Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 1358535, 13 pages
https://doi.org/10.1155/2022/1358535

Research Article

Multi-Dimensional Attention Based Spatial-Temporal Networks for Traffic Forecasting

Guangxia Xu [1,2] **and Xinting Hu** [2]

[1] *Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou 510006, China*
[2] *School of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*

Correspondence should be addressed to Guangxia Xu; xugx@cqupt.edu.cn

Received 26 April 2022; Revised 30 June 2022; Accepted 13 July 2022; Published 1 September 2022

Academic Editor: Lei Liu

Traffic flow prediction is the key problem of intelligent transportation system. Accurate prediction results are indispensable for traffic management and road planning. However, due to the complex spatial-temporal correlation of traffic flow data, including the spatial correlation and temporal correlation of adjacency, periodicity, and trend that exist between different roads. The existing forecasting methods consider the spatial-temporal correlation but lack the dynamic modeling of spatial-temporal correlation. To deal with this dynamic feature, this paper proposes a multi-dimensional attention-based spatial-temporal network (MA-STN). It mainly contains three parts, the spatial-temporal attention unit, the spatial-temporal feature extraction unit based on Graph Convolutional Network (GCN) and the fusion prediction unit, and the residual connection is also added to the model to avoid the gradient disappearance problem. Meanwhile, this paper divides the dataset into three subsets to deal with the three features in the temporal dimension separately. To verify the effectiveness of the proposed model, two real-world road traffic flow data collected by PeMS system are used for validation. By comparing six different models, the proposed network in this paper has a 7% accuracy improvement compared to the baseline model. To verify the effectiveness of the attention mechanism, ablation experiments are used in this paper for validation, and the results show that the attention mechanism can achieve a 5% accuracy improvement.

1. Introduction

With the progress of urbanization in recent years, traffic problems have become increasingly serious. The intelligent transportation system is a feasible solution for real-time traffic control, real-time scheduling, and abnormal monitoring [1], but its core cannot be separated from the real-time prediction of traffic flow [2–5]. Traffic flow data mainly contain three parameters, flow, density and speed [6], which are important indicators of traffic operation characteristics, and if traffic flow condition of the road can be accurately predicted in advance, traffic management departments can be guided in a timely and reasonable manner.

The road flow prediction problem is a typical spatial-temporal data prediction problem [7], and the difficulty lies in how to extract the spatial features of roads with features in the temporal dimension. Spatial characteristics specifically, i.e., road flows are correlated between upstream and downstream flows of the same road and correlation exists between adjacent roads. And temporal features react in proximity, trend and periodicity [8], as shown in Figure 1. In the flow variation within a day, the flow at adjacent times usually shows a trend, and in the graph of the two-day flow variation, it can be seen that in the flow variation has a cyclical character, and in the graph of the flow variation of a week and a month, there is also a cyclical character. It is also clear from the graph that the pattern of flow variation is not identical between weekdays and rest days [9].

With the in-depth research in the field of traffic facilities and traffic engineering, as well as the rapid growth of traffic flow data due to economic development, there is data support to analyse the change pattern of traffic flow data. These

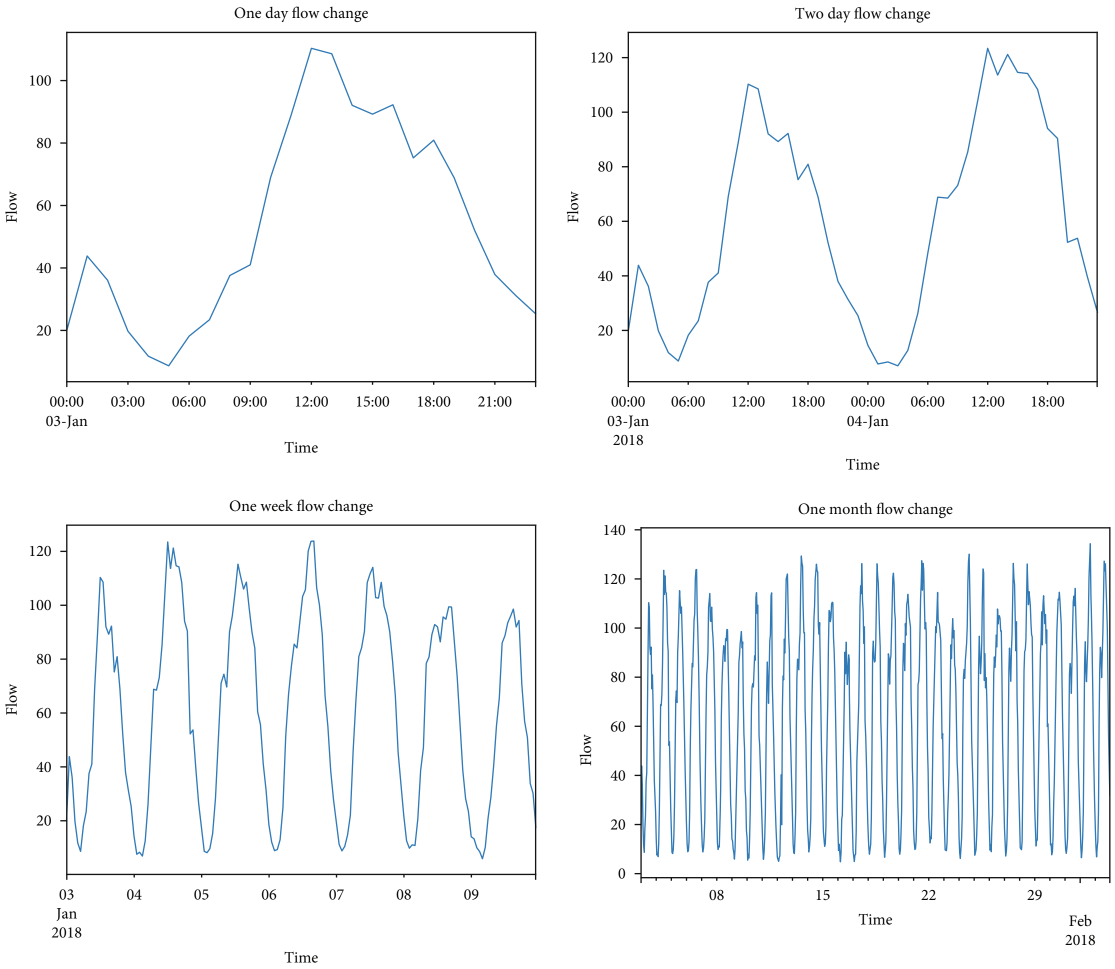


Figure 1: Temporal characteristics on traffic data.

data are usually transmitted by fixed collectors out and therefore have a strong spatial correlation. Early researchers mainly used semiotic methods to model traffic problems by mining using time series analysis models, ARIMA models [10], etc. These models are difficult to do a good fit for time series that are nonlinear and non-stationary. With the development of machine learning, researchers started to use machine learning algorithms for modeling traffic data, which have better performance on some data but rely on experienced people for feature engineering of the data and have poor portability [11]. In recent years, with the superior performance of representation learning methods in the image domain and natural language domain, researchers have turned their attention to representation learning methods. This approach learns representations in spatial-temporal data, using deep recurrent networks to obtain temporal representations and convolutional networks to obtain spatial representations. It significantly improves the prediction results, while eliminating the need for complex feature engineering since it relies only on the data itself.

After years of research and practice, traffic forecasting has experienced development from statistical-based models, traditional machine learning-based to deep learning. Traditional methods based on statistical models, including HA, ARIMA [10], VAR [12], etc., while ARIMA maintains better results after operations such as differencing and taking logarithms, and has been developed to produce seasonal ARIMA models, which are good at extracting temporal characteristics. However, these models have a large dependence on data and require complex processing to ensure the accuracy of the prediction effect, which cannot be guaranteed for the complex traffic conditions in real situations. Subsequently, with the development of machine learning, the models are able to handle more complex data, and the commonly used methods include KNN [13] and SVR [14]. SVR, as an extension of SVM for regression problems, is able to ensure better prediction results by adjusting its kernel function. However, the machine learning approach relies on feature engineering for data processing, which requires the researcher to have a good understanding of the model inputs and the features needed for the model.

