A local global attention based spatiotemporal network for traffic flow forecasting

Yuanchun Lan*[a], Jiahao Ling[a], Xiaohui Huang[a], Junyang Wang[a]

[a]*organization= Department of Information Engineering, East China Jiaotong University,addressline=NO.808, Shuanggang East Street, city=NanChang, postcode=330000, state=JiangXi, country=China*

Abstract

Accurate traffic forecasting is critical to improving the safety, stability, and efficiency of intelligent transportation systems. Although many spatiotemporal analysis methods have been proposed, accurate traffic prediction still faces many challenges, for example, it is difficult for long-term predictions to model the dynamics of traffic data in temporal and spatial to capture the periodicity and spatial heterogeneity of traffic data. Most exsiting studies relieve this problem by discovering hidden spatiotemporal dependencies with graph neural networks and attention mechanisms. However, the period-related information between spatiotemporal sequences is not sufficiently considered in these models. Therefore, we propose a local global spatiotemporal attention network (LGA) to solve the above challenge. Specifically, we present a local spatial attention module to extract the spatial correlation of hourly, daily and weekly periodic information. We propose a weight attention mechanism to assign different weights of the periodic feature extracted on local spatial attention. The local periodic temporal features are extracted through local temporal attention we proposed. And we develop the global spatiotemporal attention module to extract the global spatiotemporal information of the entire time slice, which is more conducive to learning the periodic features of traffic data. The extensive experiments on four real-world datasets demonstrate the effectiveness of our proposed model.

Keywords: Spatiotemporal, Traffic forecasting, Periodicity, Spatial heterogeneity

1. Introduction

Currently, many cities are working on improve the performance of intelligent transportation systems (ITS). Traffic flow forecasting has become an crucial part of traffic planning, control, and status assessment in smart city development[1]. Traffic flow forecasting is the use of observed historical traffic data to predict future traffic flow. Accurate traffic prediction can help reduce road congestion, promote urban traffic network management, and even improve traffic efficiency [2]. Traffic data is a type of time series data that is recorded at regular intervals by deployed road sensors. Although a lot of researches [1, 2] has been done in the field of traffic flow forecasting in recent years to improve prediction performance, it still confronts some challenges. Traffic data has complex temporal and dynamic spatial correlations. At the same time, traffic data is a kind of time series data with explicit periodicity and trends, such as morning and evening peaks, weekdays and rest days. Effectively capturing periodicity and trends requires models to accurately capture the long-term dependencies of temporal and spatial,which are major challenges in urban traffic forecasting tasks [2]. The most advanced traffic forecasting models can be divided into three categories: Grid-based models [3, 4, 5], Graph-based models [6, 7], and Multivariate time series models [8, 9, 10]. To capture spatial dependence, grid-based models usually use ordinary convolution operations[11]; graph-based models utilize graph convolution in non-Euclidean space by involving the adjacency relation between each pair of

*Email address: lyc28688@163.com

road sensors [12, 13]. At the same time, attention mechanisms [14] are also used as an alternative technique for modeling spatial and temporal dependencies. Multivariate time series (MTS) models such as GeoMAN [8] and Transformer [9] are also evolving on spatial and temporal axes. From a spatial perspective, they focus on correlations between variables. From a temporal perspective, their goal is to take advantage of the periodic patterns that appear in the time series.

Although deep learning methods consider both spatial dependences and temporal dynamics in traffic prediction, it does not fully consider the relevant information between spatiotemporal series. The current approaches has three main limitations:

- The spatial dependence between traffic flow sonsors are extracted by learning a static relationship building on the similarity of historical traffic in the most of traditional methods[15, 16]. However, the dependence of traffic flow among locations may change over time. For example, spatial dependence between a residential area and a commercial center may be strong in the morning; the spatial dependence between the two places may be weak in the middle of the night. However, this dynamic dependency is overlooked in most of previous methods.
- Many existing studies do not consider periodicity independently from different perspectives. Traffic flow data shows strong hourly, daily, and weekly periodicity, which give us an important evidence for improving the accuracy of traffic flow forecasting. Although GraphWavenet [17] and MTGNN [18] use adaptive adjacency matrices to learn spatial correlations, these methods learn global spatial dependence without taking into account different periodicity sophisticatedly.

To address these challenges, we propose a Local Global Attention based spatiotemporal network (LGA) for traffic flow forecasting. Firstly, local Graph Attention Network (GAT), local Residual Efficent Channel Attention (RECA) and Weight Attention are proposed to extract local periodic spatiotemporal information. Secondly, global GAT and global RECA are designed to process global spatial and temporal features. Our model is able to further extract the periodicity of the data to obtain more potential feature, making the model predictions more accurate.

We conducted extensive experiments on four real datasets, and the results verified the effectiveness of LGA through comprehensive comparison with existing methods. Our main contributions are as follows:

- We propose a local spatial attention and global spatial attention to extract periodic spatial features and global spatial features, respectively. We use local Graph Attention Network including WEEK-GAT, DAY-GAT, and HOUR-GAT to extract features of periodic data, and local Graph Attention Network adaptively learn the weekly node embedding, daily node embedding and hourly node embedding. Hourly node embedding can be seen as global node embedding, due to its importance in global flow prediction.
- We propose a weight attention mechanism to fuse the extracted weekly spatial features, daily spatial features, and hourly spatial features so that the our proposed model has a capacity to determine the importance of each periodicities.
- We propose the local Residual Efficient Channel Attention Module(RECA) to extract local temporal features and global RECA to extract global temporal features. Thus, RECA can extract different periodical information hidden in the historical data.
- We conducted extensive experiments on four real-world datasets and experimental results show that our model outperforms baseline models.

2. Related work

Due to the availability and importance of large-scale traffic data in the real world, traffic prediction is receiving more and more attention in the field of artificial intelligence research. In this section, we briefly review traditional time series forecasting methods and deep learning-based traffic flow forecasting methods.

