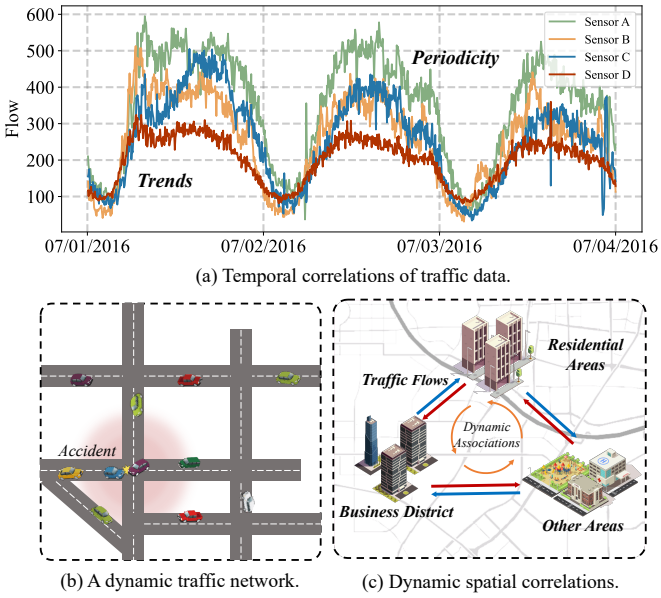
Spatial-Temporal Dynamic Graph Convolutional Network with Interactive Learning for Traffic Forecasting

Aoyu Liu and Yaying Zhang

***Abstract*—Accurate traffic forecasting is essential in urban traffic management, route planning, and flow detection. Recent advances in spatial-temporal models have markedly improved the modeling of intricate spatial-temporal correlations for traffic forecasting. Unfortunately, most previous studies have encountered challenges in effectively modeling spatial-temporal correlations across various perceptual perspectives and have neglected the interactive learning between spatial and temporal correlations. Additionally, constrained by spatial heterogeneity, most studies fail to consider distinct spatial-temporal patterns of each node. To overcome these limitations, we propose a Spatial-Temporal Interactive Dynamic Graph Convolutional Network (STIDGCN) for traffic forecasting. Specifically, we propose an interactive learning framework composed of spatial and temporal modules for downsampling traffic data. This framework aims to capture spatial and temporal correlations by adopting a perception perspective from the global to the local level and facilitating their mutual utilization with positive feedback. In the spatial module, we design a dynamic graph convolutional network based on graph construction methods. The network is designed to leverage a traffic pattern bank considering spatial-temporal heterogeneity as a query to reconstruct a data-driven dynamic graph structure. The reconstructed graph structure can reveal dynamic associations between nodes in the traffic network. Extensive experiments on eight real-world traffic datasets demonstrate that STIDGCN outperforms the state-of-the-art baseline while balancing computational costs. The source codes are available at https://github.com/LiuAoyu1998/STIDGCN.**

***Index Terms*—Interactive learning, dynamic graph construction, graph convolutional network, traffic forecasting.**



(a) Temporal correlations of traffic data.

(b) A dynamic traffic network.

(c) Dynamic spatial correlations.

Fig. 1. Spatial-temporal correlations of traffic data. (a) shows the complex temporal correlations of traffic data (periodicity, trends). (b) indicates different traffic patterns in the traffic network, such as accidents on the road that affect nearby roads. (c) shows the dynamic associations between the nodes and the spatial heterogeneity in the traffic network.

I. Introduction

With the help of available massive urban traffic data collected from sensors on the road, cabs, private car trajectories, and transaction records of public transportation, big traffic data analysis has become an indispensable part of smart city development for traffic planning, control, and condition assessment [1], [2]. Traffic forecasting, which aims to predict the urban dynamics with the observed historical traffic data, is critical for traffic services like flow control, route planning, and flow detection. Accurate traffic forecasting can reduce road congestion, facilitate city traffic network management, and improve transportation efficiency [3].

Traffic forecasting has remained a prominent research focus for several decades. Numerous dedicated research endeavors have been undertaken to enhance forecasting performance. Nevertheless, the field continues to face specific challenges. These challenges arise mainly from traffic data's complex temporal dependencies and dynamic spatial correlations. First, traffic data exhibits distinct periodicity and trends as a type of multivariate time series data. This can be illustrated in Fig. 1(a), demonstrating the traffic flow of three observation points on three consecutive days. The complex temporal correlation of traffic data also makes long-range forecasting difficult. Furthermore, traffic flow in one area is dynamically correlated with other areas in the traffic network over time, which we refer to as the dynamic associations between nodes. For instance, a traffic accident in one area affects the traffic conditions in its nearby areas, as shown in Fig. 1(b). Finally, spatial heterogeneity and dynamic associations between nodes in a traffic network are hard to capture to uncover the spatial correlations of traffic data. Here, spatial heterogeneity means that different areas (*e.g.*, residential areas and business districts) have different characteristics [4], such as road types, road width, POIs, etc., as shown in Fig. 1(c). Under the

This work was supported in part by the Shanghai Science and Technology Innovation Action Plan Project under Grant 22511100700. (Corresponding author: Yaying Zhang.)

Aoyu Liu and Yaying Zhang are with the Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 201804, China. (email: {liuaoyu, yaying.zhang}@tongji.edu.cn).

influence of spatial heterogeneity, different areas gradually develop distinct traffic patterns over time. All of the above make traffic forecasting tremendously challenging.

In response to the challenges above, more deep learning-based methods are replacing traditional approaches in traffic forecasting [4]. These methods typically use the temporal modules such as Recurrent Neural Networks (RNNs) [5], [6], [7], Temporal Convolution Networks (TCNs) [8], [9], [10], and attention mechanisms [11], [12], [13] to model temporal correlations in traffic data. Additionally, Graph Neural Networks (GNNs), such as Graph Convolutional Networks (GCNs) [14], [15], [16] and Graph Attention Networks (GATs) [17], [18], [19], are used as the spatial modules for capturing spatial correlations in traffic data. Despite significant achievements in modeling spatial-temporal correlations, two limitations of existing methods persist:

1) *Failing to model spatial-temporal correlations from multiple perspectives and neglect the interactive learning between spatial and temporal correlations.* Some studies employ a sequential [14], [20], [15] or parallel [11], [21], [17] approach by combining spatial and temporal modules, capturing spatial and temporal correlations separately. This represents a coarse-grained spatial-temporal modeling paradigm that treats traffic signals across all input time steps as a whole. This approach may weaken valuable features, amplify unimportant ones, and lacks modeling for local spatial-temporal correlations. Other studies [6], [22], [7] aim to maximize the preservation of the coupling between spatial and temporal correlations by embedding spatial modules into temporal modules, *e.g.*, embedding GCN into RNN. This constitutes a fine-grained spatial-temporal modeling paradigm for separately capturing traffic signals' temporal and spatial correlations at each time step. Nevertheless, it lacks the awareness of global spatial-temporal representations, making it challenging to capture long-range spatial-temporal dependencies. Additionally, effectively modeling temporal trends and periodicity contributes to the model's ability to reveal dynamic spatial associations within the traffic network and vice versa. Thus, the interactive learning between temporal and spatial representations can be mutually reinforcing and yield positive feedback. However, most studies simply fuse captured spatial and temporal correlations, ignoring the advantages of interactive learning.

2) *Failing to effectively consider each node's distinct spatial-temporal patterns due to spatial heterogeneity when constructing dynamic graph structures.* STGNNs are required to construct graph structures to facilitate message passing within traffic networks, enabling information aggregation among nodes. The graph structures constructed by existing methods can be categorized into three types: pre-defined graph structures, adaptive graph structures, and dynamic graph structures. Pre-defined graph structures are constructed based on a priori knowledge (*e.g.*, geographic adjacencies or distances between nodes) [14], [6], [8], reflecting only superficial spatial relationships between nodes and incapable of modeling their latent spatial associations. The adaptive graph structure parameterizes spatial associations between nodes by defining trainable embeddings [9], [23], [24], enabling the adaptive learning of latent spatial associations between nodes. However, both the pre-defined and adaptive graph structures are static models that cannot describe dynamic associations between nodes because they do not dynamically adjust to input changes during the testing phase. Recent research [22], [25], [19] trends focus on building data-driven dynamic graph structures. In these studies, graph structures adjust in response to changes in the input spatial-temporal data, thereby enhancing their ability to capture evolving spatial correlations over time. While existing data-driven dynamic graph construction methods have shown success, these studies fail to fully consider the distinct spatial-temporal patterns exhibited by each node due to spatial heterogeneity. The traffic pattern at each node is not only affected by the real-time traffic flow in other areas but also by spatial heterogeneity, such as special traffic patterns resulting from regional or road differences.

To address the first limitation, we propose a spatial-temporal interactive learning strategy that models the spatial-temporal correlations by capturing the changing perspectives from global to local. This strategy not only preserves the capturing capability for local representations but also captures long-range spatial-temporal dependencies. Specifically, we introduce the Spatial-Temporal Interaction (STI) module, which is composed of temporal and spatial modules. The STI module divides the traffic sequence into two sub-sequences as interactive learning objects based on time intervals, utilizing sequences' periodicity and trend characteristics. During the spatial-temporal interactive learning process, the captured temporal correlation enhances the construction of the dynamic graph structure, improving the ability to capture spatial correlation. This facilitates the capture of deeper temporal correlations, creating a positive feedback mechanism.

To address the second limitation, we propose a Dynamic Graph Neural Network (DGCN) module based on the dynamic graph construction method. Within the DGCN module, we introduce a pattern bank that stores distinct traffic patterns at each node, accounting for spatial heterogeneity. In the DGCN module, we propose a pattern bank for storing distinct traffic patterns at each node due to spatial heterogeneity. Through the data-driven approach, the DGCN module uses the input as a query, calculating spatial similarity with both the pattern bank and input representations to construct the dynamic fusion graph structure. The generated dynamic fusion graph structure will be used as auxiliary information to enhance the capture of dynamic spatial correlations.

In the end, the model proposed for the limitations above is named the ***S**patial-**T**emporal **I**nteractive **D**ynamic **G**raph **C**onvolutional **N**etwork* (STIDGCN). The contributions of our work are summarized as follows:

- A spatial-temporal interactive learning strategy is designed to model spatial-temporal correlations from multiple perspectives. Interactive learning between temporal and spatial correlations forms a positive feedback mechanism for representation mining.
- A data-driven dynamic graph construction-based approach is designed to model dynamic associations in traffic networks. The graph construction method fully considers the distinct spatial-temporal patterns each node presents.

- Extensive experiments are conducted on eight real-world datasets from previous work. The experimental results show that STIDGCN has state-of-the-art performance while balancing the computational costs compared to baseline models.

II. Related Work

In earlier times, classical statistical methods were used for traffic forecasting, such as Historical Averages (HA), Auto-Regressive Integrated Moving Average (ARIMA) [26], and Vector Auto-Regressive (VAR) [27]. However, these are linear time series-based methods that rely on static assumptions. Since traffic data are complex nonlinear data, these methods naturally underperform compared to machine learning-based methods. To capture complex nonlinear relationships in traffic data, some traditional machine learning methods are applied to traffic forecasting, such as Support Vector Regression (SVR) [28], Random Forest Regression (RFR) [29], and K-Nearest Neighbor (KNN) [30]. These methods are more effective but require specific experience to design manual features.

Deep learning-based methods are effective in automatically capturing features for representation learning. Early machine learning methods used RNNs-based methods (including LSTM and GRU) [31], [18] to capture traffic data's temporal correlations. RNN-based methods have limitations, such as error accumulation, slow training, and the inability to handle long sequences. Convolutional neural networks (CNNs), on the other hand, process data in parallel and require relatively low memory usage. Consequently, some CNN-based methods are widely used for time series[32], [33]. Recently, SCINet [34] expands the receptive field of convolutional operations and achieves multi-resolution analysis in a downsample-convolve-interact manner. However, the abovementioned methods do not consider the spatial dimension when modeling traffic data. To address this limitation and capture spatial correlations within traffic data, some studies have used CNNs for spatial correlation capture. In these studies, CNN treats traffic data as Euclidean data, *i.e.*, the traffic network is divided into grids to predict traffic conditions within each grid [35], [36], [37]. Practically, the real-world traffic networks possess unique topologies, rendering traffic data essentially non-Euclidean. Therefore, the spatial correlations captured by CNN-based methods are limited.