Deep learning started with the gradual development of stacking multi-layer perceptron, and then became a research method with wide interest due to the excellent performance of convolutional neural networks in the field of image recognition and with the increasing computing power of computers. For spatial-temporal data, the use of RNN and its variant GRU [15] or LSTM [16] became reliable, and in

[16] LSTM was used for cab demand prediction, which captures the dynamics in the time dimension. This is mainly due to the ability of LSTM to model the time dimension and extract the effective features from it to discard the features that interfere with the prediction results, but this approach ignores the features in the spatial dimension. Subsequently, [17] proposed to use CNN to extract features in spatial dimension and LSTM to extract features in temporal dimension for prediction after feature fusion, this method captures features in spatial dimension and prediction has better results. [18] obtained excellent performance on Beijing cab data by introducing a residual mechanism while using convolutional layers to obtain spatial dependencies and mining and model temporal correlations of proximity, trend and periodicity. However, these researchers used network-based traffic flow data, which cannot extract the spatial correlation in traffic maps well.

Convolution operation is the basis of convolutional neural network, and traditional convolutional operations are all for data in natural Euclidean space like images, which have translation invariance, that is, when window sliding is performed during convolutional operation, it can ensure that the dot product between convolution kernel and data is a fixed size tensor. However, the traffic network is a non-Euclidean structure, and it needs to transform the data into a raster type before convolution, which results in the situation that there are no roads with null values in the raster, resulting in a spatial structure of the traffic network that is not easily extracted by the deep learning framework for features. Therefore, X. Geng et al. proposed the operation of graph convolution [19], which transforms the traffic network into a matrix that can be easily learned by borrowing the ordinary convolution operation while introducing the spectral transform, and this method proposes a feasible solution for non-Euclidean spatial data. Thereafter, the introduction of Chebyshev polynomials in [20] reduces the time complexity of graph convolution substantially and avoids the problems of training time and memory occupation. Y, Li et al. will propose a diffusion graph convolution operation by borrowing the concept of random walk in propagation [21], this way convolution filters the graph nodes and their neighboring nodes and controls the learning by setting the number of hops between neighboring nodes depth of spatial features, this approach avoids the problem of constructing Laplace matrices when the spectrum changes and also enables learning the spatial structure in the network. A graph convolutional network for traffic prediction based on this method was proposed in [22], but the model did not consider the dynamic spatial-temporal correlation of traffic data.

In recent years, attention mechanisms have been widely used in various tasks such as natural language processing, image captioning, and speech recognition. The goal of attention mechanisms is to select the information that is relatively critical to the current task from all inputs. xu, K et al. proposed two attention mechanisms in an image description task [23] and used visualization to visualize the effect of the attention mechanisms. In the traffic flow prediction task, J. Wang et al. used multiple attention mechanisms with graph convolutional networks to process traffic flow data [24] and obtain spatial-temporal features, and experiments showed that this approach could significantly improve model prediction. In order to predict time series, S. Guo et al. proposed a multilevel attention network that adaptively adjusts the correlation of time series from multiple geographic sensors [25]. However, this is time-consuming in practice because separate models need to be trained for each time series.

Motivated by the aforementioned research, we use both graph convolution and attention mechanisms to model network-structured traffic data, considering the graph structure of traffic networks and the dynamic spatial-temporal patterns of traffic data.

The proposed graph convolutional neural network has led to a significant improvement in traffic prediction, which can learn spatial features from the natural graph structure of traffic road networks. However, it is still worth exploring how to capture the dynamic spatial-temporal correlation of traffic flow data in a deep network architecture with prediction models that are adaptable to different time periods. In this paper, we propose a new deep learning framework, the multidimensional attentional spatial-temporal network MA-STN, for capturing spatial-temporal dependencies in different time dimensions separately. The model can be processed directly on graph-based traffic data and can effectively capture spatial-temporal features. The main contributions of this paper are summarized as follows:

(i) In this paper, a multidimensional attentional spatial-temporal network is designed to learn dynamic spatial-temporal features in traffic data. Specifically, the spatial features between different roads are captured using graph convolution, and the dynamic temporal features between different times are captured using spatial-temporal attention mechanism with long-short term memory neural network

(ii) A module that captures spatial-temporal features is designed for obtaining spatial-temporal correlations on traffic data in certain dimensions. It consists of a graph convolution based on the structure of the traffic network capturing spatial features and a convolution to obtain adjacent time slices with periodic time slice dependence

(iii) By conducting a large number of experiments on real data, it has been shown that the model proposed in this paper has better prediction results compared with existing research methods

2. Proposed MA-STN Deep Learning Framework

2.1. Transportation Networks. In the field of transportation, the traffic road network is usually abstracted as an undirected graph $G = (V, E, A)$, Where V represents the set of nodes usually sensors or road intersections, and $\mathrm{N} = |\mathrm{V}|$

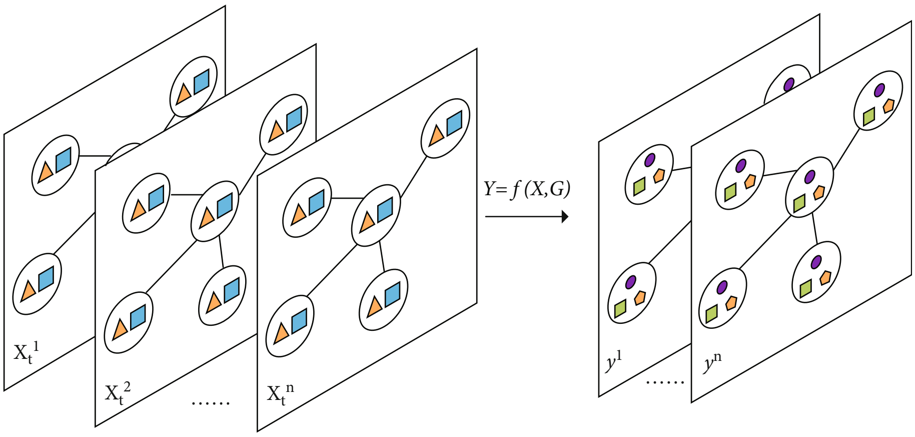


Figure 2: Traffic Forecasting Tasks.

represents the number of nodes. E represents relationships between nodes, for traffic networks it means the road path or the relationship between sensors. $\mathrm{A} \in \mathbb{R}^{N\times N}$ represents adjacency matrix.

$$A_{i,j} = \begin{cases} 1\,, i \neq j \,\text{and}\, d_{ij} \geq \varepsilon \\ \\ 0, i = j \,\text{or}\, d_{ij} < \varepsilon \end{cases} \tag{1}$$

2.2. Traffic Forecasting Tasks. Since sensors usually use the same frequency for data acquisition. The amount of data is denoted by F. For node i^{th} in time c^{th} the features represent $x_t^{c,t} \in \mathbb{R}$, and all the features are including $X_t = (x_t^1, x_t^2, \cdots, x_t^N)^T \in \mathbb{R}^{N\times F}$. For traffic forecasting tasks, the features tensor represents by $\mathscr{X}_t = (\mathrm{X}_\mathrm{t}^1, \mathrm{X}_t^2, \cdots, \mathrm{X}_\mathrm{t}^N) \in \mathbb{R}^{N\times F\times \tau}$. To obtain future traffic flow conditions is the main goal of traffic forecasting, which means using historical traffic data to predict future traffic conditions, as shown in Figure 2.

Given the measured historical data $\mathscr{X}$ on τ time slices, to predict the traffic sequence $Y = (y^1, y^2, \cdots, y^N)^T \in \mathbb{R}^{\mathrm{N}\times\mathrm{T}_p}$ for all nodes on T_p time slices in the future, where $y^i = (y^i_{\tau+1}, y^i_{\tau+2}, \cdots, y^i_{\tau+T_p}) \in \mathbb{R}^{\mathrm{T_p}}$ represents the traffic sequence of nodes starting from the moment τ.