2.1. Traffic flow forecasting based on traditional time series forecasting methods

Traffic prediction relies on a combination of spatiotemporal features and has been studied for decades. Traditional methods can only consider the relationship from a single view in the time dimension, lack spatial information, and can only learn simple time series information. Traditional time series forecasting methods such as Holt-Winters [19] are applicable to non-stationary series with linear trends and periodic fluctuations, which use exponential smoothing (EMA) to continuously adapt model parameters to the non-stationary sequences, and make short-term forecasts for future trends. The autoregressive synthetic moving average (ARIMA) [20] and the Kalman filter (KF) [21, 22] are mainly based on linear operations and are widely used in flow prediction. Due to the randomness and nonlinearity of traffic flow, nonparametric methods [23, 24] have received a lot of attention in the field of traffic flow prediction. These methods mainly include traditional machine learning methods K-Nearest Neighbors (KNN) [23], support vector regression (SVR) [24]. However, these linear arithmetic-based methods and traditional machine learning methods are mainly suitable for univariate forecasting problems, limiting their application in complex time series data. Complex nonlinear spatial and temporal correlations cannot be simulated in these traditional time series forecasting methods.

2.2. Traffic flow forecasting based on deep learning methods

In traffic forecasting, a series of studies based on deep learning techniques have been proposed. Deep neural networks combine variants of CNNs and RNNs to greatly improve the accuracy of traffic prediction, and have been extensively studied in designing spatiotemporal data modeling methods. Graph convolutional networks [13] have greatly relieve the deficiencies of previous works in extraction of the spatial information. Recent works [25] hava focused on designing complex graph convolutional recurrent network architectures to capture spatial and temporal patterns. DCRNN [26] explains the spatial dependence of traffic as a diffusion process and expands the previous GCN into a directed graph. GraphWavenet [17] follows DCRNN by combining GCN with an extended causal convolutional network to save computational costs on processing long sequences, and proposes adaptive adjacency matrices to learn hidden spatial features to capture spatial correlations. Recently, due to the effectiveness and efficiency of the attention mechanism, models based on the attention mechanism have been more and more widely used in the field of traffic prediction. Recent studies such as GMAN [7] use spatial and temporal attention mechanisms to model dynamic spatial and nonlinear temporal correlations, respectively, through gated fusion to adaptively fuse information extracted by spatial and temporal attention mechanisms. LSGCN [27] uses graph attention networks and graph convolutional networks (GCNs) to be integrated together in a gated form. Spatial gating blocks can accurately extract spatial location, adjacencency, and similar road conditions of the transportation network. In the past, deep learning models used graph neural networks or graph attention networks to prompt spatial features, but they solved the periodic problem of traffic data less and could not extract more hidden spatiotemporal features.

3. Preliminaries

In this section, we will introduce the notations and definitions related to the traffic flow prediction.

Definition 1. *Traffic spatiotemporal sequence: We define a traffic flow spatiotemporal sequence as* $X = \left(X^1, X^2, \ldots, X^T\right) \in R^{T\times N\times C}$*, where* $X^t = (x_1^t, x_2^t, \ldots, x_N^t) \in R^{N\times C}$ *denotes the feature vectors of* N *sensors with* C *attributes at timestep* t*.*

Definition 2. *Periodic Data: We define the hourly, daily and weekly time intervals as* T_h*,* T_d *and* T_w*, respectively. Given time window* τ*, the historical periodic data can be defined as:*

$$\left(\begin{array}{l} X^w = X^{t-T_w+1}, X^{t-T_w+2}, \ldots, X^{t-T_w+\tau}, \\ X^d = X^{t-T_d+1}, X^{t-T_d+2}, \ldots, X^{t-T_d+\tau}, \\ X^h = X^{t-T_h+1}, X^{t-T_h+2}, \ldots, X^{t-T_h+\tau}. \end{array}\right) \tag{1}$$

Traffic flow forecasting: Given the historical periodic data $X \in R^{T \times N \times C}$ defined as Equation 1, the aim of this work is to predict traffic flow for all sensors at the next L timesteps, i.e.,

$$f\left(X^h, X^d, X^w, E\right) \rightarrow \left(X^{t+1}, X^{t+L}, \ldots, X^{t+L}\right) \in R^{L \times N \times C}, \tag{2}$$

where $E = [e_d, e_w, e_h] \in R^{3 \times N \times F}$ are three node embedding matrices of daily, weekly and hourly periodicities, F is the embedding dimension, $f(\bullet)$ is the mapping function aimed at learning the future traffic flow.