Recently, spatial-temporal graph neural networks (STGNNs) have been widely used to capture traffic data's spatial correlations. Typically, these methods incorporate RNN-based [6], [22], [7], CNN-based [9], [8], [21], and attention mechanism-based [11], [38], [12] temporal modules to capture temporal correlations in spatial-temporal modeling. The GNN-based [14], [46] and attention mechanism-based [11], [19], [13] spatial modules are used to capture spatial correlations, which can model non-Euclidean data and is suitable for capturing spatial correlations. As shown in Table I, we count representative STGNNs, categorize them according to their learned graph relation states, and list in detail the methods they employ in the spatial and temporal modules.

TABLE I
Classification of spatial-temporal modeling methods.

Model	Publication Title	Temporal Moudle	Spatial Moudle	Grpah Relations
DCRNN [6]	ICLR 2018	RNN	GNN	Static
STGCN [14]	IJCAI 2018	CNN	GNN	Static
ASTGCN [38]	AAAI 2019	CNN+Attention	GNN+Attention	Static
GWN [9]	IJCAI 2019	TCN	GCN	Static
MTGNN [10]	KDD 2020	TCN	GCN	Static
GMAN [11]	AAAI 2020	Attention	Attention	Static
STSGCN [39]	AAAI 2020	GCN	GCN	Static
AGCRN [23]	NeurIPS 2020	RNN	GCN	Static
STGODE [40]	KDD 2021	TCN	ODE	Static
STFGNN [21]	AAAI 2021	CNN	GCN	Static
STG-NCDE [41]	AAAI 2022	NCDE	GCN+NCDE	Static
STJGCN [42]	TKDE 2023	TCN	GCN	Static
ASTGNN [12]	TKDE 2021	Attention	GCN+Attention	Dynamic
DSTAGNN [25]	ICML 2022	CNN+Attention	GCN+Attention	Dynamic
D^{2}STGNN [43]	VLDB 2022	RNN+Attention	GCN	Dynamic
RAHRA [20]	TITS 2022	Attention	GCN	Dynamic
FSTL [44]	TITS 2022	CNN	GCN	Dynamic
MVSTT [17]	TC 2022	CNN+Attention	GCN+Attention	Dynamic
STTGCN [45]	TITS 2023	CNN	GCN	Dynamic
TinT [13]	IF 2023	Attention	GCN+Attention	Dynamic
ADCT-Net [19]	IF 2023	Attention	Attention	Dynamic
MegaCRN [7]	AAAI 2023	RNN	GCN	Dynamic

DCRNN [6] and STGCN [14] were among the first studies to combine spatial and temporal modules to model spatial-temporal correlations. DCRNN models traffic data's spatial correlations as a diffusion process on directed graphs, and it uses GRU in combination with diffusion GCN for traffic forecasting. STGCN employs convolution operation fully in the time dimension and uses spectral graph convolution to capture traffic data's spatial correlations. Since urban traffic is a dynamically changing system, the pre-defined graph structure with a static adjacency matrix cannot represent such dynamics. To this end, models represented by GWN[9], AGCRN [23], and STJGCN [42] capture the dynamic spatial correlations by designing an adaptive adjacency matrix with GCN. As the model training stops, the adaptive learnable matrices are fixed, so they do not describe dynamic spatial correlations over time. Therefore, these pre-defined graph structures based on adjacency relationships and adaptive graph structures relying on trainable embeddings describe static traffic networks and struggle to adequately capture the real-time dynamic associations between nodes. To overcome this limitation, as shown in Table I, some data-driven dynamic graph generation methods generate local-period dynamic graph structures by utilizing the input traffic data. For example, DSTAGNN [25] introduces a novel dynamic spatial-temporal aware graph based on a data-driven strategy to replace the traditionally static graph used in graph convolution methods. D^{2}STGNN [43] employs a novel decoupled spatial-temporal framework to separate diffusion and intrinsic traffic information and utilizes a dynamic graph learning module to learn the dynamic features of the traffic network. MegaCRN [7] introduces a meta-graph learner embedded into the encoder-decoder structure to address the spatial-temporal heterogeneity and non-stationarity implied in the traffic stream.

In the context of the remarkable success of Transformers in natural language processing and computer vision, as illustrated in Table I, the latest STGNNs tend to incorporate attention

mechanisms to model spatial-temporal correlations, especially in capturing long-range dependencies. ASTGCN [38], AST-GNN [12] combines the attention mechanism with GCN for modeling spatial-temporal correlations; the attention mechanism is mainly used to capture temporal correlations, and GCN is used to capture spatial correlations. GMAN [11] is an encoder-decoder structure consisting purely of spatial-temporal attention modules and simulates the influence of spatial-temporal factors on traffic conditions. To prevent receptive field bias, TinT [13] employs an innovative mixture of long and short-range information routing mechanisms, along with an original anisotropic graph aggregation designed for unbalanced traffic flow propagation. To capture enduring spatial-temporal dependencies and unveil latent graphic feature representations that surpass temporal and spatial constraints, ADCT-Net [19] leverages an adaptive dual-graphic method incorporating a cross-fusion strategy. Although attention-based STGNNs excel in effectiveness, they come with a significant computational burden.

Compared to previous works, our proposed STIDGCN offers the following advantages: *i)* STIDGCN employs a spatial-temporal interactive learning strategy with continuous down-sampling, capable of capturing spatial-temporal dependencies from global to local perspectives. This evolving perceptual horizon not only captures short-range dependencies effectively but also significantly enhances the model's capability to capture long-range dependencies. *ii)* Through our spatial-temporal interactive learning strategy, STIDGCN enables captured spatial and temporal correlations to leverage each other, exploring deeper spatial-temporal representations through positive feedback. *iii)* By querying the traffic pattern bank, STIDGCN can explore latent characteristics for each node and use them for dynamic graph construction. This allows STIDGCN to model deeper dynamic correlations between nodes while considering each node's distinct traffic patterns.

III. Preliminary

A. Definitions

1) ***Traffic Network****:* Real-world traffic networks can be described using spatial information obtained from internal sensor networks or by identifying between stations and road segments. A traffic network can be defined as an undirected graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E}, \mathbf{A})$, where $\boldsymbol{V}$ denotes the set of $|\boldsymbol{V}| = N$ nodes, and each node denotes an observation point (sensor or road segment) in the traffic network, and $\boldsymbol{E}$ is a set of edges. $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the adjacency matrix of graph $\boldsymbol{G}$, which represents the degree of association between nodes.

2) ***Traffic Signal****:* Traffic Signal $\boldsymbol{X}^{(t)} \in \mathbb{R}^{D \times N}$ denotes the data collected by all observation points in the traffic network $\boldsymbol{G}$ at time step t, where N denotes the number of nodes (N sensors), D denotes the initial number of feature channels (*e.g.*, the demand, volume or speed).

B. Problem Formalization

Traffic Forecasting. The traffic forecasting task is to predict the future traffic signal sequence $[\boldsymbol{Y}^{(t+1)}, \boldsymbol{Y}^{(t+2)}, \ldots, \boldsymbol{Y}^{(t+T')}]$ using a segment of historical sequence $[\boldsymbol{X}^{(t-T+1)}, \boldsymbol{X}^{(t-T+2)}, \ldots, \boldsymbol{X}^{(t)}]$, where T denotes the length of a given historical time series, T' denotes the length of the time series to be predicted. The traffic forecasting task can be defined as:

$$\left[\boldsymbol{X}^{(t-T+1)}, \ldots, \boldsymbol{X}^{(t)}\right] \xrightarrow{F} \left[\boldsymbol{Y}^{(t+1)}, \ldots, \boldsymbol{Y}^{\left(t+T'\right)}\right], \quad (1)$$

where F denotes the learning function from the historical sequence to the predicted sequence.

IV. Methodology

A. Overview Framework

The framework of the proposed STIDGCN is shown in Fig. 2 and follows an encoder-decoder architecture. The encoder employs a spatial-temporal interactive learning strategy to model the spatial-temporal correlations. Subsequently, the decoder is utilized to regressively predict future traffic signals by integrating the captured spatial-temporal representations.

First, the raw input data undergoes a data embedding layer to obtain higher-dimensional representations. Enhancing periodicity and trends in traffic data is achieved through internal adaptive temporal embedding. Subsequently, the high-dimensional representations are fed into a binary tree structure consisting of multiple STI modules. Each STI module employs multiple Temporal Sampling Convolution (TSConv) modules to learn complex temporal correlations. Simultaneously, we introduce a weight-sharing DGCN module to capture dynamic spatial correlations and implement a spatial-temporal interactive learning strategy for spatial-temporal feature interactions.

As shown in Fig. 2, the spatial-temporal interactive learning is implemented using a divide-and-conquer approach within each STI module. We divide the input sequence into two equal subsequences based on intervals. Next, the two halved-length subsequences are passed through the TSConv and DGCN modules to capture temporal and spatial correlations and perform informative interactions. In the DGCN, a graph generator is employed to reconstruct dynamic spatial associations among nodes in the traffic network for the present period. This dynamic spatial correlation is subsequently used as auxiliary information to unveil deeper spatial correlations. Through the spatial-temporal interactive learning strategy, this deeper spatial correlation feeds back into the captured temporal correlation, thereby mining more complex temporal correlations. This establishes a positive feedback spatial-temporal modeling paradigm, where spatial and temporal correlations mutually reinforce each other.

After all STI modules have been processed, multiple subsequence representations are generated. The complete spatial-temporal representation is formed by realigning these output subsequence representations based on their temporal positions. Finally, these spatial-temporal representations are passed through the Gated Linear Unit (GLU) to the regression layer, which outputs the final prediction.

B. Data Embedding Layer

The data embedding layer aims to transform the input into a high-dimensional representation to facilitate the exploration of

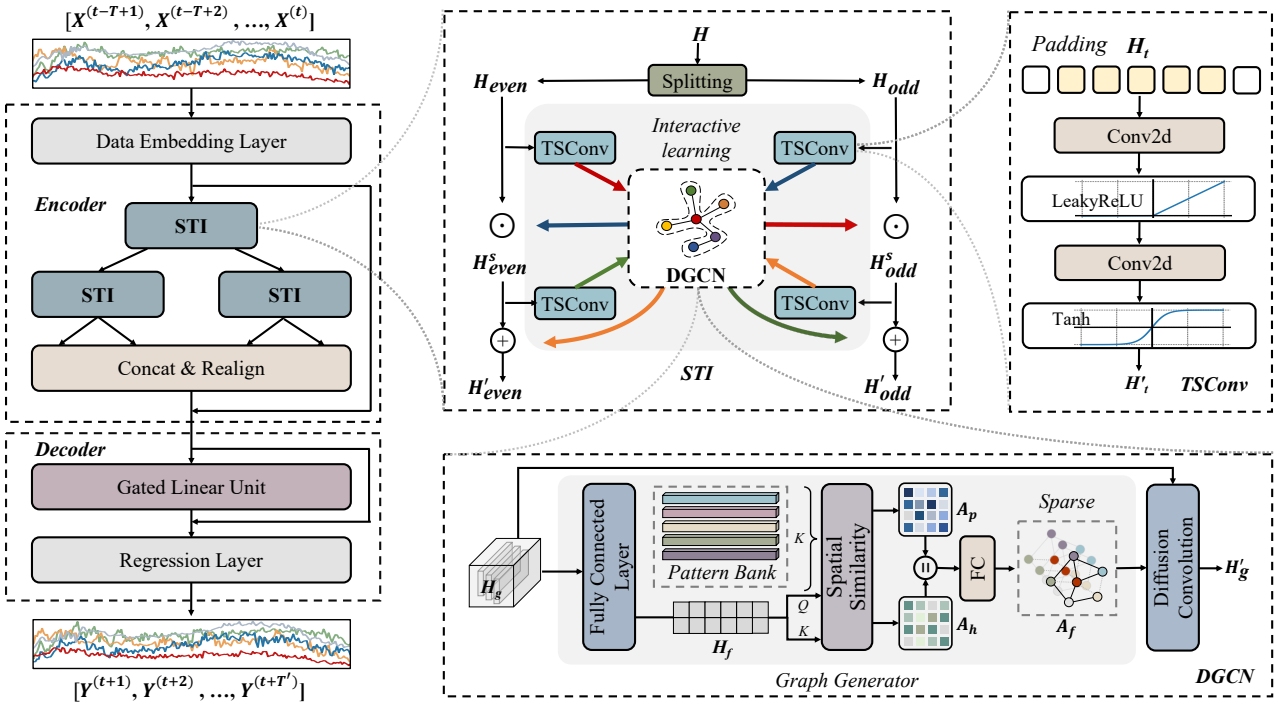


Fig. 2. The framework of STIDGCN. The STI module divides the input data by intervals and expands them downwards in a tree-like structure. The STI module combines multiple TSConvs and a DGCN with shared weights to enable spatial-temporal interactive learning of temporal and spatial correlations. The DGCN comprises a graph generator and a diffusion convolutional network on graphs. The DGCN first performs dynamic graph construction and then samples for spatial representations.

more complex spatial-temporal features. First, the raw traffic data $\boldsymbol{X} \in \mathbb{R}^{D\times N\times T}$ is transformed through a fully connected layer into $\boldsymbol{X}_{data} \in \mathbb{R}^{d_1\times N\times T}$, where d_1 represents the number of expanded feature channels. Subsequently, we introduce a learnable adaptive temporal periodic embedding [47] to incorporate prior temporal knowledge into the model.