Figure 3 illustrates the overall framework of the MASTN proposed in this paper, which consists of three main components, namely, modules for handling recent, daily-period and weekly-period dependencies in historical data, which are formed using similar and independent compositions. All of them are eventually imported into the fusion unit and the loss function is used to adjust the model parameters to achieve better prediction results.

We divide the time as three periodic, it is necessary to determine the size of the forecast time series as T_p. In order to obtain the time series input of the latest, daily period and weekly period, the historical traffic data is divided. Among them, T_h, T_d, and T_w are set to represent the time series of the most recent, daily period, and weekly period, respectively, and they are set as integer multiples of the forecast time series T_p. Specifically, the criteria for classification are as follows:

Recent time series: This part is directly connected to the expected time series. According to Figure 1, it can be seen that the traffic flow is gradually increasing and decreasing. Therefore, the traffic flow just past will inevitably have an impact on the future traffic flow, so the input features of the recent time series can be expressed as $\chi_h = (X_{t_0-T_h+1}, X_{t_0-T_h+2}, \cdots, X_{t_0}) \in \mathbb{R}^{N\times F\times T_h}$.

Daily-periodic time series: It can be seen from Figure 1 that the past few days and the forecast period have similar fluctuations, and due to the regularity of people's daily life. Traffic data may show similar fluctuations, and morning and evening peaks are a more prominent feature. The daily cycle time series is to simulate the daily cycle characteristics of traffic data and obtain these characteristics. It can be expressed as $\chi_d = (X_{t_0-(T_d/T_p)*q+1}, \cdots, X_{t_0-(T_d/T_p-1)*q+T_p}, \cdots, X_{t_0-(T_d/T_p-1)*q+1}, \cdots, X_{t_0-q+T_p}) \in \mathbb{R}^{N\times F\times T_d}$.

Weekly-periodic time series: It can be seen from Figure 1 that the traffic patterns in the past few weeks are similar. Generally speaking, working days will show similar ups and downs, but there is a big difference between working days and rest days. Therefore, it is necessary to capture these characteristics through the weekly cycle time series, which can be expressed as $\chi_w = X_{t_0-7*(\mathrm{T}_w/T_p)*q+1}, \cdots, X_{t_0-7*(\mathrm{T}_w/T_p)*q+T_p}, X_{t_0-7*(\mathrm{T}_w/T_p-1)*q+1} \cdots, X_{t_0-7*(T_w/T_p-1)*q+T_p}, \cdots, X_{t_0-7*q+1}, \cdots, X_{t_0-7*q+T_p} \in \mathbb{R}^{N\times F\times T_w}$.

These three parts are used for the same components, which contain units for dealing with time dependence and space dependence. And each spatial-temporal dependency processing unit includes a spatiotemporal attention module and a spatiotemporal convolution module. In order to avoid the problem of the disappearance of the gradient during the training process, a jump connection mechanism is introduced, that is, the residual connection, and related papers also prove the effect of the residual connection on the traffic prediction task. Finally, the outputs of the three components are fused, and the final prediction result is obtained. The entire model models the temporal and spatial dependence

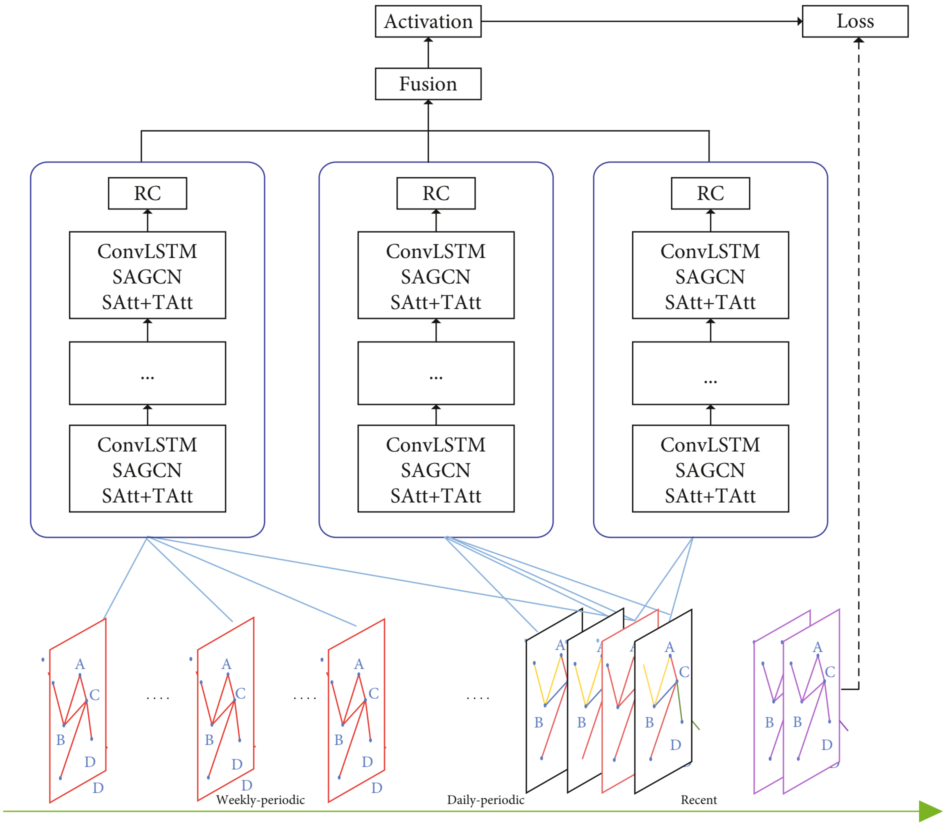


Figure 3: MA-STN architecture. SAGCN: Spatial attention Graph Convolution Networks; RC: Residual Convolution.

of traffic flow forecasting, and can handle the complex variability in historical flow data.

2.3. Spatial-Temporal Attention Mechanism. In this paper, the attention mechanism is used to capture the features in the temporal and spatial dimensions of traffic flow data, which consists of two attention mechanisms, namely, temporal attention and spatial attention.

Temporal attention: Traffic flow conditions are affected by various aspects due to their temporal dimension, including historical traffic data, special holidays such as holidays, and seasonal changes, and the features that can be obtained from the traffic data should be different for different situations. For this reason, an attention mechanism is needed to process the features, and the approach in [26] is used:

$$E = V_e \cdot \sigma\left(\left(\left(\chi_h^{r-1}\right)^T U_1\right) U_2 \left(U_3 \chi_h^{\gamma-1}\right) + b_e\right)$$
$$E'_{i,j} = \frac{\exp\left(E_{i,j}\right)}{\sum_{j=1}^{T_{r-1}} \exp\left(E_{i,j}\right)} \tag{2}$$

where the parameters include $V_e, b_e \in \mathbb{R}^{\mathrm{T_{r-1}} \times \mathrm{T_{r-1}}}$, $U_1 \in \mathbb{R}^{\mathrm{N}}$, $U_2 \in \mathbb{R}^{\mathrm{C_{r-1}} \times N}$, $U_3 \in \mathbb{R}^{\mathrm{C_{r-1}}}$, all of parameters are trainable. And $\chi_h^{r-1} = (X_1, X_2, \cdots, X_{T_{r-1}}) \in \mathbb{R}^{\mathrm{N} \times \mathrm{C_{r-1}} \times T_{r-1}}$ is the input of the r^{th} layer, C_{r-1} represents the number of channels of the input data, when $r = 1$, $C_0 = F$, i.e., the number of nodes, T_{r-1} is the input of the time dimension size, when $r = 1$, $T_0 = T_h$. (here the proximity module is used as an example, for different inputs T_0 represents different, in the daily cycle module $T_0 = T_d$, in the weekly cycle module $T_0 = T_W$). The training results are output as a matrix, and as input features those are assigned larger weights. Each element of the matrix $E_{i,j}$ represents the strength of the dependency between node i and node j. Finally, the elements in the matrix are guaranteed to be between [0,1] by the *sigmoid* function, i.e., σ. $E'_{i,j}$ is the result of processing by the softmax function, and the feature vector is processed directly using E' as the weight matrix, and the values of the matrix are updated in training to ensure that the temporal features in the historical data can be obtained, i.e., $\widehat{\chi}_h^{r-1} = (\widehat{X}_1, \widehat{X}_2, \cdots, \widehat{X}_{T_{r-1}}) = (X_1, X_2, \cdots, X_{T_{r-1}})E' \in \mathbb{R}^{N \times C_{r-1} \times T_{r-1}}$.

