Spatial dynamic graph diffusion convolution network for traffic flow forecasting

Huaying Li[1], Siyao Zheng[1], Siwei Long[2], Yue Shen[1], Haisong Feng[1]

[1]Informatics class, School of Informatics, Xiamen University
[2]AI class, Artificial Intelligent Research Institute, Xiamen University
{23020231154146,30920231154372,36920231153221,30920231154335,30920231154330}@stu.xmu.edu.cn

Abstract

Traffic flow forecasting is a challenging task due to the characteristics of high nonlinear and dynamic traffic conditions and complex traffic spatial correlation. Recent methods construct the static spatial graph to model the complex spatial relationship among traffic data and employ RNN-based models to capture temporal dependency. However, the static graph fails to reflect the dynamic changes in the relationship of traffic networks. Meanwhile, The dynamic relationship of traffic nodes has the characteristics of time delay that most methods ignore. To improve the modeling performance, we propose a spatial dynamic graph diffusion convolution network (SDGFN) for traffic flow forecasting. With the support of the time delay feature transformation network, the dynamic graph learning module integrates the passing traffic pattern and the current traffic pattern to model the dynamic characteristics of the traffic network at each time. By embedding the dual dynamic graph diffusion convolution into a gated recurrent uni, our model can capture spatiotemporal dependency simultaneously. To enhance the ability to capture long-term temporal correlation, we adopt the transformer in the SDGFN. We conduct our experiments on two public traffic datasets that demonstrate the effectiveness of our model compared with conventional spatial-temporal traffic flow forecasting model.

Introduction

Traffic flow forecasting is a core component of Intelligent Transportation Systems (ITS) and a crucial foundation for traffic information services, traffic control, and guidance (Lu et al. 2021). It involves the utilization of dynamic traffic data collected to predict future traffic flow within the road network. A reliable and accurate traffic flow prediction system is essential to alleviate network pressure, plan vehicle routes, and ensure the efficiency and safety of the transportation system.

Traffic flow forecasting methods are primarily divided into two categories, knowledge-driven and data-driven. Early knowledge-driven methods employed queuing theory and behavioral simulation, while later data-driven methods developed rapidly, incorporating traditional statistical and machine learning techniques, including auto-regressive integrated moving average (ARIMA) model, Kalman filtering, and SVR, etc. These methods utilize historical traffic data to forecast traffic conditions in the next time period but are constrained by the assumption of time series stationary and overlook spatiotemporal correlations, resulting in poor prediction accuracy. To capture the complex temporal correlation of the historical traffic flow of the road networks, existing methods have been proposed to combine the Graph Neural Networks (GNN) with time series model (Chen et al. 2019; Li et al. 2017). However, there are still some challenges in traffic flow forecasting.

First, Most GNN-based approaches (Wu et al. 2019; Kong et al. 2022) utilize static graph structures and predefined adjacency which is constructed based on Euclidean distances between sensors at road nodes to model the correlation between each traffic node. However, traffic flow data demonstrates high non-linearity and is dynamically influenced by multiple factors, causing the correlations between road networks to vary dynamically (Zheng et al. 2020a). GNN-based on static graphs are unable to dynamically capture the correlation between each node. Spatial-Temporal Fusion Graph Neural Networks (STFGNN) (Li and Zhu 2021) constructed a Spatial-Temporal Fusion Graoh, which consists of the temporal graph, temporal connectivity graph, and spatial graph, but it still ignores modeling dynamic graphs. Although GraphWaveNet (Wu et al. 2019), ASTTN (Feng and Tassiulas 2022), etc propose adaptive adjacency matrix to reflect global spatial correlation. They all fail to achieve the effect of modeling the real spatial relationship under each time node at each time step. Meanwhile, if a sudden emergency happens at a traffic node, it will influence its surrounding nodes after a few time steps. This time-delay influence is seldom considered in static graphs or adaptive graphs (Jiang et al. 2023).

Second, most time series methods have insufficient capture of long-time series influence. because of the gradient vanishing and explosion. RNN-based methods (Li et al. 2021) are representative of timing series models, capturing long-term dependencies poorly because of gradient explosion and gradient vanishing. Temporal Convolution Network (TCN), a special one-dimensional CNN, is utilized to capture features along time dimensions (Wu et al. 2019; Li and Zhu 2021; Zhao et al. 2019). By stacking several TCN layers, the models can capture long-term time influence, but each temporal convolution layer is still limited by the receptive field and cannot capture very long-term dependencies (Guo et al. 2021).