Traffic data exhibits periodicities and trends, such as peak hours, weekends, and weekdays. Consequently, we have applied absolute positional encoding to the "day" and "week" positions of each traffic data, resulting in $\boldsymbol{T}_{day} \in \mathbb{R}^{N\times T}$ and $\boldsymbol{T}_{week} \in \mathbb{R}^{N\times T}$. These encodings encapsulate prior temporal knowledge of positions that possess periodicities and trends. To facilitate adaptive learning of the periodicities and trends within traffic data, we have similarly defined two trainable temporal embeddings, namely $\boldsymbol{W}_{day} \in \mathbb{R}^{N\times f}$ and $\boldsymbol{W}_{week} \in \mathbb{R}^{N\times 7}$. The f in $\boldsymbol{W}_{day}$ denotes the number of data that can be collected by a sensor per day. For instance, with a sampling interval of 5 minutes per day, f is 288. Subsequently, from each traffic data instance, we extract the time encoding as the index and then retrieve the corresponding temporal embedding tensors, denoted as $\boldsymbol{E}_{day} \in \mathbb{R}^{d_2\times N}$ and $\boldsymbol{E}_{week} \in \mathbb{R}^{d_2\times N}$, from the matrices $\boldsymbol{W}_{day}$ and $\boldsymbol{W}_{week}$. The trainable temporal periodic embedding $\boldsymbol{E}_t \in \mathbb{R}^{d_2\times N}$ can be obtained by summing the temporal embedding tensors:

$$\boldsymbol{E}_t = \boldsymbol{E}_{day} + \boldsymbol{E}_{week}. \tag{2}$$

Finally, concatenating $\boldsymbol{E}_t$ with the high-dimensional feature representation $\boldsymbol{X}_{data}$ yields the output $\boldsymbol{X}_{emb} \in \mathbb{R}^{C\times N\times T}$ of the data embedding layer:

$$\boldsymbol{X}_{emb} = \boldsymbol{X}_{data} \parallel \boldsymbol{E}_t, \tag{3}$$

where C represents the number of feature channels. We employ broadcasting to ensure computational operations when dimensions are not exactly matched. For convenience, we use $\boldsymbol{H} \in \mathbb{R}^{C\times N\times T}$ to replace $\boldsymbol{X}_{emb}$ in the subsequent text.

C. Spatial-Temporal Interaction

Traffic data is a type of multivariate time series data that exhibits apparent periodicity and trend information along the time dimension. The periodicity and trend information are crucial for exploring deep-seated temporal correlations. Due to these characteristics, current traffic data is correlated with both past and future traffic data. When capturing temporal correlation, some methods utilize only sequence causal relationships, such as RNN-based and TCN-based temporal modules. This implies that the current data is only relevant to its past data. However, the current data is correlated with both past and future data, so leveraging the global relationship of the sequence should be employed to enhance the capture of temporal correlation. Inspired by previous work [34], [48], we propose the STI module for spatial-temporal modeling, leveraging global relationships within sequences. The STI module comprises several temporal modules capturing temporal correlations, a spatial module capturing spatial correlations, and a spatial-temporal interactive learning strategy. This strategy involves the interaction between the temporal and spatial modules,

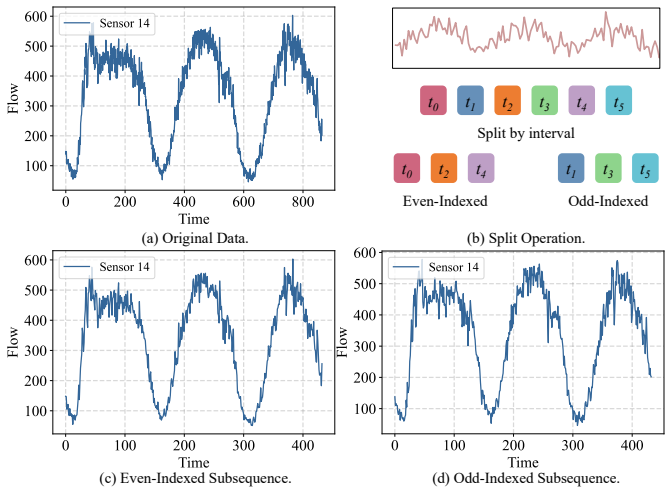


Fig. 3. Impact of splitting operations on traffic data. (a) represents the original sequence, while (c) corresponds to the subsequence generated by dividing according to the even index, and (d) corresponds to the subsequence generated by dividing according to the odd index. The subseries obtained after the interval division operation still retain the periodicity and trend information of the original series.

exchanging information to facilitate the capture of temporal and spatial correlations.

As shown in Fig. 3, we visualized the three-day traffic original sequence and interval split subsequence from sensor 14 in the PEMS08 dataset. Compared with the original sequence, the subsequences have preserved both the trend and periodicity. As shown in Fig. 2, in the STI module, we use the above characteristics to split the input traffic sequence according to intervals, generating two subsequences indexed by parity positions. The two subsequences preserve periodicity and trend and serve as two objects for the spatial-temporal interactive learning process. The two subsequences can leverage each other's information through continuous interaction and mutual learning. This process maximizes the utilization of global relationships within the original sequences, thereby enhancing the spatial-temporal representation mining capabilities of both the temporal and spatial modules. By downsampling raw sequences through multiple STI modules and forming a binary tree structure, STIDGCN can capture spatial-temporal correlations ranging from global to local perception granularity. This spatial-temporal modeling paradigm enables the modeling of long-range spatial-temporal dependencies, expanding the range of application scenarios for the model.

1) Temporal Sampling Convolution: In the STI module, we propose the TSConv module as the temporal module to capture temporal correlation. As shown in Fig. 2, four TSConv modules are employed within the STI module to capture temporal correlations. The TSConv utilizes two layers of 2D-CNNs, each using kernel sizes of (1, $s1$) and (1, $s2$), where $s1$ and $s2$ represent the set kernel sizes. The TSConv modules only operate on the temporal dimension of traffic data. The operation in TSConv can be defined as follows:

$$\boldsymbol{H}'_t = \text{Conv2d}(\text{Conv2d}(\text{Padding}(\boldsymbol{H}_t))), \tag{4}$$

where $\boldsymbol{H}_t$ and $\boldsymbol{H}'_t$ are hidden representations in TSConv, we have omitted the formula for the activation function here. The temporal correlation of a single sequence is effectively and efficiently modeled in parallel through a two-layer convolution operation.

2) Dynamic Graph Convolutional Neural Network: We design a weight-sharing DGCN module as a spatial module to capture spatial correlations dynamically. The DGCN implements two main functions: dynamic graph construction and spatial correlation capture. To fulfill these functions, DGCN specifically includes a graph generator and a diffusion-based GCN.

In contrast to previous graph construction methods, the graph construction method employed in the graph generator takes into account the distinct traffic patterns of each node and dynamic associations between nodes. Specifically, as shown in Fig. 2, we define a trainable traffic pattern bank $\boldsymbol{\Phi} \in \mathbb{R}^{C\times N}$ that can adaptively store unique traffic patterns for each node due to spatial heterogeneity during the training process. The graph generator utilizes the traffic pattern bank $\boldsymbol{\Phi}$ and the hidden representation $\boldsymbol{H}_g \in \mathbb{R}^{C\times N\times t'}$ input to the DGCN, constructing a fusion adjacency matrix that dynamically evolves over time. This matrix models the dynamic associations between nodes arising from traffic changes. The graph generator employs a data-driven dynamic graph construction approach. This allows the graph structure to be dynamically adjusted based on various traffic data, ensuring that the reconstructed graph accurately reflects the dynamic correlations of traffic changes.

In the graph generator, the representation $\boldsymbol{H}_g$, acquired by capturing temporal correlations through the TSConv module, is first nonlinearly mapped using a fully connected layer to obtain the hidden representation $\boldsymbol{H}_f \in \mathbb{R}^{C\times N}$ that eliminates the temporal dimension, which can be defined as follows:

$$\boldsymbol{H}_f = \text{FC}\left(\sum_{t=1}^{t'} \boldsymbol{H}_g[\ldots, t]\right) \tag{5}$$

Next, the representation $\boldsymbol{H}_f$ will be used as a query to perform spatial similarity with the pattern bank and itself, respectively, i.e., to compute the degree of correlation between nodes. This process generates two dynamic adjacency matrices, $\mathbf{A}_p \in \mathbb{R}^{N\times N}$ and $\mathbf{A}_h \in \mathbb{R}^{N\times N}$, respectively. The generation process of these two dynamic adjacency matrices can be defined as:

$$\mathbf{A}_p = \frac{\exp\left(\boldsymbol{H}_f^\top[\ldots, i]\boldsymbol{\Phi}[\ldots, j]\right)}{\sum_{j=1}^{N}\exp\left(\boldsymbol{H}_f^\top[\ldots, i]\boldsymbol{\Phi}[\ldots, j]\right)}, \tag{6}$$

$$\mathbf{A}_h = \frac{\exp\left(\boldsymbol{H}_f^\top[\ldots, i]\boldsymbol{H}_f[\ldots, j]\right)}{\sum_{j=1}^{N}\exp\left(\boldsymbol{H}_f^\top[\ldots, i]\boldsymbol{H}_f[\ldots, j]\right)}, \tag{7}$$

where i denotes the row index. Though the pattern bank is not directly associated with the input data, with the training, the pattern bank can adaptively adjust and store the unique traffic patterns of each node due to the spatial heterogeneity using different input information. Stored traffic patterns can overcome the limitations of reconstructing dynamic graphs solely based

on local period inputs, such as traffic signal data from the past hour. Thus, $\mathbf{A}_p$ contains latent spatial associations between nodes. On the other hand, $\mathbf{A}_h$ performs similarity calculations with itself and yields dynamic associations between nodes in the current period. These two dynamic adjacency matrices reflect the dynamic spatial associations among nodes from different perspectives.