Spatial attention: Traffic conditions at different locations are subject to mutual influence, especially on adjacent roads, and this relationship is usually highly dynamically correlated, as can be seen in Figure 4, where the influence between roads is dynamic and proximity. Therefore, to obtain these features a spatial attention mechanism is used, which can be expressed as:

$$S = V_s \cdot \sigma\left(\left(\chi_h^{r-1} W_1\right) W_2 \left(W_3 \chi_h^{\gamma-1}\right)^T + b_s\right)$$
$$S'_{i,j} = \frac{\exp\left(S_{i,j}\right)}{\sum_{j=1}^{T_{r-1}} \exp\left(S_{i,j}\right)} \tag{3}$$

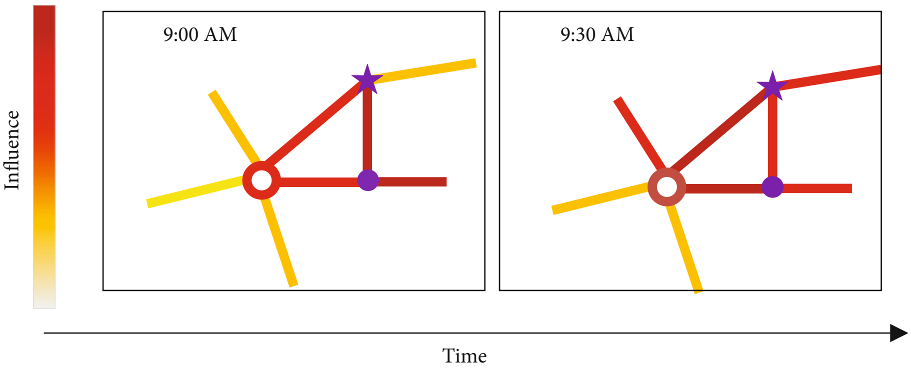


Figure 4: Spatial influence of traffic flow at different times.

which contains parameters $V_s, b_s \in \mathbb{R}^{N\times N}$, $W_1 \in \mathbb{R}^{\mathrm{T}_{\mathrm{r}-1}}$, $W_2 \in \mathbb{R}^{\mathrm{C}_{\mathrm{r}-1}\times T_{r-1}}$, $W_3 \in \mathbb{R}^{\mathrm{C}_{\mathrm{r}-1}}$, which are learnable parameters and σ is the *Sigmoid* activation function. And $\chi_h^{r-1} = (X_1, X_2, \cdots, X_{T_{r-1}}) \in \mathbb{R}^{\mathrm{N}\times\mathrm{C}_{\mathrm{r}-1}\times T_{r-1}}$ is the input of the r^{th} layer, C_{r-1} represents the number of channels of the input data, when $r = 1$, $C_0 = F$, i.e., the number of nodes, T_{r-1} is the input of time dimension size, when $r = 1$, $T_0 = T_h$ (in the recent module). Similarly, the spatial attention mechanism generates a parameter matrix for assigning different weights. $S_{i,j}$ denotes the strength of spatial correlation between nodes i and j, and the result is guaranteed to be between [0,1] by $softmax$. The spatial attention matrix S updates the internal parameters as the network is trained, and when model training is performed, the weights of node messaging are dynamically adjusted with the temporal attention matrix E as well as the proximity matrix A.

2.4. Spatial-Temporal Feature Extract. For the input data, a spatial-temporal convolution module is used to extract spatial-temporal features. Two types of convolution modules are included, convolution in spatial dimension and convolution in temporal dimension, spatial dimension is used to capture spatial features by graph convolutional neural network and temporal dimension is used to extract temporal features by ConvLSTM.

Spatial feature extract: In this paper, traffic networks are considered as graph structures, and convolutional neural networks are powerful tools used in the image domain to deal with spatial dependencies, but it is difficult to perform convolution on the graph so as to perform feature aggregation and processing, and the difficulty is that the graph does not have translation invariance, so it is impossible to perform convolutional operations using the same convolutional kernel. Shuman et al. proposed that the convolutional operation on the graph process [27], which can be seen as the process of information transfer of node signals on the graph, so the theory of spectral domain transformation is used to convert the features of grid data to graph structure data. Therefore, in order to make full use of the topological features of the traffic network, for each traffic flow data collected is considered as a graph with node features, and the spectral method is used to convert the feature graph into algebraic form to obtain the spatial features on the graph.

In the spectral change-based approach, the graph is transformed into a Laplace matrix defined as $L = D - A$, where A represents the adjacency matrix and $D \in \mathbb{R}^{N\times N}$ is the degree matrix as well as a diagonal matrix, where $D_{ii} = \sum_j A_{ij}$, which represents the spatial structural features of the graph. To avoid the effect of the magnitude, after normalizing it $L = I_N - D^{-1/2} A D^{-1/2} \in \mathbb{R}^{N\times N}$, where I_N is the unit array. The Laplace matrix is a symmetric matrix for which the matrix decomposition can be expressed as $L = U \Lambda U^T$, where U is an orthogonal matrix or called Fourier basis and $\Lambda = \mathrm{diag}\left([\lambda_0, \cdots, \lambda_{N-1}]\right) \in \mathbb{R}^{\mathrm{N}\times\mathrm{N}}$ is a diagonal matrix. For the traffic flow at moment t, the signal on the traffic graph can be expressed as $x = x_t^f \in \mathbb{R}^N$ and the graph Fourier transform can be defined as $\hat{x} = U^T x$, whose inverse variation can be expressed as $x = U\hat{x}$. The graph convolution is a linear algorithm that uses the Fourier change to diagonalize the convolution operator instead of the convolution operator [28], and according to these derivations, the signal on the graph G can be represented by the convolution kernel g_θ as:

$$g_\theta *_G x = g_\theta(L)x = \mathrm{U} g_\theta(\Lambda)\mathrm{U}^{\mathrm{T}} x \tag{4}$$

Where $*_G$ stands for convolution operation, for the convolution operation of the signal on the graph that is for the Fourier change of the graph, the essence is the convergence and product of the graph signal, so the above formula can be understood as using the convolution operation to transform the convolution kernel g_θ and the signal x on the graph into the spectral domain, respectively, and multiply the results and perform the Fourier inverse transform to get the final result. However, when the graph is more complex, the time complexity of the eigen decomposition of the Laplacian matrix is exponential. Therefore, M. Simonovsky, et al. proposed that Chebyshev polynomials can solve this problem by approximation [20].

$$g_\theta *_G x = g_\theta(L)x = \sum_{K=0}^{K-1} \theta_k T_k(\tilde{L})x \tag{5}$$

where the parameter θ_k represents the coefficients of the polynomial, and the Chebyshev polynomial is defined as