To address the above problems, we propose a Spatial Dynamic Graph Diffusion Convolution Network (SDGFN) for traffic flow forecasting. First, a dynamic graph is generated at each time step with the effect of the time delay feature transformation network (Jiang et al. 2023) and hypernetwork. The dynamic graph at each time not only reflects the current traffic node relationship but also combines the influence of the traffic node of the past time on the current time node, considering the time delay on node characteristics. Second, we utilize the gated recurrent unit with transformer to capture short-term and long-term time series. The diffusion graph convolution is embedded into the grated recurrent units to learn global spatial correlation and temporal correlation. In summary, the main contributions are the following:

- We propose a graph learning network with a time delay feature transformation network to generate a dynamic graph at each time node without any prior knowledge, considering the current traffic node correlation and the passing traffic node effect.
- We propose that the diffusion graph convolution into the gated recurrent unit to capture spatio-temporal dependency simultaneously. The diffusion graph convolution network captures the dependencies of surrounding nodes by considering the traffic data as a diffusion signal. To handle long-term dependency capturing, we adopt the transformer to discover the global temporal dependency.
- We conduct experiments on two real-world traffic datasets, the experimental results demonstrate the effectiveness of SDGFN on traffic flow forecasting.

Related work

Traffic flow forecasting

The task of traffic flow forecasting is to learn a function that maps the historical traffic flow into the future traffic flow. A traffic network can be denoted as $G = (V, E, A)$, where V represents the set of nodes $|V| = N$ and E is the set of edges. The spatial adjacency matrix is represented as $A \in \mathbb{R}^{N\times N}$. $A_{ij} = 1$ means there is an edge between node i and node j, otherwise 0. $X_t \in \mathbb{R}^{N\times D}$ represented the traffic signal, where D is the number of traffic features.(e.g., road network occupancy, traffic speed, capacity, date) in time step t, Given the historical traffic flow $X_{t-P+1:t}$ and the future $X_{t+1:t+Q}$, the function is formulated as:

$$F(X_{(t-P+1):t}, G) \rightarrow \hat{X}_{(t+1):(t+Q)} \tag{1}$$

where $X_{(t-P+1):t} = (X_{t-P+1}, X_{t-P+2}, ..., X_t) \in \mathbb{R}^{P\times N\times D}$ and $\hat{X}_{(t+1):(t+Q)} = (\hat{X}_{t+1}, \hat{X}_{t+2}, ..., \hat{X}_{t+Q})$.

Spatial-temporal Graph Neural Networks

Spatial-temporal Graph Neural Networks (STGNN) (Yu, Yin, and Zhu 2017) is a common idea to model the transportation system. The key idea for STGNN is to jointly model spatial dependency and temporal dependency at the same time, combining various techniques, including graph neural network and time sequence models. SDGCN (Li et al. 2023) combines the GRU and the variant of GCN to capture spatial and temporal dependency simultaneously. Diffusion Convolutional Recurrent Neural Network(DCRNN) (Li et al. 2017) incorporates a proposed diffusion graph convolutional layer into a GRU layer. Diffusion convolution is a variant of graph convolution networks (GCN) and is particularly suitable for handling non-Euclidean relationships between multiple time series in traffic data. Previous methods utilize a pre-defined graph to reflect the correlation between each traffic nodes, it is limited to learning effective correlations of network with missing genuine relations, some methods proposed adaptive GCN which can capture the hidden dependency to complete incomplete information (Wu et al. 2019; Kong et al. 2022).

Self-Attention mechanism

Attention is a fundamental operation to model the dependency between a collection of values and the target under a query. Attention mechanism has been widely applied in natural language processing (NLP) (Young et al. 2018; Liu and Guo 2019), image recognition (Hossain et al. 2019; Huang et al. 2022), protein identification (Zhou et al. 2022; Dou et al. 2022), recommended system (Zheng et al. 2020b) and etc.

Self-attention which emphasizes many-to-many is a variant of the Attention mechanism. In recent years, multi-head self-attention has been applied to traffic flow prediction (Wang et al. 2020; Guo et al. 2021; Kong et al. 2022). First, many attention models are currently attached to the Encoder-Decoder framework. The framework stacks multiple layers to capture the dynamics of traffic data more effectively with multi- head attention (Feng and Tassiulas 2022; Zheng et al. 2020a; Guo et al. 2021). Second, multi-head attention is encapsulated in transformer layers which follow recurrent network structure. The processed sequence information via a recurrent network is passed into the attention layer to aggregate information (Wang et al. 2020; Yan, Ma, and Pu 2021). Lastly, Multi-head attention is also embedded in other types of network structures to predict traffic flow (Kong et al. 2022; Fang et al. 2022; Huang et al. 2020).