These two dynamic adjacency matrices undergo a concatenation operation. They are then fused by a fully connected layer, resulting in the dynamic fusion adjacency matrix $\mathbf{A}_f \in \mathbb{R}^{N\times N}$:

$$\mathbf{A}_f = \mathrm{FC}(\mathbf{A}_p \parallel \mathbf{A}_h), \tag{8}$$

$\mathbf{A}_f$ refers to modeling dynamic associations between nodes in the traffic network. Optimizing the graph structure's sparsity enhances the model's computational efficiency and robustness [10]. To achieve this, we introduce a mask matrix $\mathbf{M} \in \mathbb{R}^{N\times N}$, to sparsify the existing dynamic adjacency matrix $\mathbf{A}_f$. Specifically, we construct the $\mathbf{M}$ by indexing each node with its most relevant neighbors based on their maximum relevance. The sparsification operation is defined as follows:

$$\mathbf{M}_{ij} = \begin{cases} 1, & \mathbf{M}_{ij} \in \mathrm{TopK}\left(\mathbf{M}_{i*}, \tau\right) \\ 0, & \text{otherwise} \end{cases}, \tag{9}$$

$$\mathbf{A}_f = \mathbf{A}_f \odot \mathbf{M}, \tag{10}$$

where τ is the threshold of the TopK function and denotes the max number of neighbors of a node.

Finally, we use $\mathbf{A}_f$ as auxiliary information and employ a diffusion-based GCN to capture dynamic spatial correlations. The diffusion-based GCN treats dynamic changes in the traffic network as a diffusion process. It aims to aggregate information about the diffusion process between nodes on the graph, where the diffusion signal at a target node depends on the recent values of its neighboring nodes. The diffusion-based GCN can be defined as follows:

$$\begin{aligned} \boldsymbol{H}'_g &= \mathrm{GCN}\left(\boldsymbol{H}_g, \mathbf{A}_f\right) \\ &= \sum_{k=0}^{K} \mathbf{A}_f^k \boldsymbol{H}_g \boldsymbol{W}, \end{aligned} \tag{11}$$

where $\boldsymbol{H}'_g \in \mathbb{R}^{C\times N\times t'}$ represents the output of DGCN and k denotes the diffusion steps.

DGCN first considers the latent traffic patterns of nodes due to spatial-temporal heterogeneity and the dynamic correlations among nodes as two perspectives for constructing a dynamic fusion graph structure and then utilizing this graph structure to capture features, thereby modeling dynamic spatial correlations. It is important to note that STIDGCN performed dynamic graph construction in each STI module. This means STIDGCN can model dynamic associations at different global and local granularities to explore deeper dynamic spatial correlations.

3) Spatial-Temporal Interactive Learning: The spatial-temporal interactive learning strategy is a temporal and spatial feature interaction between two subsequences. Assuming that $\boldsymbol{H} \in \mathbb{R}^{C\times N\times T}$ represents the input to the STI module. The subsequences of $\boldsymbol{H}$ obtained after interval division (based on parity index intervals) can be denoted as $\boldsymbol{H}_{even} \in \mathbb{R}^{C\times N\times T/2}$ and $\boldsymbol{H}_{odd} \in \mathbb{R}^{C\times N\times T/2}$. The TSConv modules within the STI module are labeled TSConv1, TSConv2, TSConv3, and TSConv4. The output after the first round of interactive learning consists of $\boldsymbol{H}^s_{even} \in \mathbb{R}^{C\times N\times T/2}$ and $\boldsymbol{H}^s_{odd} \in \mathbb{R}^{C\times N\times T/2}$ (as indicated by the red and blue arrows in Fig. 2). These sequences, $\boldsymbol{H}^s_{even}$ and $\boldsymbol{H}^s_{odd}$, will undergo additional interactive learning in the form of feedback. The resulting final subsequences are denoted as $\boldsymbol{H}'_{even} \in \mathbb{R}^{C\times N\times T/2}$ and $\boldsymbol{H}'_{odd} \in \mathbb{R}^{C\times N\times T/2}$ (as indicated by the green and orange arrows in Fig. 2). The operations within the STI module can be defined as follows:

$$\boldsymbol{H}_{even}, \boldsymbol{H}_{odd} = \mathrm{Split}(\boldsymbol{H}) \tag{12}$$

$$\boldsymbol{H}^s_{even} = \tanh\left(\mathrm{DGCN}\left(\mathrm{TSConv1}\left(\boldsymbol{H}_{odd}\right)\right)\right) \odot \boldsymbol{H}_{even}, \tag{13}$$

$$\boldsymbol{H}^s_{odd} = \tanh\left(\mathrm{DGCN}\left(\mathrm{TSConv2}\left(\boldsymbol{H}_{even}\right)\right)\right) \odot \boldsymbol{H}_{odd}, \tag{14}$$

$$\boldsymbol{H}'_{even} = \boldsymbol{H}^s_{even} + \tanh\left(\mathrm{DGCN}\left(\mathrm{TSConv3}\left(\boldsymbol{H}^s_{odd}\right)\right)\right), \tag{15}$$

$$\boldsymbol{H}'_{odd} = \boldsymbol{H}^s_{odd} + \tanh\left(\mathrm{DGCN}\left(\mathrm{TSConv4}\left(\boldsymbol{H}^s_{even}\right)\right)\right), \tag{16}$$

where $\odot$ denotes Hadamard product, $\tanh$ is the activation function. This spatial-temporal interactive learning strategy allows the subsequences to capture each other's spatial-temporal features in a mutual feedback manner. STIDGCN can learn the representations of sequences at a higher resolution by increasing the number of STI module layers to handle longer sequence data. However, deeper layers will also lead to increased memory usage. Our empirical study demonstrates that achieving excellent performance is possible with just two to three layers in most cases. Finally, we rearrange the output of the bottom STI module into the original temporal order, resulting in a complete spatial-temporal representation. This representation is then fused with the original input sequence through residual connections to obtain the output $\boldsymbol{H}_e \in \mathbb{R}^{C\times N\times T}$ of the encoder.

Overall, the spatial-temporal interactive learning strategy aims to perform spatial-temporal information sharing and mutual feedback between sequence data. The shared temporal features are used to optimize the generation of dynamic fusion graph structures to explore deeper spatial correlations. The effective exploration of spatial correlations feeds back to the mining of temporal correlations, thus forming positive feedback.

D. Decoder

As shown in Fig. 2, the $\boldsymbol{H}_e$ obtained from the encoder are fed into a GLU. Compared to the conventional Multi-Layer Perceptron (MLP), the GLU can leverage gating mechanisms to select and fuse spatial-temporal representations [49]. In this context, GLU is employed for the fusion and exploration of global spatial-temporal representations while also performing representation filtering and correction. The regression layer uses a fully connected layer to further process the representations output by GLU to obtain the final forecasting results. The operations within the decoder can be defined as follows:

$$\boldsymbol{Y} = \mathrm{FC}(\mathrm{ReLu}(\mathrm{FC}(\boldsymbol{H}_e) \odot \boldsymbol{\sigma}(\mathrm{FC}(\boldsymbol{H}_e)))), \tag{17}$$

where σ represents the sigmoid function and $\boldsymbol{Y} \in \mathbb{R}^{N \times T}$ denotes the traffic signal to be predicted. We employ a non-autoregressive approach to generate forecasting results to enhance computational efficiency and minimize error accumulation.

V. Experiment Results and Analysis

In this section, we conduct extensive experiments on eight real-world datasets to answer six research questions:

- **RQ1**: How does our proposed STIDGCN perform compared with the various state-of-the-art baselines?
- **RQ2**: How do the different components of STIDGCN affect its performance?
- **RQ3**: How do hyperparameters affect STIDGCN?
- **RQ4**: How is the robustness of STIDGCN?
- **RQ5**: How does STIDGCN perform in real cases?
- **RQ6**: How does the efficiency of STIDGCN compare to the baselines?

A. Experimental Settings

1) Datasets: We conduct extensive experiments on eight real-world traffic datasets, which include four highway traffic flow datasets [39] (**PEMS03**, **PEMS04**, **PEMS07**, **PEMS08**), two traffic demand datasets [50] (**NYCBike** and **NYCTaxi**) and two grid-based urban traffic datasets (**TDrive** [5] and **NYTaxi** [51]). Table II provides detailed statistics for these datasets. The highway traffic flow datasets were collected by Caltrans' Performance Measurement System (PEMS) [52], aggregated into 5-minute observations. The traffic demand datasets are collected from city traffic data. The NYCBike dataset comprises demand data for shared bikes used by residents from bike stations in New York City daily, while the NYCTaxi dataset contains data on taxicab trip records within New York City. Both of these datasets are aggregated at 30-minute observation intervals. Two grid-based urban traffic datasets cover taxi inflow and outflow data in Beijing and New York City.

On the highway traffic flow datasets and the traffic demand datasets, we use the data from the past 12 time horizons to predict the data from the next 12 time horizons (multistep forecasting) and split them into training, validation, and test sets in the ratio of 6:2:2. On the grid-based urban traffic datasets; we use data from the past six time horizons to predict data from the next one-time horizon (single-step forecasting) and divide them into training, validation, and test sets in the ratio of 7:1:2 to maintain consistency with the previous studies [53]. Additionally, we use Z-score normalization on all datasets to standardize the inputs.

2) Model Settings: Experiments are conducted under a computer environment with one Intel(R) Xeon(R) Gold 6230 CPU @ 2.10GHz and one NVIDIA Tesla V100 GPU card. We train our model using Ranger optimizer [54], and the initial learning rate is set to 0.001. The batch size is set to 64 for the highway traffic flow datasets and 16 for the traffic demand datasets. The training epoch is set to 300, and we use an early stop mechanism during training to ensure the model is over-optimized.

TABLE II
Statistics of the eight traffic datasets.

Dataset	Nodes	Granularity	Samples	Time range
PEMS03	358	5min	26208	09/01/2018-11/30/2018
PEMS04	307	5min	16992	01/01/2018-02/28/2018
PEMS07	883	5min	28224	05/01/2017-08/31/2017
PEMS08	170	5min	17856	07/01/2016-08/31/2016
NYCBike	250	30min	4368	04/01/2016-06/30/2016
NYCTaxi	266	30min	4368	04/01/2016-06/30/2016
TDrive	1024	60min	3600	02/01/2015-06/30/2015
NYTaxi	75	30min	17520	01/01/2014-12/31/2014

3) Evaluation Metrics: To evaluate the models' performance, we use the following evaluation metrics in the experiments: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). We select MAE as the loss function. Consistent with previous work [9], missing values in the highway traffic flow datasets and the traffic demand datasets are masked during both the training and testing phases. Additionally, samples with flow values below 10 for the grid-based urban traffic dataset are masked to ensure consistency with the previous works [53].

B. Baseline Methods

We compare STIDGCN with the following baselines.

1) Traditional models:

- **HA**: Historical Average uses the average results of historical data to predict future data.
- **VAR** [55]: Vector Auto-Regression is a time series model that captures traffic data's temporal correlations.
- **SVR** [56]: Support Vector Regression uses support vector machines to do regression on traffic sequences.

2) Models Designed for Garph-based Datasets:

- **DCRNN** [6]: This model is an encoder-decoder structure that combines diffusion GCN with GRU to capture traffic data's spatial-temporal correlations.
- **STGCN** [14]: This model combines the spectral GCN with 1D convolution to capture spatial-temporal correlations.
- **ASTGCN** [38]: This model captures spatial-temporal correlations by designing spatial and temporal attention mechanisms, respectively.
- **GWN** [9]: This model combines gated TCN with spatial GCN and proposes an adaptive adjacency matrix to learn dynamic spatial correlations.
- **MTGNN** [10]: This model learns spatial-temporal correlations through the mix-hop propagation layer in the spatial module, the dilated inception layer in the temporal module, and a more refined graph learning layer.
- **AGCRN** [23]: This model is a model that combines GCN with GRU using an adaptive graph structure.
- **GMAN** [11]: This model captures spatial-temporal correlations using multiple spatial-temporal attention modules.
- **STSGCN** [39]: This GCN model constructs multiple local spatial-temporal graphs to capture spatial-temporal correlations synchronously.

TABLE III
PERFORMANCE ON HIGHWAY TRAFFIC FLOW DATASETS. **BOLD**: BEST, UNDERLINE: SECOND BEST.