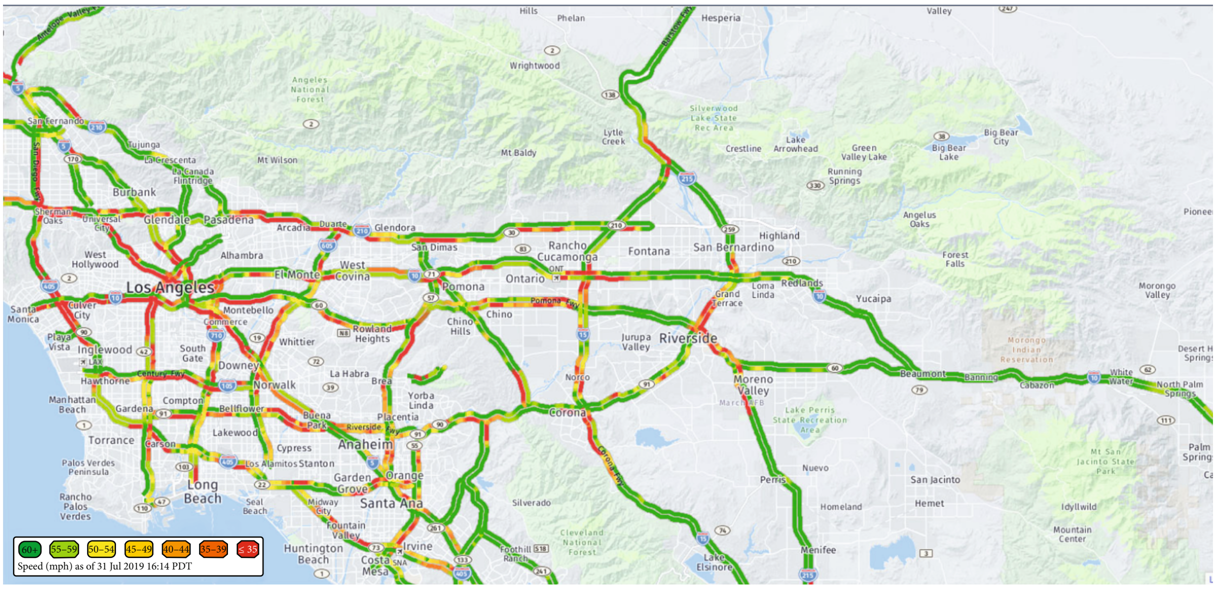


Figure 5: Road Route Map.

$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$, where $T_0(x) = 1$, $T_1(x) = x$, and $\tilde{L} = (2/\lambda_{\max})L - I_N$, and $\lambda_{\max}$ is the maximum eigenvalue of the Laplace matrix. Chebyshev polynomials for approximation calculations are able to fit the convolution kernel g_θ and perform the process of convolution operations and obtain information from the center of each node to the surrounding 0 to K-1 hop neighbor nodes. After the graph convolution module using the *Relu* function as the activation function, the graph convolution can be expressed as $Relu(g_\theta *_G x)$.

In order to obtain the dynamic changes of the traffic graph, such as the occurrence of traffic accidents, traffic construction, etc., the attention mechanism is also used to focus on the dynamic changes of node traffic when performing graph convolution, i.e., for each convolution operation the attention matrix S' is added, and the graph convolution formula is defined as $g_\theta *_G x = \sum_{K=0}^{K-1} \theta_k T_k(\tilde{L} \odot S')x$, where $\odot$ represents the Hadamard product.

Temporal feature extraction module: After the graph convolution each node on the time slice acquires its own features with the neighboring nodes, and then these features are converged in time dimension by convolution operation, the temporal feature extraction module can be expressed as:

$$\begin{gathered}
\mathrm{i_t} = \mathrm{Sigmoid}(\mathrm{Conv}(\mathrm{x_t}\,;\mathrm{w_{xi}}) + \mathrm{Conv}(\mathrm{h_{t-1}}\,;\mathrm{w_{hi}}) + \mathrm{b_i}) \\
\mathrm{f_t} = \mathrm{Sigmoid}(\mathrm{Conv}(\mathrm{x_t}\,;\mathrm{w_{xf}}) + \mathrm{Conv}(\mathrm{h_{t-1}}\,;\mathrm{w_{hf}}) + \mathrm{b_f}) \\
\mathrm{o_t} = \mathrm{Sigmoid}(\mathrm{Conv}(\mathrm{x_t}\,;\mathrm{w_{xo}}) + \mathrm{Conv}(\mathrm{h_{t-1}}\,;\mathrm{w_{ho}}) + \mathrm{b_o}) \\
\mathrm{g_t} = \mathrm{Tanh}\left(\mathrm{Conv}\left(\mathrm{x_t}\,;\mathrm{w_{xg}}\right) + \mathrm{Conv}\left(\mathrm{h_{t-1}}\,;\mathrm{w_{hg}}\right) + \mathrm{b_g}\right) \\
\mathrm{c_t} = \mathrm{f_t} \odot \mathrm{c_{t-1}} + \mathrm{i_t} \odot \mathrm{g_t} \\
\mathrm{h_t} = \mathrm{o_t} \odot \mathrm{Tanh}(\mathrm{c_t})
\end{gathered} \tag{6}$$

Where $*$ denotes the standard convolution operation, Φ is the convolution kernel in the time dimension, and the activation function also adopts the ReLU function.

After the spatial-temporal convolution module, both temporal and spatial features in the traffic data can be extracted effectively, and long-range spatial-temporal correlation can be obtained in a larger scale by stacking multiple spatial-temporal convolution modules. Full connection layer is used to ensure that the output have the same dimension and shape as the target. And we use *ReLU* function as the activation function.

2.5. Multi-Dimensional Fusion Unit. After feature extraction in the three modules, feature fusion is needed for these outputs. For traffic flows at different moments may have different characteristics, for example, for traffic flows in the traffic network at 5 pm on Monday, the impact of the nearest moment time series will be less than the output of the weekly cycle time series versus the weekly cycle time series. However, the traffic flow at certain moments does not have this traffic cycle pattern, so the output of daily cycle and weekly cycle has less impact. Therefore, different weights need to be applied to the features of different modules for fusion.

$$\hat{Y} = W_h \odot \hat{Y}_h + W_d \odot \hat{Y}_d + W_w \odot \hat{Y}_w \tag{7}$$

Where $\odot$ is the Hadamard product, W_h, W_d, W_w are learning parameters that reflect the degree of influence of the three modules on the prediction results.

3. Results and Discussion

In order to evaluate the performance of the model, two real traffic datasets are used for comparison.

3.1. Datasets. The datasets come from two California highway traffic datasets, PeMSD4 and PeMSD8, representing two different areas as in Figure 5, collected from the Performance Measurement System PeMS [29], which is collected at a frequency of once every 30 seconds. The system deployed 39,000 detectors on California freeways, with geographic information between sensors in the raw data set. Three main types of traffic measurements were used in the laboratory, namely road flow, average speed and average occupancy.

PeMS4 refers to traffic data for the San Francisco Bay Area, which contains 29 roadways with 3,848 detectors. The data were collected over a period of time spanning from January to February 2019. Fifty days of these data were selected as the test set and the others as the validation set.

PeMS8 refers to traffic data for the San Bernardino area, which contains 8 roads with 1,979 detectors. The data collection spans from July 2019 to August 2019, with the first 50 days of data as the validation set and the others as the test set.

To avoid data redundancy, the nodes with excessive node distances are removed and the missing values are filled by linear interpolation. Also to avoid the effect of magnitude normalization is used to transform the feature vector to a form with mean zero.

3.2. Model Settings and Baselines. The model is built by Pytorch, and the K value of Chebyshev polynomial is set to 3 according to [30], and 64 convolution kernels are set for the graph convolution layer, and 64 convolution kernels are also set for the time convolution layer, and the time length of the input is adjusted by controlling the step size of the time convolution. For the three modules, $T_h = 12$, $T_d = 6$, $T_w = 3$, and the time window of prediction is set to $T_p = 12$, which means the traffic flow condition in one hour is predicted. The mean square error between the true value and the predicted value is used as the loss function, and the parameters in the model are optimized by back propagation. For the training phase, the batch block size is set to 64, and to avoid the effect of small data values, the Adam optimizer is used and the learning rate is set to 0.0001. In order to verify the effect of the attention mechanism on the dynamics of the graphs in the paper, a comparison model is also set, the multidimensional convolutional residual network MSTN, which only eliminates the attention mechanism other than no difference with this model.

This model compares the following models: HA, Historical average model. The model is based on a statistical approach that uses the traffic flow of the first 12 time slices of the forecast to predict the traffic flow of the next time slice; ARIMA [10], Autoregressive Integrated Moving Average model. This model is mainly used for time series forecasting, by using difference operations to transform the time series into a smooth time series and forecasting by autoregression, where the moving average method can effectively eliminate the effect of noise in forecasting on the results; GRU [15], Gated Recurrent Unit neural network. This model is used to avoid gradient disappearance and gradient explosion of recurrent neural networks by adding gating units; LSTM [16], Long-Short Term memory neural network, a variant of convolutional neural network, capable of obtaining temporal features in long time sequences; Graph-WaveNet [31], A Spatio-Temporal Graph model constructed by introducing adaptive matrices with extended convolution, capable of handling long time series; STGCN [5], A Graph Convolutional neural network model for processing spatial-temporal features, with modules for processing temporal and spatial features.

