.33**
	MAPE(%)	19.40	15.58	17.32	14.60	16.59	13.90	12.98	13.79	12.88
	RMSE	40.50	37.31	38.22	35.45	35.20	33.67	32.29	32.87	**31.41**
PEMS07	MAE	29.89	32.66	25.34	25.38	28.10	24.37	22.31	22.59	**21.73**
	MAPE(%)	14.40	14.39	11.56	11.10	13.78	10.24	9.62	9.58	**9.54**
	RMSE	43.80	42.24	38.60	38.81	42.55	39.03	35.54	37.45	**35.05**
PEMS08	MAE	22.20	22.66	17.96	18.12	18.65	17.23	15.98	16.80	**15.77**
	MAPE(%)	15.12	14.06	11.50	11.47	13.09	11.03	10.10	10.77	**9.97**
	RMSE	33.06	35.80	27.85	28.19	26.80	25.32	26.19	25.88	**24.98**

In addition, our proposed spatiotemporal attention mechanism can better capture the dynamic changes of data to improve prediction performance. We quantified the test data and plotted 60 minutes of predicted values versus true values. As shown in the Figure 2, it can be seen from the peak point that MSTA-GNN predicts peak changes relatively well, and the trend can fit well with the real value.

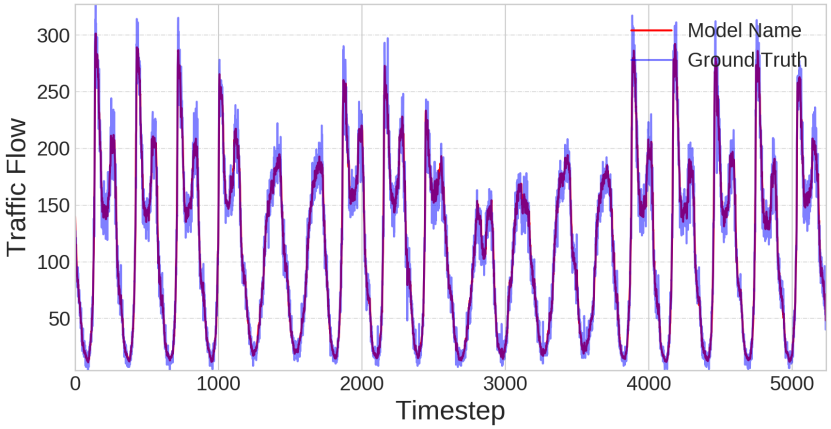
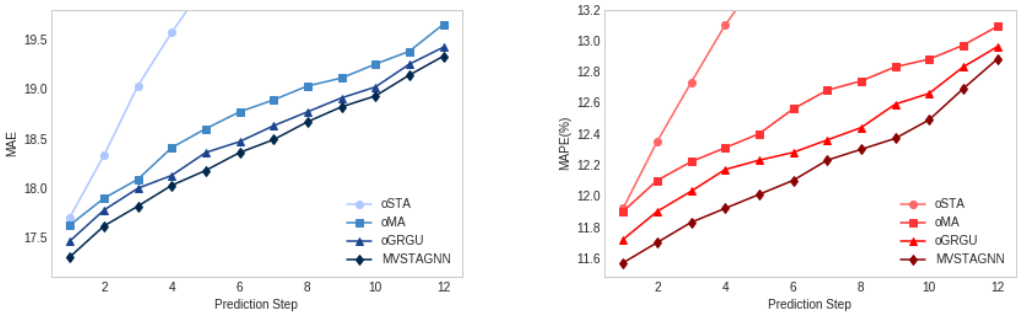


FIGURE 2. Comparison of prediction curves on PEMS04

5.5. **Abaltion experiment.** To verify the effectiveness of individual components in the model, we made the following variants of the model: (1) MSTA-GNN/oSTA: completely remove the spatiotemporal attention mechanism; (2) MSTA-GNN/oMA: remove the multi-head attention mechanism; (3) MSTA-GNN/oGRGU: Remove gated graph convolution units; we performed ablation experiments on the above variants on the PEMS04 dataset. The Figure 3 shows the measurement results of MAE and MAPE. Our model performs better than other variants, which also verifies the effectiveness of each component in our model.



(a) MAE per prediction step (b) MAPE per prediction step

FIGURE 3. Ablation experiment of module effectiveness.

6. **Conclusion.** We propose a novel deep-learning model for traffic flow prediction, named MSTA-GNN, which fully utilizes dynamic Spatio-Temporal similarity generated from historical traffic data without relying on predefined static adjacency matrices. The method could effectively enhance the expression of dynamic correlation attributes between road network nodes. In addition, the model pays attention to the graph attention mechanism and performs graph convolution operations on the dynamic spatio-temporal similarity graph generated by dynamic spatio-temporal similarity, being beneficial to reduce the dependence of the prediction process on prior knowledge.

At the same time, the spatiotemporal attention module is sequentially combined with the spatiotemporal convolution module, and GRGU is used to capture practical relevance under multiple views. As a result, on the four publicly available datasets, our model improves on all metrics compared to recent baseline methods, and on the Pems04 dataset, our model improves by 2.8% compared to state-of-the-art models.

However, our model still lacks the consideration of different fine-grained relationship matrices, so we plan to further explore the construction of relationship matrices in the future and generate relationship matrices from different aspects to capture more types of spatial correlations and further enhance the construction of spatial relationship matrices for road networks.