Method	PEMS03			PEMS04			PEMS07			PEMS08		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
HA	30.08	46.22	28.64%	38.51	55.75	28.21%	45.32	65.74	21.56%	31.99	46.49	20.28%
VAR	23.65	38.26	24.51%	24.54	38.61	17.24%	50.22	75.63	32.22%	19.19	29.81	13.10%
SVR	21.97	35.29	21.51%	28.70	44.56	19.20%	32.49	50.22	14.26%	23.25	36.16	14.64%
DCRNN	15.53	27.18	15.62%	19.63	31.26	13.59%	21.16	34.14	9.02%	15.22	24.17	10.21%
STGCN	15.65	27.31	15.39%	19.57	31.38	13.44%	21.74	35.27	9.24%	16.08	25.39	10.60%
ASTGCN	17.34	29.56	17.21%	22.93	35.22	16.56%	24.01	37.87	10.73%	18.25	28.06	11.64%
GWN	14.80	25.88	14.92%	18.54	30.09	12.71%	19.84	32.86	8.44%	14.54	23.67	9.41%
MTGNN	14.88	25.24	15.47%	18.96	31.05	13.65%	20.98	34.40	9.31%	15.12	24.23	9.65%
AGCRN	15.29	26.95	15.15%	19.83	32.26	12.97%	20.57	34.40	8.74%	15.95	25.22	10.09%
GMAN	16.52	27.18	17.36%	18.84	30.75	13.25%	20.97	34.20	9.05%	14.57	24.71	9.98%
STSGCN	17.48	29.21	16.78%	21.19	33.65	13.90%	24.26	39.03	10.21%	17.13	26.80	10.96%
STFGNN	16.77	28.34	16.30%	19.83	31.88	13.02%	22.07	35.80	9.21%	16.64	26.22	10.60%
STGODE	16.50	27.84	16.69%	20.84	32.82	13.77%	22.59	37.54	10.14%	16.81	25.97	10.62%
ASTGNN	14.78	25.00	**14.79%**	18.60	30.91	12.36%	20.62	34.00	8.86%	15.00	24.70	9.50%
DGCRN	14.80	25.94	15.04%	18.80	30.65	12.82%	20.48	33.25	9.06%	14.60	24.16	9.33%
STG-NCDE	15.57	27.09	15.06%	19.21	31.09	12.76%	20.53	33.84	8.80%	15.45	24.81	9.92%
DSTAGNN	15.57	27.21	14.68%	19.30	31.46	12.70%	21.42	34.51	9.01%	15.67	24.77	9.94%
D^2STGNN	14.88	26.01	15.12%	18.34	29.93	12.81%	19.68	33.19	8.43%	14.35	24.18	9.33%
MegaCRN	14.84	26.25	15.16%	18.70	30.52	12.76%	19.89	33.12	8.47%	14.68	23.68	9.53%
STIDGCN	**14.76**	**24.59**	15.28%	**18.16**	**29.77**	**12.24%**	**19.26**	**32.51**	**8.11%**	**13.45**	**23.28**	**8.77%**

- **STFGNN** [21]: This model efficiently learns hidden correlations by performing fusion operations on the generated spatial-temporal graphs.
- **STGODE** [40]: This model captures spatial-temporal dynamics through an ordinary differential equation.
- **DGCRN** [22]: This model employs a hypernetwork to extract dynamic features of node attributes and generate dynamic iterator parameters.
- **ASTGNN** [12]: This self-attention-based traffic forecasting model combines a time-trending self-attention mechanism with a defined dynamic GCN.
- **STG-NCDE** [41]: This model uses spatial-temporal NCDEs to process traffic data and is a controlled differential equation method.
- **DSTAGNN** [25]: This model learns a spatial-temporal graph and uses multi-head attention to represent dynamic spatial correlations.
- **D^2STGNN** [43]: This model identifies the diffusion process and inherent process in traffic data and further decouples them for spatial-temporal modeling.
- **MegaCRN** [7]: This model employs a meta-graph learner for spatial-temporal meta-graph learning.

3) Models Designed for Grid-based Datasets:

- **STResNet** [35]: This model uses the spatial-temporal properties of crowd flow using residual neural networks to predict traffic in different regions.
- **DMVSTNet** [53]: This model uses a framework that combines temporal, spatial, and semantic views to establish spatial-temporal correlations.
- **DSAN** [37]: This model uses Multi-Space Attention mechanisms to extract useful information from the traffic data.

C. Comparison and Analysis of Results (RQ1)

1) Overall Comparison: The experimental results on the three types of datasets are shown in Table III, IV, and V, respectively. The datasets used for the experiments cover a wide range of scenarios, including graph-based and grid-based traffic networks, multi-step traffic forecasting, single-step traffic forecasting, traffic flow forecasting, and demand forecasting. For all traffic datasets, the results are the average MAE, RMSE, and MAPE values for the prediction horizons. The experimental results show that the comprehensive performance of STIDGCN outperforms the baselines on all eight flow datasets, especially on the PEMS08 and TDrive datasets. Our proposed model demonstrates stable and excellent performance across various forecasting scenarios, showcasing a robust capability for extracting spatial-temporal representations.

The results in Table III indicate that the traditional methods HA, VAR, and SVR do not yield satisfactory performance. While these models account for temporal correlations, they overlook the intricate spatial correlations in traffic data. The earliest proposed STGNNs, such as DCRNN and STGCN, outperform traditional methods. This is attributed to their consideration of spatial correlation modeling and pioneers in employing pre-defined graph structures to enhance the capture of spatial correlations. The subsequent introduction of adaptive graph structure methods, such as GWN, MTGNN, and AGCRN, improves performance by utilizing trainable adaptive embeddings to model dynamic spatial correlations. However, these methods still describe static spatial correlations because the generation of graph structures cannot dynamically adapt as the input data changes. Some data-driven dynamic graph construction methods, such as DGCRN, D^2STGNN, and MegaCRN, outperform methods that statistically model spatial correlations. As evidenced by the results in Table III, IV,

TABLE IV
PERFORMANCE ON TRAFFIC DEMAND DATASETS. **BOLD**: BEST, UNDERLINE: SECOND BEST.

Method	NYCBike Drop-off			NYCBike Pick-up			NYCTaxi Drop-off			NYCTaxi Pick-up		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
DCRNN	1.96	2.94	51.42%	2.09	3.30	54.22%	5.19	9.63	37.78%	5.40	9.71	35.09%
STGCN	2.01	3.07	50.45%	2.08	3.31	53.12%	5.38	9.60	39.12%	5.71	10.22	36.51%
ASTGCN	2.79	4.20	69.88%	2.76	4.45	64.23%	6.98	14.70	45.48%	7.43	13.84	47.96%
GWN	1.95	2.98	50.40%	2.04	3.20	53.08%	5.03	8.78	35.63%	5.43	9.39	37.79%
MTGNN	1.94	2.91	50.47%	2.03	3.19	53.73%	5.02	8.76	37.62%	5.39	9.41	37.21%
AGCRN	2.06	3.19	51.91%	2.16	3.46	56.35%	5.45	9.56	40.67%	5.79	10.11	40.40%
GMAN	2.09	3.00	54.82%	2.20	3.35	57.34%	5.09	8.95	35.00%	5.43	9.47	34.39%
STSGCN	2.73	4.50	57.89%	2.36	3.73	58.17%	5.62	10.21	37.92%	6.19	11.14	39.67%
ASTGNN	2.24	3.35	57.21%	2.37	3.67	60.08%	6.28	12.00	49.78%	5.90	10.71	40.15%
DGCRN	1.96	2.93	51.99%	2.06	3.21	54.06%	5.14	9.39	35.09%	5.44	9.82	35.78%
STG-NCDE	2.28	3.42	60.96%	2.15	3.97	55.49%	5.38	9.74	40.45%	6.24	11.25	43.20%
D^2STGNN	1.92	2.90	51.94%	2.02	3.18	53.60%	5.01	8.74	35.81%	5.32	9.12	35.51%
MegaCRN	2.18	3.30	61.42%	2.31	3.59	67.07%	5.07	9.11	35.08%	5.47	9.96	35.13%
STIDGCN	**1.88**	**2.80**	**49.43%**	**2.00**	**3.11**	**51.71%**	**4.89**	**8.51**	**34.30%**	**5.15**	**8.90**	**33.74%**

TABLE V
PERFORMANCE ON TRAFFIC DEMAND DATASETS. **BOLD**: BEST, UNDERLINE: SECOND BEST.

Method	NYTaxi Inflow			NYTaxi Outflow			TDrive Inflow			TDrive Outflow		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
STResNet	14.49	24.05	14.54%	12.80	20.63	14.37%	19.64	34.89	17.83%	19.62	34.60	18.50%
DMVSTNet	14.38	23.73	14.31%	12.57	20.41	14.32%	19.60	34.48	17.68%	19.53	34.33	17.62%
DSAN	14.29	23.59	14.21%	12.46	20.29	14.27%	19.38	34.31	17.47%	19.29	34.27	17.38%
DCRNN	14.42	23.88	14.35%	12.82	20.07	14.34%	22.12	38.65	17.75%	21.76	38.17	17.38%
STGCN	14.38	23.86	14.21%	12.55	19.96	14.10%	21.37	38.05	17.54%	20.91	37.62	16.98%
ASTGCN	16.18	26.85	16.21%	13.85	22.75	15.42%	21.28	37.06	15.99%	22.49	39.48	17.02%
GWN	14.31	23.80	14.20%	12.28	19.62	13.69%	19.56	36.16	17.19%	19.55	36.20	15.93%
MTGNN	14.19	23.66	13.98%	12.27	19.56	13.65%	18.98	35.39	17.06%	18.93	35.99	15.76%
AGCRN	15.00	24.99	14.30%	12.87	21.07	14.14%	22.83	48.04	16.66%	23.22	49.85	16.36%
GMAN	14.28	23.73	14.11%	12.27	19.59	13.67%	19.24	35.99	17.11%	18.96	36.12	15.79%
STSGCN	15.60	26.19	15.20%	13.23	21.65	14.70%	23.83	41.19	18.55%	24.29	42.26	19.04%
ASTGNN	13.84	23.18	13.69%	12.11	19.20	13.60%	18.80	33.87	16.10%	18.79	34.00	15.58%
DGCRN	18.95	36.21	34.41%	16.56	33.28	29.85%	20.57	43.31	26.30%	20.50	42.79	27.07%
STG-NCDE	14.28	23.74	14.17%	12.28	19.61	13.68%	19.35	36.09	17.13%	19.23	36.14	15.87%
D^2STGNN	14.62	25.97	13.8 %	12.14	19.82	13.70%	20.36	35.81	16.40%	20.60	36.25	15.98%
MegaCRN	13.96	23.49	13.61%	12.38	20.18	13.77 %	17.48	31.88	14.37%	18.24	32.97	15.32%
STIDGCN	**13.71**	**23.05**	**13.10%**	**11.80**	**18.87**	**12.92%**	**17.05**	**29.79**	**13.76%**	**17.19**	**29.90**	**13.85%**

and V, dynamic modeling of spatial associations is a necessary component to ensure a competitive model.

Although differential equation-based methods allow for the continuous modeling of temporal correlations in traffic data, the performance of STGODE and STG-NCDE indicates their inferiority compared to recent STGNNs. Therefore, there is a need for further exploration of how to harness the latent differential equation-based methods fully. Notably, models based on the attention mechanism, such as GMAN and ASTGNN, also demonstrate strong performance. This is attributed to the attention mechanism's effective calculation of correlation degrees between different time steps and nodes, allowing for better capture of spatial-temporal dependencies over both short and long ranges.

In addition, in Table V, we include spatial-temporal models explicitly designed for grid-based data, such as ST-ResNet, DMVSTNet, and DSAN, for comparison with the STGNNs. The results reveal that some early STGNNs do not perform as well as those specifically crafted to model spatial-temporal correlations on grid-based data. However, recent studies, including ASTGNN and D^2STGNN, have surpassed the performance of these methods. Currently, more STGNNs are being applied to grid-based traffic networks.

Our proposed STIDGCN exhibits superior performance for the following reasons: *i*) STIDGCN uses a spatial-temporal interactive learning strategy to share spatial-temporal information between input sequence data and capture spatial-temporal correlations from global to local change granularity. In the spatial-temporal interactive learning process, the captured temporal correlations and the captured spatial correlations positively affect each other. This feedback mechanism allows STIDGCN to explore deeper spatial-temporal correlations. *ii*) By fully leveraging spatial-temporal information, we employ a dynamic graph construction method to simulate the dynamic associations between nodes in the traffic network. Simultaneously, we utilize a pattern bank to store valuable traffic patterns for each node, mitigating the impact of spatial heterogeneity. This graph structure learning approach enables STIDGCN to

TABLE VI
PERFORMANCE COMPARISON ON LONG-RANGE TRAFFIC FLOW FORECASTING. **BOLD**: BEST, UNDERLINE: SECOND BEST.