Table 1: Average Performance Of Difference Methods On PeMSD4 And PeMSD8 Datasets.

Model	PeMSD4		PeMSD8	
	RMSE	MAE	RMSE	MAE
HA	46.08	31.53	38.00	26.05
ARIMA	47.56	29.94	35.53	28.66
GRU	39.69	28.58	29.67	23.59
LSTM	39.3	27.83	27.83	22.31
Graph-WaveNet	37.79	26.52	26.34	21.39
STGCN	38.29	25.62	27.87	20.31
MASTN(propose)	**35.01**	**22.53**	**22.43**	**19.33**
MSTN(propose)	**35.77**	**23.72**	**26.76**	**19.78**

The comparison uses root mean square error and mean absolute error as evaluation indexes, where:

(1) Root Mean Squared Error (RMSE)

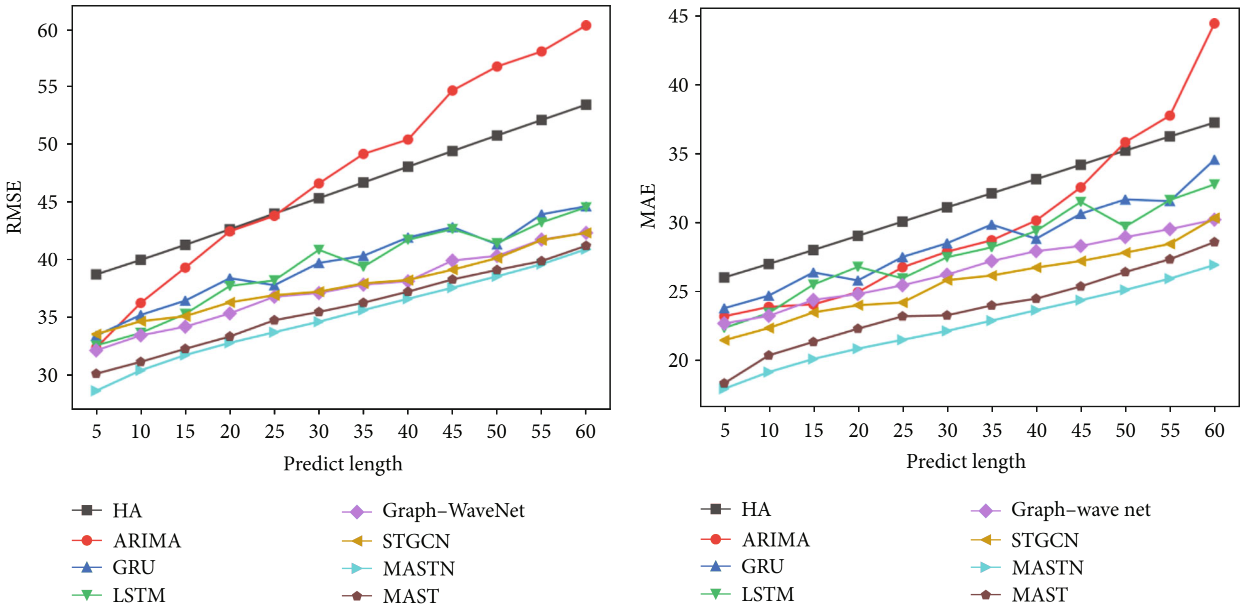
$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{8}$$

(2) Mean Absolute Error (MAE)

$$MAE = \frac{1}{N} \tag{9}$$

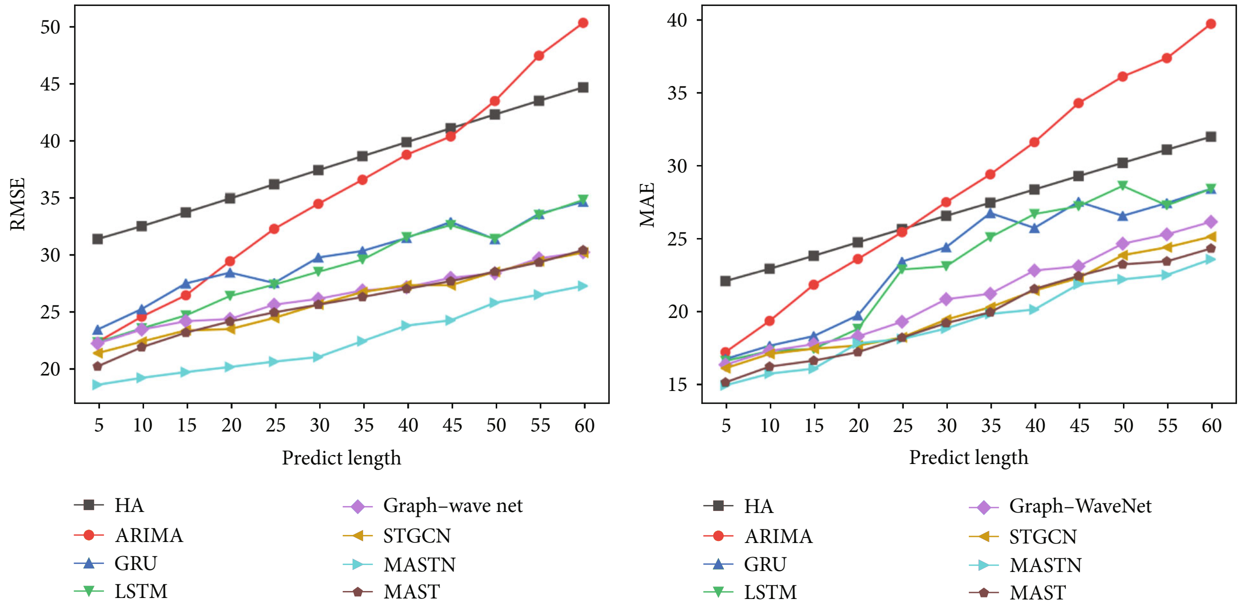
3.3. Experimental Results and Analysis. By comparing the experiments of the model proposed in this paper with the five baseline methods on the PeMSD4 and PeMSD8 datasets, Table 1 can be derived, representing the prediction results for up to one hour in the future. From Table 1, it can be seen that the method proposed in this paper obtains the best results on both datasets. The traditional method, on the other hand, is less effective for the predicted results, which indicates the difficulty of the traditional method to deal with the complex dependencies in a large amount of data.

In contrast, the deep learning-based methods perform better than the traditional methods with machine learning. Among them, STGCN, as a model that considers both spatial-temporal correlation, will perform better than a deep learning model like LSTM that considers only temporal correlation. For the proposed two models, the model with the



(a) (b)

Figure 6: The prediction results of different methods on PEMSD4.