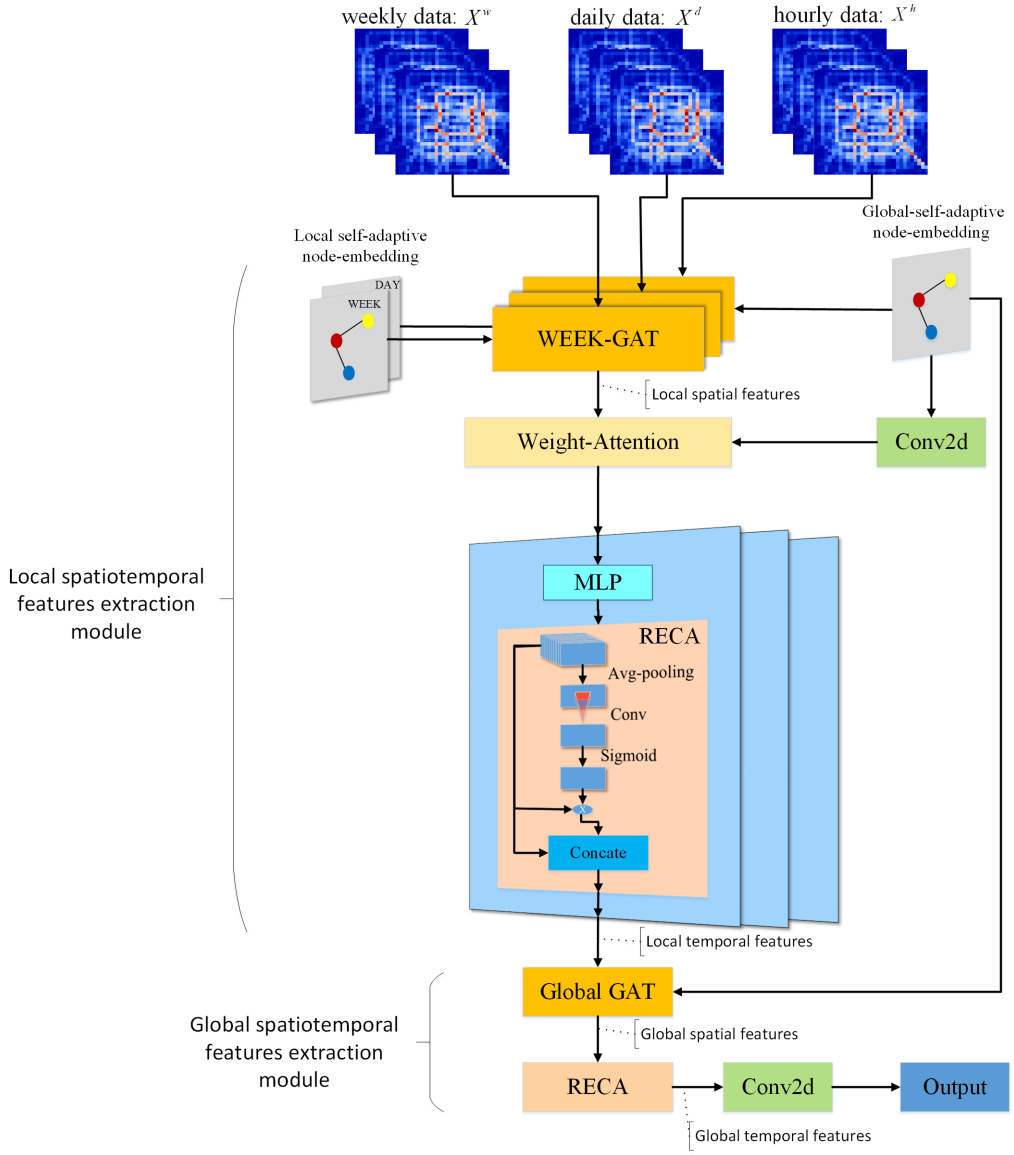


Figure 1: The framework of our proposed model.

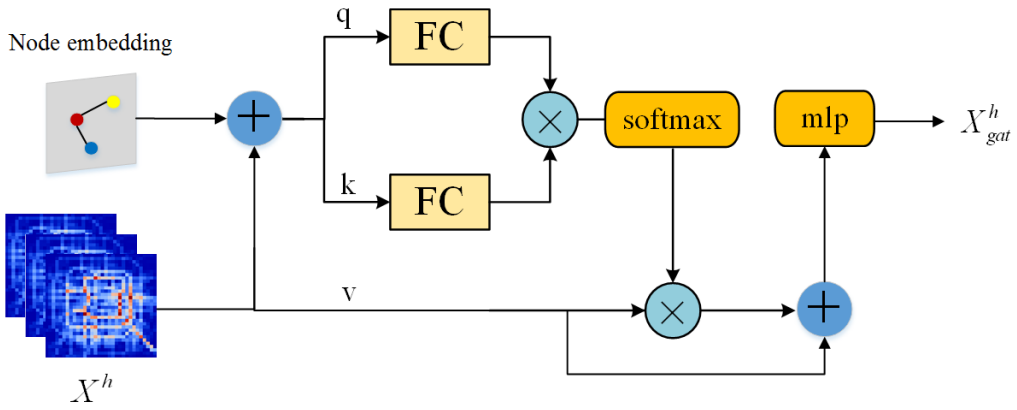


Figure 2: The framework of GAT.

4. Method

In this section, we will introduce our proposed model from three aspects: the overall structure, weighted local spatial attention module, and global spatiotemporal attention module.

4.1. The framework

We show the skeleton of our proposed model in Figure 1. It consists of a local spatiotemporal module and a global spatiotemporal mudule. Firstly, we feed the node embedding matrix of hourly periodicity learned adaptively with the framework and hourly data into HOUR-GAT extract the spatial features in this periodicity. The processing procedure of daily and weekly data is same as that of hourly data. Thus, we can gain hourly spatial features, daily spatial features and weekly spatial features. Then, the weight attention module is used to fuse the three kinds of periodic features with global node embeddings to obtain the fusing period spatial features of three kinds of views. The local temporal features extraction module consists of three layers, each of layers is processed by Multi-Layer Perception (MLP) and RECA to output local temporal features. Finally, the global spatiotemporal features are extracted by the global spatiotemporal features extraction module. In this module, the features extracted by the local spatiotemporal features extraction module and the global node embedding matrix are input into Global-GAT to learn the global spatial features, and then the global temporal features are extracted by the global RECA module. After all processing, we get the spatiotemporal features with periodicity features and global spatiotemporal features.

4.2. Local spatiotemporal features extraction module

The local spatiotemporal features extraction module includes the local spatial features extraction module, weight attention module, and the local temporal features extraction module. The local spatial features extraction module includes HOUR-GAT, DAY-GAT, and WEEK-RGAT. The local temporal features extraction module continuously processes the fusing period spatial features with three same blocks where each block is composed of a MLP and a RECA. In the following, we introduce the details of local spatiotemporal features extraction module from two aspects.