Dataset	Method	@Horizon 36			@Horizon 48			@Horizon 60			Average		
		MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
PEMS04	DCRNN	23.67	37.62	16.03%	24.47	38.86	16.81%	25.39	39.92	18.10%	22.79	36.19	15.70%
	STGCN	26.85	41.78	18.32%	27.27	42.40	18.74%	28.90	44.54	19.65%	26.66	41.58	18.30%
	GWNet	24.25	39.13	17.94%	24.32	39.29	18.01%	24.98	39.97	18.48%	24.26	39.20	17.78%
	MTGNN	23.18	37.12	15.63%	23.78	37.80	16.17%	25.11	39.16	17.65%	22.37	35.88	15.25%
	AGCRN	23.03	36.76	15.72%	23.47	37.65	16.38%	24.93	39.38	17.87%	22.29	35.58	15.72%
	GMAN	22.45	36.91	16.07%	22.72	39.13	16.33%	23.20	40.48	17.00%	22.26	35.66	15.93%
	ASTGNN	24.12	40.20	15.45%	24.90	41.64	16.17%	25.96	43.21	17.12%	23.00	38.34	14.90%
	DGCRN	23.74	37.36	16.64%	28.62	42.85	21.60%	38.68	57.36	31.15%	24.59	38.90	17.68%
	STG-NCDE	24.27	38.50	16.65%	24.81	39.26	17.32%	26.24	41.25	19.18%	23.56	37.23	16.45%
	MegaCRN	28.50	42.86	22.74%	29.64	44.52	24.31%	31.03	46.46	25.71%	27.17	41.32	21.17%
	D^{2}STGNN	22.24	35.78	15.35%	22.74	36.68	15.49%	27.81	43.05	19.70%	21.49	34.84	14.81%
	STIDGCN	**21.02**	**35.72**	**14.71%**	**21.51**	**35.18**	**14.92%**	**22.30**	**36.12**	**15.80%**	**20.45**	**33.35**	**14.45%**
PEMS08	DCRNN	20.48	31.73	13.73%	21.38	33.03	14.56%	23.12	35.22	16.35%	20.02	30.93	13.58%
	STGCN	27.25	41.28	17.02%	27.63	41.97	17.27%	28.80	43.55	18.24%	26.83	40.86	16.77%
	GWNet	21.27	35.06	13.75%	21.54	35.76	13.96%	22.01	36.38	14.45%	21.00	34.63	13.63%
	MTGNN	19.57	31.55	13.14%	20.45	32.74	14.19%	22.10	34.57	15.95%	18.86	30.36	12.52%
	AGCRN	20.06	31.92	14.39%	20.49	32.71	14.56%	21.58	34.04	14.80%	19.10	30.56	12.99%
	GMAN	17.69	30.69	13.75%	18.13	31.20	14.06%	18.95	32.73	14.52%	17.48	30.14	13.56%
	ASTGNN	21.23	33.38	14.82%	22.22	35.09	15.87%	23.55	36.56	17.06%	20.58	32.29	14.54%
	DGCRN	18.84	30.99	12.65%	23.68	36.47	16.83%	33.45	49.61	26.47%	19.67	32.35	13.62%
	STG-NCDE	22.20	34.28	15.99%	22.73	35.32	15.93%	23.57	36.52	16.67%	21.00	32.58	14.99%
	MegaCRN	25.93	38.97	18.01%	27.13	40.32	19.07%	28.53	41.62	20.42%	23.95	36.36	16.49%
	D^{2}STGNN	17.75	30.21	12.41%	18.55	31.16	13.23%	20.34	32.93	14.89%	17.25	29.16	12.08%
	STIDGCN	**16.84**	**29.44**	**11.57%**	**17.35**	**30.36**	**11.98%**	**18.01**	**31.15**	**12.75%**	**16.24**	**28.11**	**11.19%**

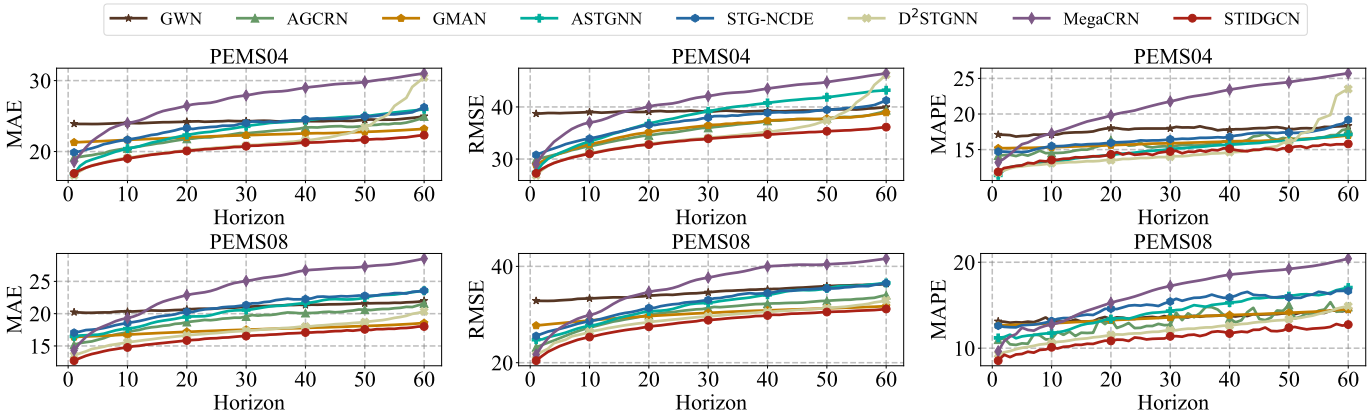


Fig. 4. Long-range forecasting performance comparison at each horizon.

perform well without needing pre-defined graph structures as inputs.

2) Long-range Traffic Flow Forecasting Comparison: To evaluate the model's ability to predict long-range periods, we repartitioned the PEMS04 and PEMS08 datasets using long-range time windows. Since traffic data is a distinct type of time-series data involving both temporal and spatial dimensions, partitioning datasets and deploying models requires careful consideration of significant memory costs. To ensure the successful deployment of the most representative comparison baseline, we repartitioned the PEMS04 and PEMS08 datasets using 60 time horizons.

The experimental results are presented in Table VI, while Fig. 4 visually represent the real-time prediction MAE for each forecasting time horizon in long-range forecasting tasks. With the increasing data length, long-range forecasting becomes more challenging. This requires models that can capture long-range spatial-temporal dependencies. Notably, some well-performing models, such as GWN and MegaCRN, demonstrate relatively mediocre performance in long-range forecasting. GMAN employs an attention mechanism to calculate the correlation between each time step, enabling it to capture long-range temporal dependencies more effectively and thus demonstrate superior performance. On the other hand, D^{2}STGNN, with its unique decoupled dynamic structure, holds an advantage in dealing with long-range forecasting, showcasing remarkable performance.

Compared to the baseline models, our proposed STIDGCN exhibits the fewest errors in long-range forecasting. STIDGCN achieves this by employing continuous data splitting through

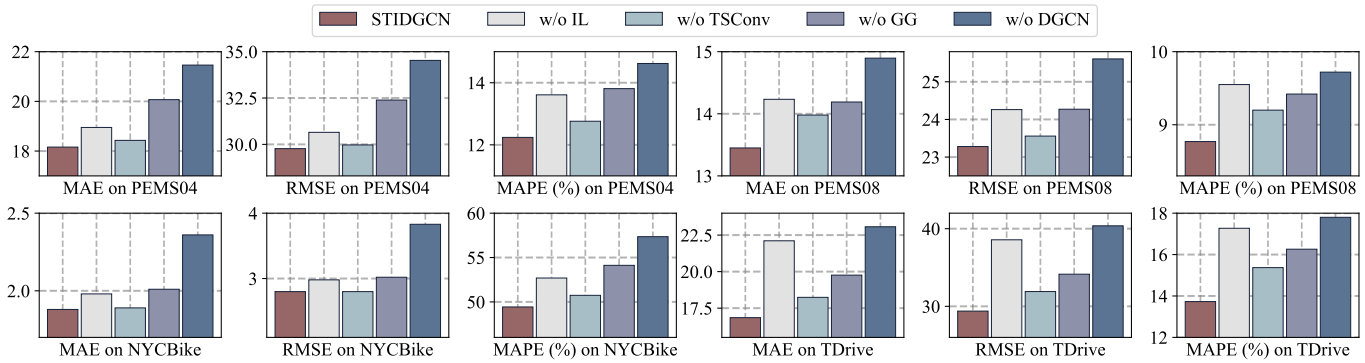


Fig. 5. Ablation study on four traffic datasets.

spatial-temporal interactive learning and conducting feature interactive exploration. The change in perceptual granularity from global to local contributes to the model's exploration of finer-grained spatial-temporal representations. Furthermore, the spatial-temporal representations are employed to reconstruct the dynamical graph's structure at various temporal granularities throughout the sampling process from the global to the local level. This optimization enhances the capture of spatial correlations across different time periods.

D. Ablation Study (RQ2)

To further verify the effectiveness of each component in STIDGCN, we conduct an ablation study on the PEMS04, PEMS08, NYCBike, and TDrvie datasets. We design four variants of STIDGCN as follows:

- ***w/o* IL**: STIDGCN replaces the spatial-temporal interactive learning strategy with a sequential strategy. It initially utilizes the TSConv to capture temporal correlations and subsequently employs DGCN to capture spatial correlations;
- ***w/o* TSConv**: STIDGCN removes the TSConv module from the STI module;
- ***w/o* GG**: STIDGCN removes the graph generator from the DGCN;
- ***w/o* DGCN**: STIDGCN replaces the DGCN with a normal GCN, and the adjacency matrix input to the GCN is the pre-defined adjacency matrix.

The results of the ablation experiments are presented in Fig. 5. Only NYCBike's drop-off and TDrive's inflow results are shown in Fig. 5, as pick-up and outflow behave similarly to the former. The effectiveness of different components in STIDGCN have essentially similar distributions on the experimental datasets. The component with the most significant impact on model performance is the DGCN. DGCN utilizes its internally generated dynamic fusion graph for spatial-temporal representation mining, endowing STIDGCN with the ability to capture dynamic spatial correlations. The result of ***w/o* DGCN** indicates that the performance of STIDGCN significantly deteriorates when DGCN is omitted, as it loses its spatial modeling capability. Similarly, the result of ***w/o GG*** suggests that, although STIDGCN possesses spatial modeling capabilities, the absence of a dynamically fused graph structure generated using historical input information and the failure to take into account the distinct traffic patterns at each node, limit its ability to capture dynamic spatial correlations. Moreover, the result of ***w/o* TSConv** demonstrates that if the model cannot capture temporal correlations and only engages in the interactive capture of dynamic spatial correlations, the model's performance is compromised. This is because spatial-temporal interactive learning results in positive feedback between captured spatial and temporal correlations. The model naturally exhibits poorer performance once this positive feedback in both spatial and temporal dimensions is lost. Furthermore, the result of ***w/o* IL** indicates that the interactive learning between spatial and temporal correlations is more effective than the paradigm of sequentially capturing spatial and temporal correlations. This is because interactive learning enables mutual reinforcement and positive feedback between spatial and temporal correlations.

E. Parameter Sensitivity Analysis (RQ3)

To further investigate parameter sensitivity, we conduct hyperparameter studies on the PEMS04, PEMS08, NYCBike, and TDrive datasets. As mentioned earlier, we only present drop-off demand results for NYCBike and inflow results for TDrive. We select three hyperparameters to investigate their impact on the model's performance: the number of feature channels in the encoder, the kernel sizes of the 2D convolution layers in the TSConv module, and the number of diffusion steps in the DGCN. Among these, kernel size "[5, 3]" indicates that the kernel sizes for the two 2D convolution layers in the TSConv module are (1, 5) and (1, 3), respectively.