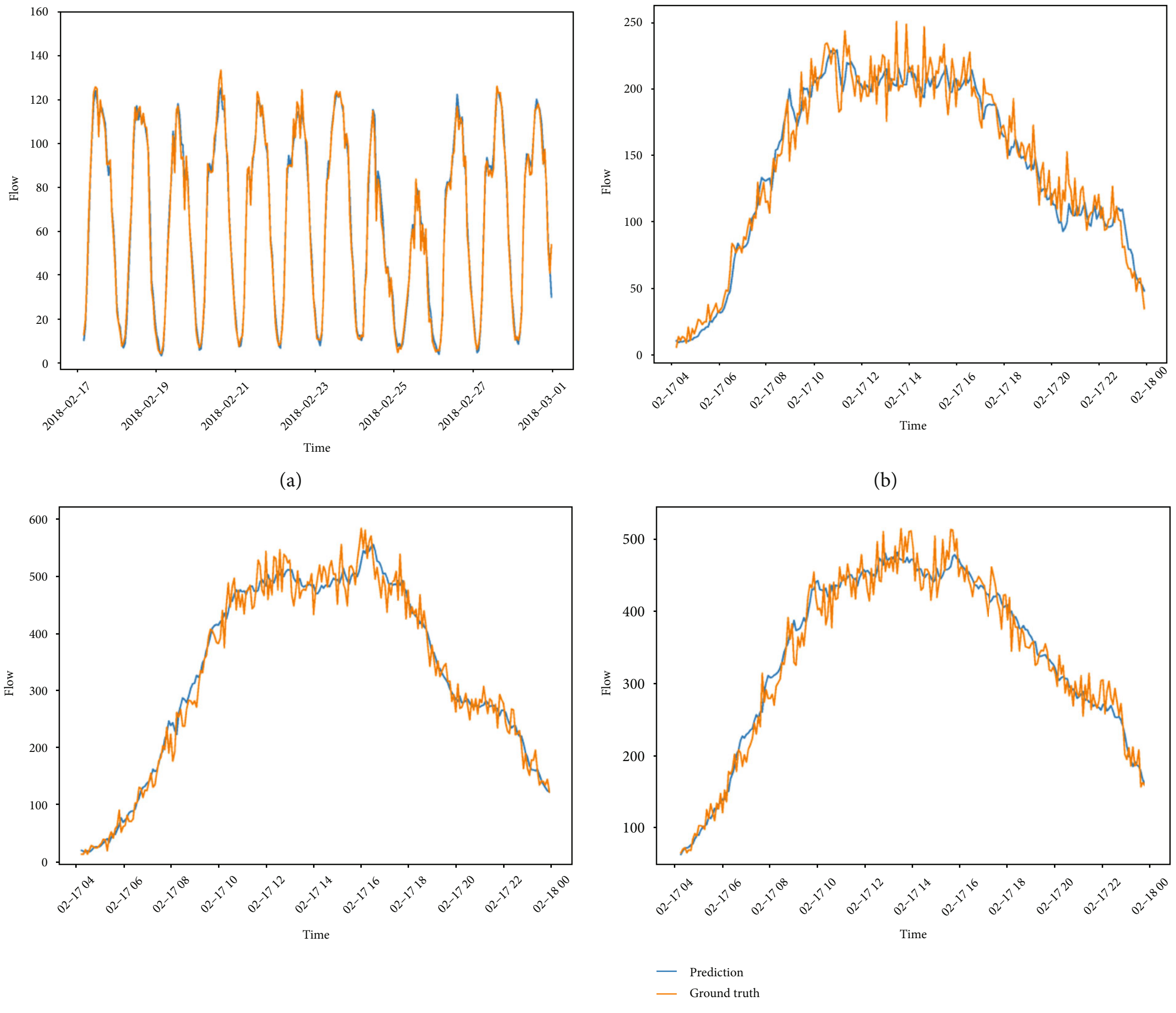


(a) (b)

Figure 7: The prediction results of different methods on PEMSD8.

attention mechanism performs better, indicating that the attention mechanism is able to capture the dynamic changes of the traffic network. The model that does not use the attention mechanism also has better results than the previous model, showing the superiority of the proposed model in extracting spatial-temporal features in traffic flow prediction and can combine with the spatial-temporal attention mechanism to improve the model effectiveness.

The prediction results of different models are compared by increasing the prediction time. From the overall view, as the prediction time keeps increasing, the possibility of disturbance becomes greater and the prediction becomes more difficult, so the prediction error also increases. It can be seen from the Figures 6 and 7 that if only temporal features are considered, such as HA, ARIMA and LSTM models, the prediction can still achieve good results in the short term, but as time grows, the prediction effect decreases sharply. In contrast, forecasting methods based on temporal features tend to maintain a stable effect. The model proposed in this paper achieves the best performance after introducing the attention mechanism, and also ensures excellent results in the long-term prediction process.



(a)

(b)

(c)

(d)

Figure 8: Predicted traffic flow vs. real traffic flow.

3.4. Forecast Result. In order to verify the validity of the model, the data in the test set is used for verification, and the result is shown in Figure 8. They, respectively, represent the overall prediction effect on the test data set and the one-day prediction effect. It can be seen that the prediction effect of the model proposed in this article on the test set is relatively ideal, and it can basically fit the real data, which proves the performance of the model proposed in this article. Effectiveness.

3.5. Ablation Experiment. Many studies have proved that ablation experiments can analyze the effectiveness of each module [32, 33], so we make the following changes to the model, let -A and -C represent the removal of the attention mechanism and the use of LSTM as the temporal feature extraction module, respectively. The variants of the model are, respectively, for:

(1) MASTN(-A/-C): LSTM is used to extract temporal features without the attention mechanism

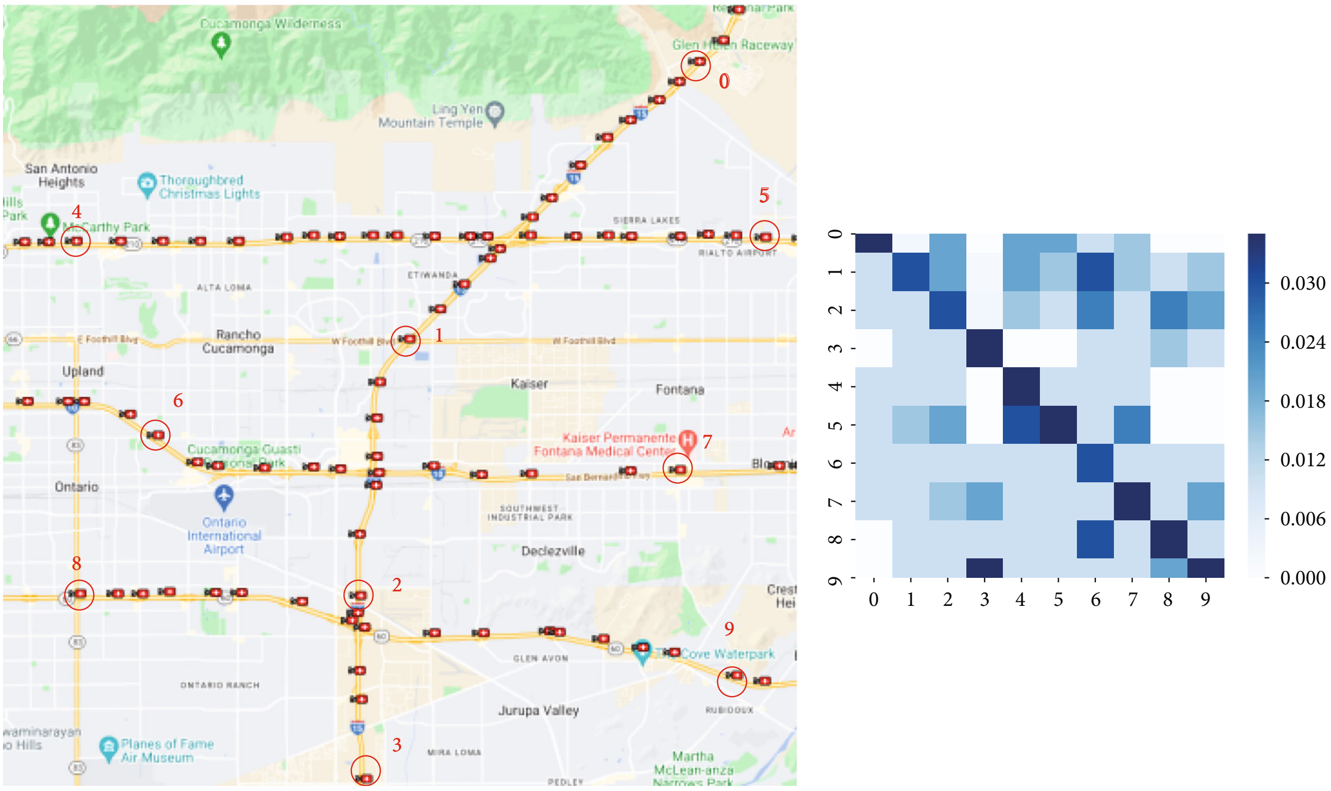
(2) MASTN(-A): does not use the attention mechanism

(3) MASTN(-C): Replace ConvLSTM with LSTM

Table 2: Ablation experiment results.