Local spatial features extraction module: Considering different traffic conditions, the impact to the traffic flow of each sensor from other roads can be dynamic. For example, when roads are congested, the impact of one road on other roads may be stronger than when it has low capacity [28]. In order to better learn dynamic spatial dependence, we introduce a self-attention module to extract features from the different periodic traffic flow data and adaptively learn the node embedding matrix, which is shown in Figure 2. We feed the the orgin periodic data and node embedding to GAT modules (including HOUR-GAT, DAY-GAT

and WEEK-GAT) to learn correlations between nodes. In the follow, we introduce the computational procedure of the correlation between node i and node j on hourly periodicity as an example:

$$S_{i,j}^{h} = \frac{\left[W_q^h \left(X_i^h \left\| e_i^h \right.\right)\right] \otimes \left[W_k^h \left(X_j^h \left\| e_j^h \right.\right)\right]}{\sqrt{d}}, \tag{3}$$

where $X^h \in R^{\tau \times N \times C}$ is the original hourly periodic data, and X_i^h and X_j^h represents the original hourly periodic data of node i and node j; τ represents the number of time steps on each periodicity; The node embedding matrix $e^h \in R^{N \times F}$ is reshaped and repeated to $e^h \in R^{\tau \times N \times C}$, where $e_i^h \in R^{\tau \times C}$ represents the node embedding of node i on hourly periodicity; $\|$ represents a concatenation operation, $\otimes$ denotes the inner product operation, $S_{i,j}^h \in R$ represents the correlation between node i and node j on hourly periodicity; W_q^h and W_k^h is a weighting learnable parameters of query and key, d is the dimension of key and value. After calculating the correlation between nodes, we use the softmax function to $S_{i,j}^h$ to generate attention scores $a_{i,j}^h$, the formula is as follows:

$$a_{i,j}^h = soft\max(S_{i,j}^h). \tag{4}$$

After getting the attention score, we calculate a weighted sum from the correlation of all nodes, and get the output $X_{gat}^h \in R^{\tau \times N \times C}$, the formula is as follows:

$$X_{i,gat}^h = \sum_{j=1}^{n} a_{i,j}^h X_j^h + X_j^h \tag{5}$$

where $X_{i,gat}^h$ is the feature produced by GAT module on the hourly periodic data for node i, X_j^h represents the hourly periodic data of node j. X_{gat}^d and X_{gat}^w are processed similarly to X_{gat}^h, We put daily data and weekly data respectively into DAY-GAT and WEEK-GAT modules to generate periodic daily features and weekly features.

Weight attention module: In order to determine the importance of hourly data, daily data, and weekly data, we use weight attention to integrate the three periodicial features adaptively. Therefore the features of periodic have a certain bias on training. And we fuse the global node embedding matrix with the weighted periodic features to obtain the local spatial features X_{wgat}, the formula is as follows:

$$X_{watt} = X_{gat}^h \left\| \alpha X_{gat}^d \right\| \beta X_{gat}^w \| e_h \ , \tag{6}$$

where $\|$ represents a concatenation operation, e_g represents a global node embedding matrix. α and β represent the adaptive parameters of weighted attention mechanisms which is learnt with the whole framwork. Since the two adaptive parameters change, the third parameter is also relatively changed, so here we only use two adaptive parameters to save the amount of parameters and reduce the time complexity.

Local temporal features extraction module: The local temporal feature extraction module has three layers, and each layer includes an MLP and a RECA module. We extract local periodic temporal features through the local temporal feature extraction module, which can be formulated as:

$$X_{mlp}^{l+1} = FC_2^l \left(\sigma \left(FC_1^l \left(X_{RECA}^l\right)\right)\right) + X_{RECA}^l, \tag{7}$$

where FC_2^l and FC_1^l represent two different layers of fully-connected network, σ represents the sigmoid function, and l represents the MLP layer of the l^{th} layer, X_{RECA}^l represents the RECA module output of the previous layer, when $l = 1$, $X_{RECA}^l = X_{watt}$. Then we design a Residual Efficient Channel Attention Module (RECA) to extract temporal features, the formula is as follows:

$$g\left(X\right) = \frac{\sum_{i=1,t=1}^{N,\tau} X_i^t}{N \times \tau}, \tag{8}$$

$$X_{RECA}^{l+1} = \sigma\left(\theta_{local} * g\left(X_{mlp}^{l+1}\right)\right) \otimes X_{mlp}^{l+1} + X_{mlp}^{l+1}, \tag{9}$$

where $g(X)$ is the global average pooling characteristic information, σ stands for the sigmoid function, θ_{local} represents a weight, $*$ represents one-dimensional convolution operation, N represents the number of nodes and τ represents the time step. RECA only involves a few parameters, but can significantly improve the model performance, the experiment shows that RECA module has certain effect.

4.3. Global spatiotemporal feature extraction module

Although our local spatiotemporal feature extraction module has extracted the periodic features of the data, we still have to further extract global features. The global spatiotemporal feature extraction module includes the Global-GAT module and the global RECA module to extract the global spatial features and temporal features, respectively. First of all, we input the extracted local spatiotemporal features into Global-GAT, extract global spatial features. The calculation process of this module is the same as the local spatial features extraction module, the formula is as follows:

$$X_{global-gat} = \sum a_{i,j}^{h} \times X_{RECA}, \tag{10}$$

where X_{RECA} is the output of the last RECA module layer on local temporal features extraction module. After global spatial feature extraction. We feed $X_{global-gat}$ into the global RECA module, the formula is as follows:

$$X_{global-RECA} = \sigma\left(\theta_{global} * g\left(X_{global-gat}\right)\right) \otimes X_{global-gat} + X_{global-gat}, \tag{11}$$

θ_{global} represents a weight, $*$ represents one-dimensional convolution operation.