7. **Acknowledgment.** This work was supported in part by the projects of the National Natural Science Foundation of China (62376059, 41971340), in part by projects of Fujian Provincial Department of Science and Technology (2021Y4019), projects of Fujian Provincial Department of Finance (GY-Z230007), and in part by the project of Fujian Provincial Universities Key Laboratory of Industrial Control and Data Analysis (KF-J21011).

REFERENCES

[1] F. Zhang, T.-Y. Wu, Y. Wang, R. Xiong, G. Ding, P. Mei, and L. Liu, "Application of quantum genetic optimization of lvq neural network in smart city traffic network prediction," *IEEE Access*, vol. 8, pp. 104 555–104 564, 2020.

[2] S. Kumar, A. Damaraju, A. Kumar, S. Kumari, and C.-M. Chen, "Lstm network for transportation mode detection," *Journal of Internet Technology*, vol. 22, no. 4, pp. 891–902, 2021.

[3] X. Wang, Y. Ma, Y. Wang, W. Jin, X. Wang, J. Tang, C. Jia, and J. Yu, "Traffic flow prediction via spatial temporal graph neural network," in *Proceedings of the Web Conference 2020*, 2020, pp. 1082–1092.

[4] Y. Huang, S. Zhang, J. Wen, and X. Chen, "Short-term traffic flow prediction based on graph convolutional network embedded lstm," in *International Conference on Transportation and Development 2020*. American Society of Civil Engineers Reston, VA, 2020, pp. 159–168.

[5] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 922–929.

[6] Z. Chen, Z. Lu, Q. Chen, H. Zhong, Y. Zhang, J. Xue, and C. Wu, "Spatial–temporal short-term traffic flow prediction model based on dynamical-learning graph convolution mechanism," *Information Sciences*, vol. 611, pp. 522–539, 2022.

[7] S. Lan, Y. Ma, W. Huang, W. Wang, H. Yang, and P. Li, "Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting," in *International Conference on Machine Learning*. PMLR, 2022, pp. 11 906–11 917.

[8] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," *arXiv preprint arXiv:1312.6203*, 2013.

[9] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," *Advances in Neural Information Processing Systems*, vol. 29, no. 2, pp. 3844–3852, 2016.

[10] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[11] A. Micheli, "Neural network for graphs: A contextual constructive approach," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 498–511, 2009.

[12] J. Atwood and D. Towsley, "Diffusion-convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 29, no. 4, pp. 1993–2003, 2016.

[13] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[14] P. Wang, T. Zhang, Y. Zheng, and T. Hu, "A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation," *International Journal of Geographical Information Science*, vol. 36, no. 6, pp. 1231–1257, 2022.

[15] G. Jin, Z. Xi, H. Sha, Y. Feng, and J. Huang, "Deep multi-view spatiotemporal virtual graph neural network for significant citywide ride-hailing demand prediction," *arXiv preprint arXiv:2007.15189*, 2020.

[16] H. Li, D. Jin, X. Li, H. Huang, J. Yun, and L. Huang, "Multi-view spatial–temporal graph neural network for traffic prediction," *The Computer Journal*, vol. 66, no. 10, pp. 2393–2408, 2023.

[17] Z. Liu, J. Bian, D. Zhang, Y. Chen, G. Shen, and X. Kong, "Dynamic multi-view coupled graph convolution network for urban travel demand forecasting," *Electronics*, vol. 11, no. 16, p. 2620, 2022.

[18] Q. Zhou, J. Gu, X. Lu, F. Zhuang, Y. Zhao, Q. Wang, and X. Zhang, "Modeling heterogeneous relations across multiple modes for potential crowd flow prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 5, 2021, pp. 4723–4731.

[19] C. Li, L. Bai, W. Liu, L. Yao, and S. T. Waller, "A multi-task memory network with knowledge adaptation for multimodal demand forecasting," *Transportation Research Part C: Emerging Technologies*, vol. 131, p. 103352, 2021.

[20] X. Huang, Y. Ye, C. Wang, X. Yang, and L. Xiong, "A multi-mode traffic flow prediction method with clustering based attention convolution lstm," *Applied Intelligence*, pp. 1–14, 2021.

[21] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," *arXiv preprint arXiv:1906.00121*, 2019.

[22] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 804–17 815, 2020.

[23] C. Park, C. Lee, H. Bahng, K. Kim, S. Jin, S. Ko, J. Choo *et al.*, "Stgrat: A spatio-temporal graph attention network for traffic forecasting," *arXiv preprint arXiv:1911.13181*, vol. 26, 2019.

[24] M. Li and Z. Zhu, "Spatial-temporal fusion graph neural networks for traffic flow forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 5, 2021, pp. 4189–4196.

[25] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 914–921.

[26] Z. Fang, Q. Long, G. Song, and K. Xie, "Spatial-temporal graph ode networks for traffic flow forecasting," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 364–373.

[27] Y. Ma, Y. Peng, and T.-Y. Wu, "Transfer learning model for false positive reduction in lymph node detection via sparse coding and deep learning," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2121–2133, 2022.

[28] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," *arXiv preprint arXiv:1709.04875*, 2017.
[29] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in Neural Information Processing Systems*, vol. 27, no. 4, pp. 3104–3112, 2014.
[30] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.
[31] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.