Model	RMSE	MAE
MA-STN	22.43	19.33
MASTN($-A$)	28.91	21.99
MASTN($-C$)	28.31	20.41
MASTN($-A/-C$)	30.78	22.2

As shown in Table 2, after removing the attention mechanism, the prediction effect of the model drops significantly. At the same time, in order to further analyze the role of the attention mechanism in the model, a subgraph containing 10 nodes is selected, and the spatial attention matrix between nodes on the PeMSD8 dataset is displayed. Figure 9 is the



(a) subgraph location

(b) attention matrix

Figure 9: Spatial Attention Matrix.

attention matrix of the subgraph Compared with the actual road node position, Figure 9(a) represents the geographic location and label of the selected node, and Figure 9(b) represents the weight status of the node in the attention matrix, which represents the correlation strength between different nodes, It can be seen that the traffic flow on the 9th node on the subgraph is closely related to the 3rd and 8th nodes, which is determined by the spatial proximity on the road. Therefore, the model proposed in this paper can reflect the spatial characteristics of real roads and is interpretable.

As shown in Table 2, when the ConvLSTM module in the MASTN model is replaced by the LSTM model, or the attention mechanism and ConvLSTM are replaced by the LSTM model, the performance of the model will be degraded. ConvLSTM is a convolution operation that adds spatial feature extraction to the model in the time dimension, so it is necessary to effectively consider the spatial features in the time dimension for the task of traffic flow prediction, which can effectively improve the prediction effect of the model.

4. Conclusions

In this paper, we propose a multidimensional spatial-temporal attention graph model MA-STN and successfully deal with the traffic flow prediction problem. The model combines the spatial-temporal attention mechanism with the spatial-temporal convolution module to process the recent time series, the daily cycle time series and the weekly cycle time series separately, and use the fusion unit to fuse the output features to predict the traffic flow data. Experiments on two real datasets show that the model has better prediction accuracy than existing models and is interpretable. For a real road traffic situation, the traffic condition is affected by various external factors, and the road condition is not constant, for example, the construction of a critical path can have a great impact on the traffic condition of the whole road network. Therefore, in the future, we will consider adding the dynamic changes of the traffic map into consideration. Since the model proposed in this paper can be generalized to spatial-temporal prediction tasks, it can also be applied to other practical applications, such as predicting arrival times, predicting pedestrian flow, etc.

Data Availability

The datasets used are public. PeMSD4 and PeMSD8 can be downloaded from https://drive.google.com/drive/folders/17fwxGyQ3Qb0TLOalI-Y9wfgTPuXSYgiI.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the Technology Innovation and Application Development Projects of Chongqing (Grant No. cstc2021jscx-gksbX0032, cstc2021jscx-gksbX0029); the Chongqing Research Program of Basic Research and Frontier Technology (Grant No. cstc2021jcyj-msxmX0530) the Key R & D plan of Hainan Province (Grant No. ZDYF2021GXJS006); the National Natural Science Foundation of China (Grant No. 62106030).

References

[1] Z. Diao, X. Wang, D. Zhang, Y. Liu, K. Xie, and S. He, "Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 890–897, 2019.

[2] L. Zhou, S. Zhang, J. Yu, and X. Chen, "Spatial–temporal deep tensor neural networks for large-scale urban network speed prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3718–3729, 2019.

[3] J. Zhou, G. Cui, S. Hu et al., "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.

[4] C. Zheng, X. Fan, C. Wen, L. Chen, C. Wang, and J. Li, "DeepSTD: mining Spatio-temporal disturbances of multiple context factors for citywide traffic flow prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3744–3755, 2020.

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

[6] L. Zhao, Y. Song, C. Zhang et al., "T-GCN: a temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3848–3858, 2020.

[7] Z. Zhang, M. Li, X. Lin, Y. Wang, and F. He, "Multistep speed prediction on traffic networks: A deep learning approach considering spatio-temporal dependencies," *Transportation Research Part C: Emerging Technologies*, vol. 105, pp. 297–322, 2019.

[8] Q. Zhang, J. Chang, G. Meng, S. Xiang, and C. Pan, "Spatio-temporal graph structure learning for traffic forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1177–1185, 2020.

[9] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: a deep learning framework for network-scale traffic learning and forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4883–4894, 2020.

[10] G. Yu and C. Zhang, "Switching ARIMA model based forecasting for traffic flow," *Institute of Electrical and Electronics Engineers Inc.*, vol. 2, pp. I429–I432, 2004.

[11] J. Ye, J. Zhao, K. Ye, and C. Xu, "How to build a graph-based deep learning architecture in traffic domain: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, pp. 453–476, 2020.

[12] E. Zivot and J. Wang, "Vector autoregressive models for multivariate time series," *Modeling Financial Time Series with SPLUSR*, vol. 21, pp. 385–429, 2006.

[13] S. Cheng, F. Lu, P. Peng, and S. Wu, "Short-term traffic forecasting: An adaptive ST-KNN model that considers spatial heterogeneity," *Computers, Environment and Urban Systems*, vol. 71, pp. 186–198, 2018.

[14] R. Chen, C. Y. Liang, W. C. Hong, and D. X. Gu, "Forecasting holiday daily tourist flow based on seasonal support vector regression with adaptive genetic algorithm," *Applied Soft Computing*, vol. 26, pp. 435–443, 2015.

[15] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *NIPS 2014 Workshop on Deep Learning*, 2014.

[16] H. Yao, F. Wu, J. Ke et al., "Deep multi-view spatial-temporal network for taxi demand prediction," *In AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, pp. 2588–2595, 2018.

[17] C. Song, Y. Lin, S. Guo, and H. Wan, "Spatial-Temporal synchronous graph convolutional networks: a new framework for spatial-temporal network data forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 1, pp. 914–921, 2020.

[18] J. Zhang, Y. Zheng, and D. Qi, "Deep Spatio-temporal residual networks for citywide crowd flows prediction," *Thirty-first AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, pp. 2354–2364, 2017.

[19] X. Geng, Y. Li, L. Wang et al., "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 3656–3663, 2019.

[20] M. Simonovsky and N. Komodakis, "Dynamic edgeconditioned filters in convolutional neural networks on graphs," *Computer Vision and Pattern Recognition*, pp. 3693–3702, 2017.

[21] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: data-driven traffic forecasting," https://arxiv.org/abs/1707.01926, 2017.

[22] S. Guo, Y. Lin, S. Li, Z. Chen, and H. Wan, "Deep Spatial–Temporal 3D Convolutional Neural Networks for Traffic Data Forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3913–3926, 2019.

[23] K. Xu, J. Ba, R. Kiros et al., "Show, attend and tell: neural image caption generation with visual attention," *International conference on machine learning*, pp. 2048–2057, 2015.

[24] J. Wang, Q. Chen, and H. Gong, "STMAG: A spatial-temporal mixed attention graph-based convolution model for multi-data flow safety prediction," *Information Sciences*, vol. 525, pp. 16–36, 2020.

[25] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 922–929, 2019.

[26] X. Feng, J. Guo, B. Qin, T. Liu, and Y. Liu, "Effective deep memory networks for distant supervised relation extraction," *Processing In International Joint Conference on Artificial Intelligence*, pp. 19–25, 2017.

[27] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2019.

[28] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," https://arxiv.org/abs/1506.05163, 2015.

[29] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: mining loop detector

data," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1748, no. 1, pp. 96–102, 2001.

[30] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *International Conference on Learning Representations*, pp. 3625–3634, 2016.

[31] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modelling," in *International Joint Conferences on Artificial Intelligence*, pp. 1907–1913, Macao, China, 2019.

[32] M. Lin, C. Huang, R. Chen, H. Fujita, and X. Wang, "Directional correlation coefficient measures for Pythagorean fuzzy sets: their applications to medical diagnosis and cluster analysis," *Complex & Intelligent Systems*, vol. 7, no. 2, pp. 1025–1043, 2021.

[33] M. Lin, Q. Zhan, and Z. Xu, "Decision making with probabilistic hesitant fuzzy information based on multiplicative consistency," *International Journal of Intelligent Systems*, vol. 35, no. 8, pp. 1233–1261, 2020.