5. Experiment

In this section, we conducted experiments with LGA on four publicly available datasets to demonstrate the effectiveness of LGA in traffic flow prediction. We introduce the experimental datasets and preprocessing, baseline models, results comparison, analysis and training strategies. We compare the performance of LGA with the baseline models in recent years to verify the effectiveness of our proposed model. In addition, we designed some ablation experiments to evaluate the impact of our developed modules.

5.1. Datasets and pre-processing

We verified LGA's performance on three public traffic datasets (PEMS07, PEMS08, and PEMS-BAY) and on an electricity consumption dataset. The PEMS07, PEMS08, and PEMS-BAY datasets were collected by the Caltrans Performance Measurement System (PeMS). The Electricity dataset is the measurements of electric power consumption in one household with a one-minute sampling rate over a period of almost 4 years in a house located in Sceaux (Paris, France). The properties of the datasets are shown in Table 1.

Table 1: Details of the datasets.

Dataset	Number of Sensors	Time Period	Time Interval
PEMS-BAY[29, 30, 17]	325 sensors	2017/1/1-2017/5/31	5 minutes
PEMS07 [25]	883 sensors	2017/5/1-2017/8/31	5 minutes
PEMS08[31, 25]	170 sensors	2016/7/1-2016/8/31	5 minutes
Electricity [32]	321 sensors	2017/1/1-2017/5/31	1 hour

To make a fair comparison, we followed the dataset division with the previous works[33]. The ratio of the training, validation, and testing sets of the PEMS-BAY dataset is 7:1:2, while the ratio of other datasets is 6:2:2. Our goal is to predict future time series with length 12, i.e.. We compared the performance of these methods at time steps 3, 6, and 12, as well as the performance of an average of 12 time steps.

5.2. Baseline Methods

In our experiments, we compare our proposed method with the following 9 baselines:

- DCRNN(2018) [26]:A diffusion convolutional recurrent neural network which combines diffusion graph convolution and recurrent neural networks.
- STGCN(2018) [34]: A spatial–temporal graph convolution network that combines graph convolution and one-dimensional convolution.
- Graph Wavenet(2019) [17]: A spatial–temporal graph convolutional network that integrates diffusion graph convolution and one-dimensional expansion convolution.
- AGCRN(2020) [25]: Using two adaptive modules for enhancing Graph Convolutional Network with new capabilities to capture fine-grained spatial and temporal correlations in traffic series.
- GMAN(2020) [7]: Graph multi-attention network with spatial and temporal attention.
- MTGNN(2020) [18]: A spatial-temporal network for generating one-way adaptive graphs using external features.
- StemGNN(2020) [35]: Combining Graph Fourier Transform models inter-series correlations and Discrete Fourier Transform models temporal dependencies in an end-to-end framework.
- STNorm(2021) [36]: Using two kinds of normalization modules: temporal and spatial normalization which separately refine the high-frequency component and the local component underlying the raw data.
- STID(2022) [33]: Identifying the indistinguishability of samples in both spatial and temporal dimensions, attaching spatial and temporal identity information for MTS forcasting.

5.3. Parameters setting

In the experiments, the parameter setting of each module is shown in the Table 2. First, the number layers in local temporal features extraction module is set to 3. The embedding dimension F of node embeddings is set to 16 in the experiments. The time steps τ is set 12 in the experiment. The optimization algorithm used in the training process is Adam, and the learning rate on four datasets is set to 0.002. The weight decay in the training process is set to 0.0001. the batch size in the training process is set to 32.

Table 2: The parameter setting in the experiment.

Parameter	**Value**
Number of layers	3
F	16
τ	12
Optimizer	Adam
Learning rate	0.002
Weight decay	0.0001
Batch size	32

5.4. Experimental results and analyses

Tables 3, 4, 5 and 6 show the comparison results produced by the different methods for the next hour on three real traffic datasets and one a power dataset. Our proposed LGA outperforms all baseline methods on these four datasets in most cases. On the Electricity dataset, LGA achieved 4.6% and 4.1% improvement with respects to MAE and RMSE compared with the best method in the baselines, respectively. On the PEMS08 dataset, LGA obtained 2.6% MAE improvement, 2.1% RMSE improvement, and 3.2% MAPE improvement in comparison to the STID method.

The results of Tables 3, 4, 5 and 6 show that GWNet based on adaptive adjacency matrix has better performance than STGCN with 7.16% MAE improvement, 6.30% RMSE improvement, and 3.2% MAPE improvement; The possible reason is that an adaptive adjacency matrix with good ability to capture spatial correlations. The models GMAN based on the attention mechanism performs better than GWNet in the long term forecast with 2.43% MAE improvement, 2.42% RMSE improvement, and 2.63% MAPE improvement in timestep 12, the possible reason is that the attention mechanism can capture the temporal correlation of long sequences. STID can achieve better performance than GMAN with 4.55% MAE improvement, 1.37% RMSE improvement, and 5.33% MAPE improvement, which only use basic multi-layer MLP, the possible reason is that they improve on the preprocessing of data, divides the data into periodic data. The results demonstrates the importance of the periodicity of data to traffic data.

Our proposed LGA extracts features from local periodicity and global trendency. We use GAT capture more hidden spatial correlations at different periodic data to and learn the node embedding matrix of the periodicity. Tables 3, 4, 5 and 6 show that the difficulty of forecasting will increase with the increment of the predicting time steps. With MAE, RMSE and MAPE will continue to increase, our proposed LGA consistently outperforms the baseline model.

Table 3: The result of LGA compared to other baseline models on PEMS07.