The results of the experiments are shown in Fig. 6. First, as the number of feature channels increases, the performance of STIDGCN gradually improves, as having more feature channels allows STIDGCN to learn higher-dimensional spatial-temporal features. However, when the number of feature channels reaches a certain threshold, the performance of STIDGCN stabilizes or even declines. Secondly, different kernel size settings in the convolution layers enable STIDGCN to perceive temporal correlations at different scales, and the optimal kernel size varies depending on the dataset. Lastly, increasing the diffusion step size does not necessarily improve the performance of STIDGCN. Based on prior experience, increasing the

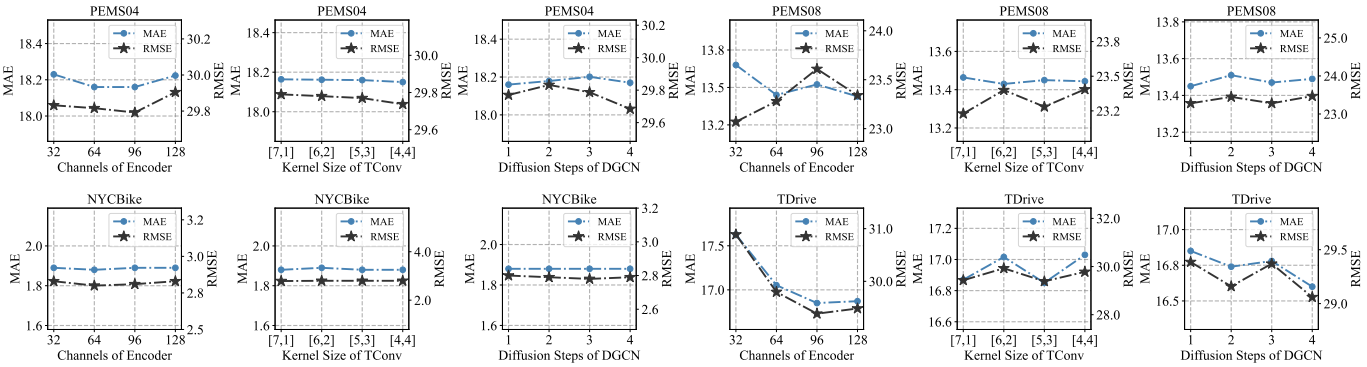


Fig. 6. Hyperparameter study on four traffic datasets.

TABLE VII
HYPERPARAMETER SETTINGS.

Dataset	Batch Size	Learning Rate	Weight Decay	Epoch	Channels C	Kernel Size $s1, s2$	Diffusion Step k
PEMS03	64	0.001	0.0001	200	64	[5, 3]	1
PEMS04	64	0.001	0.0001	300	64	[4, 4]	1
PEMS07	64	0.001	0.0001	300	96	[5, 3]	1
PEMS08	64	0.001	0.0001	300	64	[5, 3]	1
NYCBike	16	0.001	0.0001	300	64	[7, 1]	1
NYCTaxi	16	0.001	0.0001	300	64	[4, 4]	2
TDrive	16	0.001	0.0001	300	96	[4, 4]	2
NYTaxi	16	0.001	0.0001	300	96	[5, 3]	1

diffusion step size in diffusion convolutional networks allows nodes to perceive information from more distant neighbors, thus enhancing the capture of spatial correlations. For all the datasets used in the experiments, we employ three STI modules to process input representations, and each STI undergoes four rounds of interactive dynamic graph convolution operations. Like convolutional neural networks, multiple layers of dynamic graph convolution operations increase the receptive field, enabling nodes to aggregate information from multiple-order neighbors. Increasing the diffusion step size does not significantly improve performance, as multiple DGCN operations already effectively expand the spatial receptive field. Therefore, the hyperparameter settings for STIDGCN vary across different datasets. We outline the detailed optimal hyperparameter settings for STIDGCN on each dataset in Table VII.

F. Robustness Testing (RQ4)

In real-world traffic forecasting scenarios, the data collected may be prone to issues such as missing values and noise. Therefore, when applying models for traffic forecasting in practical applications, it is essential to test the robustness of models. To achieve this, we conduct robustness testing on the PEMS04 and PEMS08 datasets, comparing three suboptimal models, GWN, ASTGNN, and D^{2}STGNN, alongside STIDGCN.

For real-world scenarios involving missing data, such as sensor malfunctions or data upload failures, we simulate data with missing rates of 20%, 40%, and 60% for the training sets of both the PEMS04 and PEMS08 datasets while ensuring the integrity of the test and validation sets. As shown in Fig. 7, as the proportion of missing data increases, the performance of all three models declines. However, STIDGCN consistently exhibits the best performance. Notably, on the PEMS08 dataset, when the missing data rate reaches 60%, STIDGCN's performance even surpasses that of the GWN, ASTGNN, and D^{2}STGN trained on complete data.

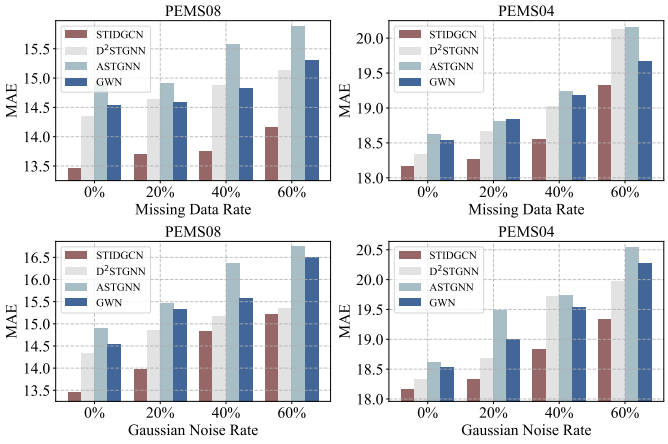


Fig. 7. Robustness testing on PEMS04 and PEMS08 datasets.

For real-world scenarios involving noisy data, such as sensor anomalies and data transmission errors, we introduce Gaussian noise with means of 10 and standard deviations of 500 to the training data of PEMS04 and PEMS08 at proportions of 20%, 40%, and 60%, respectively. The data in the test and validation sets remain free of Gaussian noise. As shown in Fig. 7, across datasets with varying data quality, STIDGCN exhibits greater robustness compared to GWN, ASTGNN, and D^{2}STGN.

G. Case Study (RQ5)

In this section, we conduct visual experiments to demonstrate STIDGCN's performance in real-world application scenarios. To illustrate how STIDGCN learns dynamic graph structure using spatial-temporal information, we visualize the heatmap of the three dynamic adjacency matrices ($\mathbf{A}_p$, $\mathbf{A}_h$, and $\mathbf{A}_f$) during peak and off-peak traffic hours on the PEMS08 dataset. As shown in Fig. 8, We compare them with a predefined and adaptive adjacency matrix. The adaptive adjacency

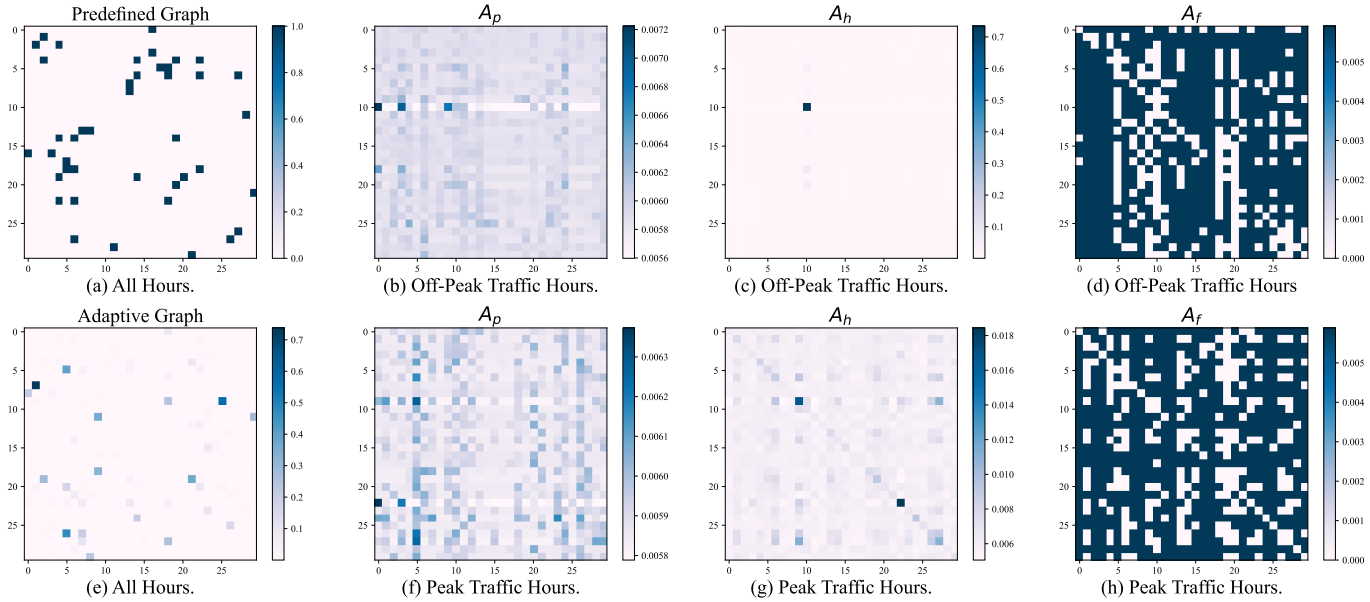


Fig. 8. The heatmap of pre-defined graph, adaptive graph and dynamic graph for the first 30 nodes on PEMS08 dataset.

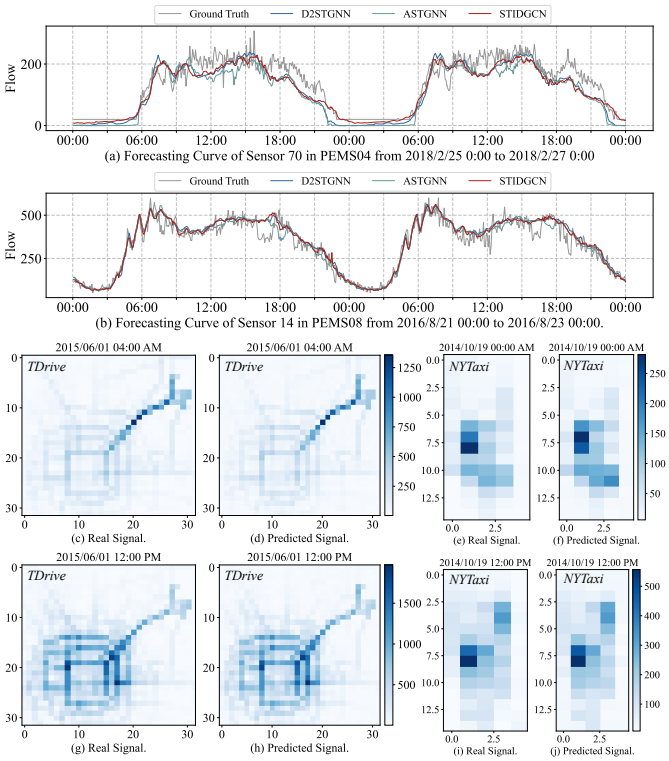


Fig. 9. Visualization of real traffic forecasting results.

matrix is learned by training the STIDGCN without the graph generator. For clarity, we select the first 30 nodes. $\mathbf{A}_p$ is generated by querying the traffic pattern bank using spatial-temporal representations to form latent spatial associations between nodes. This method considers the fact that each node presents different traffic patterns due to spatial heterogeneity, leading to low relevance values and dense node relationships. $\mathbf{A}_h$ is generated by calculating the spatial similarity of each node using spatial-temporal representations, reflecting the degree of relevance of each node to others in the current period. Consequently, it exhibits higher correlation values and sparse node relationships. $\mathbf{A}_f$ is the final dynamic fusion adjacency matrix, which exhibits distinct traffic patterns during different periods, unlike the pre-defined and adaptive adjacency matrices.