Method	15 min			30 min			60 min			Avg		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
DCRNN	19.45	31.39	8.29%	21.18	34.42	9.01%	24.14	38.84	10.42%	21.20	34.43	9.06%
STGCN	20.33	32.73	8.68%	21.66	35.35	9.16%	24.16	39.48	10.26%	21.71	35.41	9.25%
GWNet	18.69	30.69	8.02%	20.26	33.37	8.56%	22.79	37.11	9.73%	20.25	33.32	8.63%
AGCRN	19.31	31.68	8.18%	20.70	34.52	8.66%	22.74	37.94	9.71%	20.64	34.39	8.74%
GMAN	19.25	31.20	8.21%	20.33	33.30	8.63%	22.25	36.40	9.48%	20.43	33.30	8.69%
MTGNN	19.23	31.15	8.55%	20.83	33.93	9.30%	23.60	38.10	10.10%	20.94	34.03	9.10%
StemGNN	19.74	32.32	8.27%	22.07	36.16	9.20%	26.20	42.32	11.00%	22.23	36.46	9.20%
STNorm	19.15	31.70	8.26%	20.63	35.10	8.84%	22.60	38.65	9.60%	20.52	34.85	8.77%
STID	18.31	30.39	7.72%	19.59	32.90	8.30%	21.52	36.29	9.15%	19.54	32.85	8.25%
LGA	18.15	30.06	7.67%	19.41	32.56	8.18%	21.29	35.77	9.02%	19.37	32.46	8.18%

Table 4: The result of LGA compared to other baseline models on PEMS08.

Method	15 min			30 min			60 min			Avg		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
DCRNN	14.16	22.20	9.31%	15.24	24.26	9.90%	17.70	27.14	11.13%	15.26	24.18	9.96%
STGCN	14.95	23.48	9.87%	15.92	25.36	10.42%	17.65	28.03	11.34%	15.98	25.37	10.43%
GWNet	13.72	21.71	8.80%	14.67	23.50	9.49%	16.15	25.95	10.74%	14.67	23.49	9.52%
AGCRN	14.51	22.87	9.34%	15.66	25.00	10.34%	17.49	27.93	11.72%	15.65	24.99	10.17%
GMAN	13.80	22.88	9.41%	14.62	24.12	9.57%	15.72	26.47	10.56%	14.81	24.19	9.69%
MTGNN	14.30	22.55	10.56%	15.25	24.41	10.54%	16.80	26.96	10.90%	15.31	24.42	10.70%
StemGNN	14.49	23.02	9.73%	15.84	25.38	10.78%	18.10	28.77	12.50%	15.91	25.44	10.90%
STNorm	14.44	22.68	9.32%	15.53	25.07	9.98%	17.20	27.86	11.30%	15.54	25.01	10.03%
STID	13.28	21.66	8.62%	14.21	23.57	9.24%	15.58	25.89	10.33%	14.20	23.49	9.28%
LGA	13.03	21.38	8.40%	13.86	23.11	8.97%	15.05	25.17	9.91%	13.84	23.01	8.99%

5.5. Ablation experiments

To further evaluate the efffectiveness of modules in LGA, we performed ablation experiments on the PEMS08 and Electricity datasets. We have designed six LGA variants as follows:

- w/o Global-GAT: On the base of LGA, the Global-GAT module is removed;

Table 5: The result of LGA compared to other baseline models on PEMS-BAY.

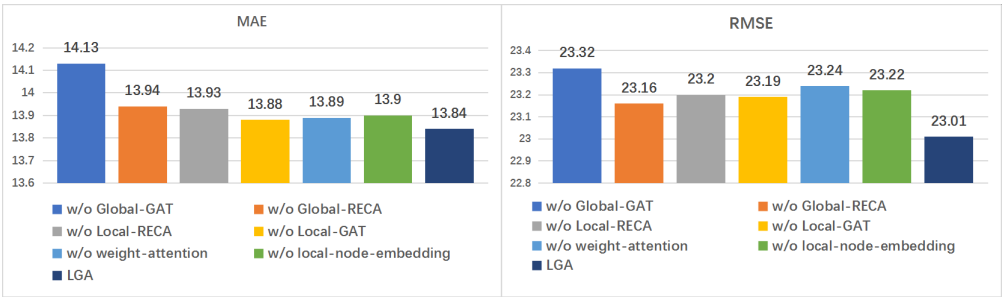
Method	15 min			30 min			60 min			Avg		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
DCRNN	1.31	2.80	2.73%	1.67	3.81	3.75%	1.99	4.66	4.73%	1.62	3.74	3.61%
STGCN	1.35	2.88	2.88%	1.69	3.83	3.85%	2.01	4.56	4.74%	1.63	3.73	3.69%
GWNet	1.30	2.78	2.71%	1.63	3.73	3.66%	1.95	4.52	4.63%	1.58	3.65	3.52%
AGCRN	1.37	2.93	2.95%	1.70	3.89	3.88%	1.99	4.64	4.72%	1.63	3.78	3.73%
GMAN	1.34	2.92	2.88%	1.65	3.81	3.71%	1.89	4.38	4.51%	1.58	3.75	3.69%
MTGNN	1.34	2.84	2.80%	1.67	3.79	3.74%	1.97	4.55	4.57%	1.60	3.70	3.57%
StemGNN	1.44	3.12	3.08%	1.93	4.38	4.54%	2.57	5.88	6.55%	1.92	4.46	4.54%
STNorm	1.34	2.88	2.82%	1.67	3.83	3.75%	1.96	4.52	4.62%	1.60	3.71	3.60%
STID	1.30	2.81	2.73%	1.62	3.72	3.68%	1.89	4.40	4.47%	1.55	3.62	3.51%
LGA	1.30	2.81	2.74%	1.60	3.63	3.61%	1.86	4.24	4.36%	1.53	3.51	3.44%

Table 6: The result of LGA compared to other baseline models on Electricity.

Method	15 min			30 min			60 min			Avg		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
AGCRN	22.88	49.98	41.33%	24.47	54.17	48.93%	27.24	59.76	52.57%	23.88	53.02	45.83%
MTGNN	16.78	36.91	48.16%	18.43	42.62	51.31%	20.49	48.33	56.25%	18.18	42.04	50.77%
StemGNN	21.45	41.09	57.12%	23.56	46.95	65.34%	24.98	51.97	62.81%	22.89	46.21	57.26%
STNorm	18.74	40.86	32.66%	21.14	48.24	37.07%	24.05	55.27	42.63%	20.69	47.55	35.98%
STID	16.08	34.49	31.95%	17.87	41.65	37.80%	19.25	45.77	40.26%	17.39	40.80	35.53%
LGA	15.01	34.03	31.06%	17.13	39.83	38.53%	18.99	45.21	47.74%	16.62	39.20	37.24%

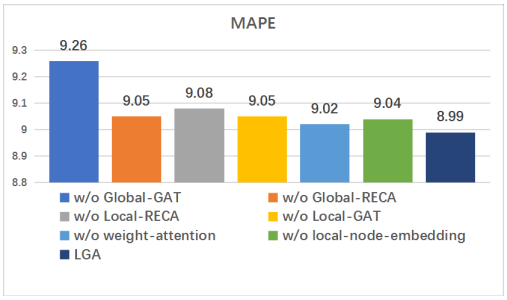
- w/o Global-RECA: On the base of LGA, the Global-RECA module is removed;
- w/o Local-RECA: On the base of LGA, the Local-RECA module is removed;
- w/o Local-GAT: The HOUR-GAT, DAY-GAT, and WEEK-GAT are removed;
- w/o weight-attention: On the base of LGA, the weight-attention module is removed;
- w/o Local-node-embedding: On the base of LGA, we only use one node-embedding instead of three node embedding.

The results of the ablation experiment are shown in Figure 3 and Figure 4. The impact of these components on the model is similar on the PEMS08 and Electricity datasets. Ablation eperiments have shown that it can significantly improve the performance of the model with 2.10% MAE improvement on PEMS08 and 3.61% MAE improvement on Electricity, suggesting that global spatial feature extraction is critical. After canceling the Global-RECA structure, the model performance degrades 3.31% MAE, 3.60% RMSE on Electricity, which indicates that global temporal feature extraction is effective. In addition, we conducted ablation studies on the local spatiotemporal feature extraction modules (Local-GAT, Local-RECA) within the LGA module, and Figure 3 and Figure 4 shows that local spatiotemporal feature extraction modules are effective for periodic feature processing of data. At the same time, we carried out an ablation experiment on the node embedding matrix of graph attention, and it can be seen from the ablation experiment that different node embedding matrices in different periods of the data can further strengthens the characteristic difference of node periodicity.



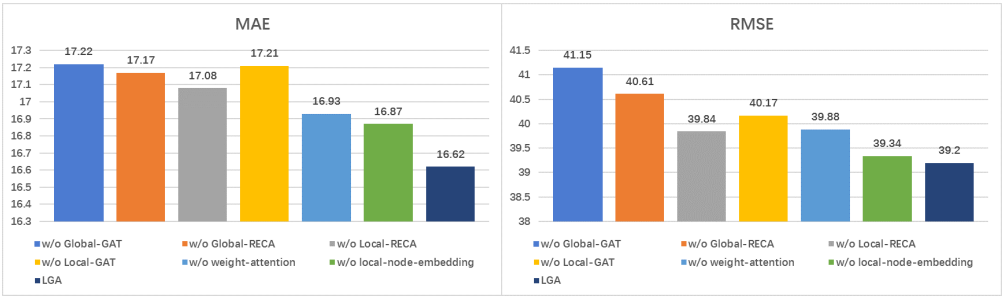
(a)

(b)



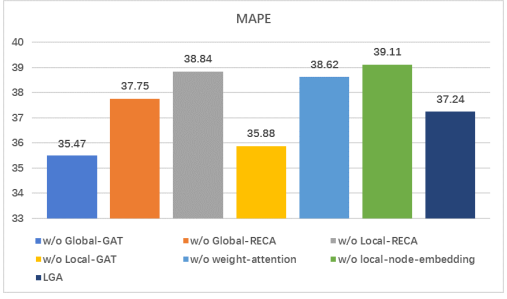
(c)

Figure 3: Component analysis of LGA on PEMS08 datasets. Each of the components in LGA contributes a positive effect. Among them, Global-GAT is the component that influences the model most.



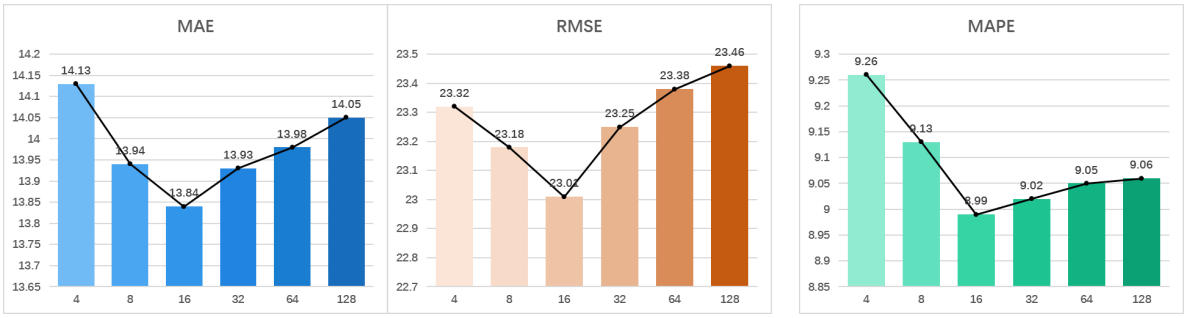
(a)

(b)



(c)

Figure 4: Component analysis of LGA on Electricity datasets. Each of the components in LGA contributes a positive effect. Among them, Global-GAT is the component that influences the model most.



(a) (b) (c)

Figure 5: Effect of the number of embedding dimension on model performance. Increasing the number of embedding dimension from 4 to 128 shows that LGA has the best performance in the 16.