We also visually compare the traffic forecasting curves between STIDGCN, ASTGNN, and D^{2}STGNN, as well as the ground truth curves, within specific time periods. Specifically, we select node 70 from PEMS04 and node 14 from PEMS08 for this visual comparison. These nodes exhibit high connectivity by calculating based on the initial adjacencies, indicating their significance within the traffic network's graph structure and representing crucial positions in the transportation network. We select periods spanning weekdays (Friday) and weekends (Saturday) to compare the PEMS04 and PEMS08 datasets comprehensively. As shown in Fig. 9, our proposed STIDGCN model performs better than ASTGNN and D^{2}STGNN, particularly during high traffic flow and pronounced traffic fluctuations. In addition, we perform predictive visualization for peak and off-peak periods using grid-based datasets (TDrive and NYTaxi). As shown in Fig. 9, our proposed model accurately forecasts spatial-temporal changes in traffic flow, providing valuable and constructive suggestions for decision-makers in real-world traffic forecasting tasks.

H. Efficiency Study (RQ6)

In this section, we compare STIDGCN with other methods on PEMS04 and PEMS08 datasets regarding computational costs. As shown in Fig. 10, we compare training time, MAE, and memory occupancy. All models run in the same experimental environment and the batch size is uniformly set to 64. Due to parallel data processing and a lightweight model, GWN has lower computational costs. While methods like GMAN

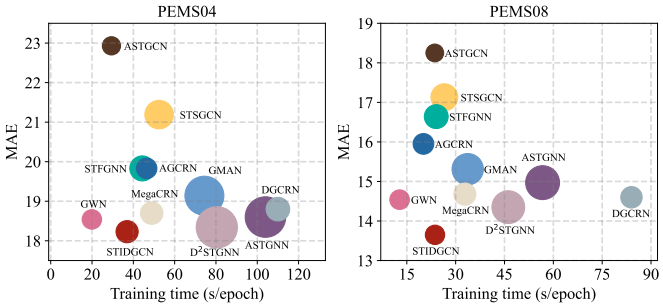


Fig. 10. Computational costs. The area of the scatter represents the memory occupancy.

and ASTGNN, based on attention mechanisms, exhibit excellent performance, their frequent attention calculations lead to significant memory usage. Some autoregressive models, such as D2STGNN and DGCRN, have shortcomings in terms of computational efficiency. Despite D2STGNN's outstanding performance, it comes at the cost of increased computational overhead. Although STIDGCN is not the lowest-cost model, it does not excessively sacrifice computational efficiency in pursuit of performance improvements. Instead, it offers excellent performance without incurring significantly higher computational costs compared to these state-of-the-art baselines.

VI. Conclusion

In this paper, we propose a Spatial-Temporal Interactive Dynamic Graph Convolutional Network (STIDGCN) for traffic forecasting. We propose a spatial-temporal interactive learning strategy to capture relationships by incorporating feature interactions between temporal and spatial information. Additionally, we introduce an effective DGCN module based on the dynamic graph construction method for modeling spatial correlations. This module fully utilizes the spatial-temporal information acquired through interactive learning and incorporates the distinct traffic patterns at each node. The effective capture of these spatial correlations is then propagated through a spatial-temporal interactive learning strategy to reveal deeper spatial and temporal associations. This process establishes a spatial-temporal feedback mechanism that optimizes prediction results. Extensive experiments on eight real-world datasets demonstrate that our model outperforms the comparison baselines while achieving a balance in computational costs. Nevertheless, STIDGCN remains a non-lightweight model. It may require substantial computational resources when deployed on large-scale traffic networks with tens of thousands of nodes. Therefore, investigating methods to reduce the computational resources required for spatial-temporal interactive learning is a promising avenue for further research.

References

[1] S. Rahmani, A. Baghbani, N. Bouguila, and Z. Patterson, "Graph neural networks for intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8846–8885, 2023.

[2] Z. Xiao, X. Fu, L. Zhang, and R. S. M. Goh, "Traffic pattern mining and forecasting technologies in maritime traffic service networks: A comprehensive survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1796–1825, 2019.

[3] 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.

[4] G. Jin, Y. Liang, Y. Fang, Z. Shao, J. Huang, J. Zhang, and Y. Zheng, "Spatio-temporal graph neural networks for predictive learning in urban computing: A survey," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–20, 2023.

[5] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, "Urban traffic prediction from spatio-temporal data using deep meta learning," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1720–1730.

[6] 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.

[7] R. Jiang, Z. Wang, J. Yong, P. Jeph, Q. Chen, Y. Kobayashi, X. Song, S. Fukushima, and T. Suzumura, "Spatio-temporal meta-graph learning for traffic forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 7, 2023, pp. 8078–8086.

[8] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-gcn: A temporal graph convolutional network for traffic prediction," *IEEE transactions on intelligent transportation systems*, vol. 21, no. 9, pp. 3848–3858, 2019.

[9] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling." in *IJCAI*, 2019.

[10] 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.

[11] 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.

[12] S. Guo, Y. Lin, H. Wan, X. Li, and G. Cong, "Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[13] W. Zheng, H. F. Yang, J. Cai, P. Wang, X. Jiang, S. S. Du, Y. Wang, and Z. Wang, "Integrating the traffic science with representation learning for city-wide network congestion prediction," *Information Fusion*, vol. 99, p. 101837, 2023.

[14] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *IJCAI*, 2018.

[15] Q. Zhang, J. Chang, G. Meng, S. Xiang, and C. Pan, "Spatio-temporal graph structure learning for traffic forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 1177–1185.

[16] C. Liu, Z. Xiao, D. Wang, L. Wang, H. Jiang, H. Chen, and J. Yu, "Exploiting spatiotemporal correlations of arrive-stay-leave behaviors for private car flow prediction," *IEEE Transactions on Network Science and Engineering*, 2021.

[17] B. Pu, J. Liu, Y. Kang, J. Chen, and S. Y. Philip, "Mvstt: a multiview spatial-temporal transformer network for traffic-flow forecasting," *IEEE transactions on cybernetics*, 2022.

[18] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Stacked bidirectional and unidirectional lstm recurrent neural network for forecasting network-wide traffic state with missing values," *Transportation Research Part C: Emerging Technologies*, vol. 118, p. 102674, 2020.

[19] J. Kong, X. Fan, M. Zuo, M. Deveci, X. Jin, and K. Zhong, "Adct-net: Adaptive traffic forecasting neural network via dual-graphic cross-fused transformer," *Information Fusion*, vol. 103, p. 102122, 2024.

[20] W. Zhang, Z. Wu, X. Zhang, G. Song, Y. Wang, and J. Chen, "Robust and hierarchical spatial relation analysis for traffic forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 1, pp. 201–217, 2023.

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

[22] F. Li, J. Feng, H. Yan, G. Jin, F. Yang, F. Sun, D. Jin, and Y. Li, "Dynamic graph convolutional recurrent network for traffic prediction: Benchmark and solution," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2021.

[23] 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.
[24] L. Han, B. Du, L. Sun, Y. Fu, Y. Lv, and H. Xiong, "Dynamic and multi-faceted spatio-temporal deep learning for traffic speed forecasting," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 547–555.
[25] S. Lan, Y. Ma, W. Huang, W. Wang, H. Yang, and P. Li, "Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting," in *International Conference on Machine Learning*. PMLR, 2022, pp. 11 906–11 917.
[26] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *Journal of transportation engineering*, vol. 129, no. 6, pp. 664–672, 2003.
[27] Z. Lu, C. Zhou, J. Wu, H. Jiang, and S. Cui, "Integrating granger causality and vector auto-regression for traffic prediction of large-scale wlans," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 10, no. 1, pp. 136–151, 2016.
[28] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in neural information processing systems*, vol. 9, 1996.
[29] U. Johansson, H. Boström, T. Löfström, and H. Linusson, "Regression conformal prediction with random forests," *Machine learning*, vol. 97, no. 1, pp. 155–176, 2014.
[30] J. Van Lint and C. Van Hinsbergen, "Short-term traffic and travel time prediction models," *Artificial Intelligence Applications to Critical Transportation Issues*, vol. 22, no. 1, pp. 22–41, 2012.
[31] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.
[32] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio." *SSW*, vol. 125, p. 2, 2016.
[33] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.
[34] M. Liu, A. Zeng, Z. Xu, Q. Lai, and Q. Xu, "Time series is a special sequence: Forecasting with sample convolution and interaction," *arXiv preprint arXiv:2106.09305*, 2021.
[35] 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.
[36] 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.
[37] H. Lin, R. Bai, W. Jia, X. Yang, and Y. You, "Preserving dynamic attention for long-term spatial-temporal prediction," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 36–46.
[38] 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.
[39] 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.
[40] Z. Fang, Q. Long, G. Song, and K. Xie, "Spatial-temporal graph ode networks for traffic flow forecasting," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 364–373.
[41] J. Choi, H. Choi, J. Hwang, and N. Park, "Graph neural controlled differential equations for traffic forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 6, 2022, pp. 6367–6374.
[42] C. Zheng, X. Fan, S. Pan, H. Jin, Z. Peng, Z. Wu, C. Wang, and S. Y. Philip, "Spatio-temporal joint graph convolutional networks for traffic forecasting," *IEEE Transactions on Knowledge and Data Engineering*, 2023.
[43] Z. Shao, Z. Zhang, W. Wei, F. Wang, Y. Xu, X. Cao, and C. S. Jensen, "Decoupled dynamic spatial-temporal graph neural network for traffic forecasting," *Proc. VLDB Endow.*, vol. 15, no. 11, pp. 2733–2746, 2022.
[44] C. Guo, C.-H. Chen, F.-J. Hwang, C.-C. Chang, and C.-C. Chang, "Fast spatiotemporal learning framework for traffic flow forecasting," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 8, pp. 8606–8616, 2023.
[45] Q. Li, X. Yang, Y. Wang, Y. Wu, and D. He, "Spatial–temporal traffic modeling with a fusion graph reconstructed by tensor decomposition," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12, 2023.
[46] 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.
[47] 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.
[48] J. Wang, Z. Wang, J. Li, and J. Wu, "Multilevel wavelet decomposition network for interpretable time series analysis," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2437–2446.
[49] W. Hua, Z. Dai, H. Liu, and Q. Le, "Transformer quality in linear time," in *International Conference on Machine Learning*. PMLR, 2022, pp. 9099–9117.
[50] J. Ye, L. Sun, B. Du, Y. Fu, and H. Xiong, "Coupled layer-wise graph convolution for transportation demand prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 5, 2021, pp. 4617–4625.
[51] L. Liu, J. Zhen, G. Li, G. Zhan, Z. He, B. Du, and L. Lin, "Dynamic spatial-temporal representation learning for traffic flow prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 7169–7183, 2021.
[52] C. Chen, K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia, "Freeway performance measurement system: mining loop detector data," *Transportation Research Record*, vol. 1748, no. 1, pp. 96–102, 2001.
[53] 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.
[54] L. Wright and N. Demeure, "Ranger21: a synergistic deep learning optimizer," *arXiv preprint arXiv:2106.13731*, 2021.
[55] E. Zivot and J. Wang, "Vector autoregressive models for multivariate time series," *Modeling financial time series with S-PLUS®*, pp. 385–429, 2006.
[56] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in neural information processing systems*, vol. 9, 1996.

Aoyu Liu received the B.S. degree in computer science from Anhui Agricultural University in Hefei, Anhui, China, in 2021. He is currently working toward a Ph.D. degree in the School of Electronic and Information Engineering at Tongji University. His current research interests include spatial-temporal data mining and representation learning.

Yaying Zhang received the B.S. degree in computer science and the M.S. degree in electrical engineering from Shandong University of Science and Technology, Shandong, China, in 1996 and 1999, respectively, and the Ph.D. degree in computer science from Shanghai Jiaotong University, Shanghai, China, in 2004. She is a professor at the Key Laboratory of Embedded System and Service Computing, Tongji University, Shanghai China. Her research interests include spatial-temporal data analysis, data mining and intelligent transportation systems.