5.6. Hyperparameter study

To further investigate the impact of hyperparameter settings, we conducted parameter experiments on the PEMS08 dataset. The hyperparameter associated with the model structure is the embedding dimension F of the node embedding matrix. As shown in Figure 5, the numerical value of the embedding dimension affects LGA performance. The model works best at $F = 16$. When the mbedding dimensions $F < 16$ and $F > 16$, the performance of LGA will gradually deteriorate.

6. Conclusion

In this paper, we propose a new neural network structure, a local global attention based (LGA) for modeling and predicting the traffic flow data. Traffic flow data has significant periodicity and correlation between sensors. Therefore, we propose an attention-based model for extracting periodic features of data, which can adaptively learn the periodic correlation between sensors. LGA uses weighted attention to fuse the different periodic characteristics in the data to emphasize the periodicities. Extensive experiments on publicly available real-world traffic datasets and electricity dataset demonstrate the superiority of our model over state-of-the-art methods.

Acknowledgement

This research was supported by the National Natural Science Foundation of China (No.62062033).

References

[1] R. Jiang, D. Yin, Z. Wang, Y. Wang, J. Deng, H. Liu, Z. Cai, J. Deng, X. Song, and R. Shibasaki, "Dl-traff: Survey and benchmark of deep learning models for urban traffic prediction," in *Proceedings of the 30th ACM international conference on information & knowledge management*, 2021, pp. 4515–4525.

[2] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.

[3] A. Zonoozi, J.-j. Kim, X.-L. Li, and G. Cong, "Periodic-crn: A convolutional recurrent model for crowd density prediction with recurring periodic patterns." in *IJCAI*, 2018, pp. 3732–3738.

[4] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 5668–5675.

[5] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin, "Deepstn+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 1020–1027.

[6] 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.

[7] C. Zheng, X. Fan, C. Wang, and J. Qi, "Gman: A graph multi-attention network for traffic prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 01, 2020, pp. 1234–1241.
[8] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "Geoman: Multi-level attention networks for geo-sensory time series prediction." in *IJCAI*, vol. 2018, 2018, pp. 3428–3434.
[9] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," *Advances in neural information processing systems*, vol. 32, 2019.
[10] S.-Y. Shih, F.-K. Sun, and H.-y. Lee, "Temporal pattern attention for multivariate time series forecasting," *Machine Learning*, vol. 108, no. 8, pp. 1421–1441, 2019.
[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[12] 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, 2016.
[13] M. Welling and T. N. Kipf, "Semi-supervised classification with graph convolutional networks," in *J. International Conference on Learning Representations (ICLR 2017)*, 2016.
[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
[15] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Thirty-first AAAI conference on artificial intelligence*, 2017.
[16] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial-temporal network for taxi demand prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.
[17] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," in *The 28th International Joint Conference on Artificial Intelligence (IJCAI)*. International Joint Conferences on Artificial Intelligence Organization, 2019.
[18] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 753–763.
[19] C. C. Holt, "Forecasting seasonals and trends by exponentially weighted moving averages," *International journal of forecasting*, vol. 20, no. 1, pp. 5–10, 2004.
[20] T. Alghamdi, K. Elgazzar, M. Bayoumi, T. Sharaf, and S. Shah, "Forecasting traffic congestion using arima modeling," in *2019 15th international wireless communications & mobile computing conference (IWCMC)*. IEEE, 2019, pp. 1227–1232.
[21] N. Zhang, Y. Zhang, and H. Lu, "Seasonal autoregressive integrated moving average and support vector machine models: prediction of short-term traffic flow on freeways," *Transportation Research Record*, vol. 2215, no. 1, pp. 85–92, 2011.
[22] S. Shekhar and B. Williams, "Adaptive seasonal time series models for forecasting short-term traffic flow," *Transportation Research Record Journal of the Transportation Research Board*, vol. 2024, no. 2024, pp. 116–125, 2007.
[23] H. Sun and H. X. Liu, "Short term traffic forecasting using the local linear regression model," *Center for Traffic Simulation Studies*, 2002.
[24] H. Yan, L. Fu, Y. Qi, L. Cheng, Q. Ye, and D.-J. Yu, "Learning a robust classifier for short-term traffic state prediction," *Knowledge-Based Systems*, vol. 242, p. 108368, 2022.
[25] 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.
[26] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *International Conference on Learning Representations*, 2018.
[27] R. Huang, C. Huang, Y. Liu, G. Dai, and W. Kong, "Lsgcn: Long short-term traffic prediction with graph convolutional networks." in *IJCAI*, 2020, pp. 2355–2361.
[28] A. Salamanis, D. D. Kehagias, C. K. Filelis-Papadopoulos, D. Tzovaras, and G. A. Gravvanis, "Managing spatial graph dependencies in large volumes of traffic data for travel-time prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 6, pp. 1678–1687, 2015.
[29] W. Chen, L. Chen, Y. Xie, W. Cao, Y. Gao, and X. Feng, "Multi-range attentive bicomponent graph convolutional network for traffic forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 04, 2020, pp. 3529–3536.
[30] 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.
[31] 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.
[32] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.
[33] Z. Shao, Z. Zhang, F. Wang, W. Wei, and Y. Xu, "Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 4454–4458.
[34] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018, pp. 3634–3640.
[35] D. Cao, Y. Wang, J. Duan, C. Zhang, X. Zhu, C. Huang, Y. Tong, B. Xu, J. Bai, J. Tong *et al.*, "Spectral temporal graph neural network for multivariate time-series forecasting," *Advances in neural information processing systems*, vol. 33, pp. 17 766–17 778, 2020.

[36] J. Deng, X. Chen, R. Jiang, X. Song, and I. W. Tsang, "St-norm: Spatial and temporal normalization for multi-variate time series forecasting," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 269–278.