Proposed Solution

The framework of the Spatial Dynamic Graph Diffusion Convolution Network is presented in Figure 1. It consists of three main components, graph learning, dynamic diffusion convolutional recurrent module, and transformer. We describe them in more detail in the following subsection.

Traffic node embedding

Traffic flow is influenced by many external effects, including people's travel patterns and lifestyle, such as Urban areas can be particularly congested during the morning rush hour and evening rush hour. To model the comprehensive traffic feature, we consider two additional embeddings to cover weekly and daily periodicity. The traffic node embedding

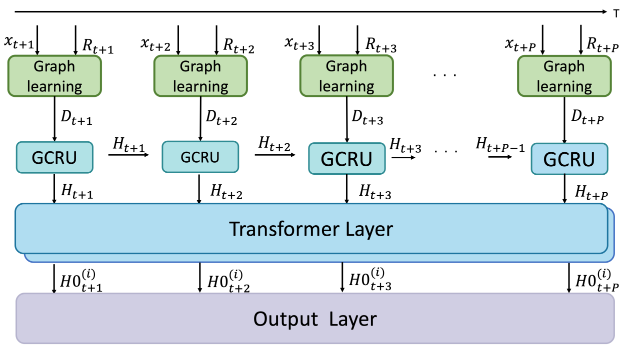

Figure 1: The framework of SDGFN.

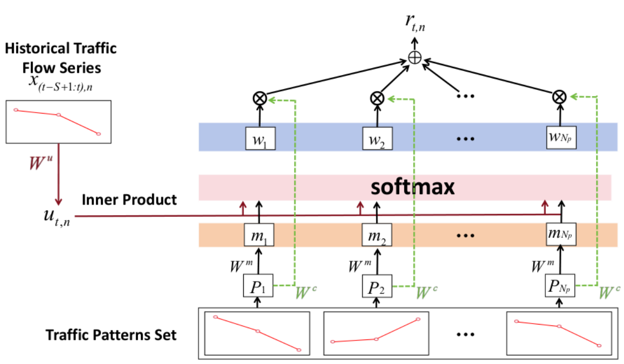

Figure 2: Time Delay Feature Transformation Module

$\mathbf{x} \in \mathbb{R}^{N\times T\times d}$ can be formulated as follows.

$$\begin{aligned} \mathbf{x} &= X_{data} + X_w + X_d \\ X_w &= Embedding(X[:,:,1]) \\ X_d &= Embedding(X[:,:,2]) \end{aligned} \tag{2}$$

Where $X_{data} \in \mathbb{R}^{N\times T\times d}$ is the traffic data $X \in \mathbb{R}^{N\times T\times D}$ being mapped into the high dimensions. $X_w \in \mathbb{R}^{N\times T\times d}$ and $X_d \in \mathbb{R}^{N\times T\times d}$ are stands for the week embedding and daily embedding respectively.

Time delay feature transformation module

In real-world traffic conditions, there exists a time delay in the impact between traffic nodes. For example, when a traffic accident occurs in one region, it may take several minutes to affect traffic conditions in neighboring regions. Therefore, we employ the time delay feature transformation module (Jiang et al. 2023) to capture the propagation delay from the short-term historical traffic flow of each node. This module incorporates delay information to graph learning to model current traffic correlation at each time step.

First, we slice the historical traffic data with a sliding window of size S to obtain a set of traffic flow series. Then, we perform k-shape clustering algorithm (Paparrizos and Gravano 2015) to cluster the similar pattern traffic flow series. The centroid $\mathbf{p_i}$ of each cluster is to represent the cluster, the set $\mathcal{P} = \{\mathbf{p}_i \mid i \in [1, \cdots, N_p]\}$ to represent the clustering results, where N_p is the total number of clusters. The $\mathcal{P}$ can be regarded as a set of short-term traffic patterns.

Similar traffic patterns share similar effects on neighborhood traffic conditions, especially abnormal traffic patterns, congestion. We fuse the information of similar patterns into the historical flow series representation of each node by comparing the historical traffic flow series for each node with the extracted traffic pattern set $\mathcal{P}$, shown in Figure 2. Given the S-step historical traffic flow series of node n from time slice $(t-S+1)$ to t, denoted as $\mathbf{x}_{(t-S+1:t),n}$, we first use the transformation matrix $\mathbf{W}^u$ and $\mathbf{W}^m$ to obtain high-dimensional representations $\mathbf{u}_{t,n}$, $\mathbf{m}_i$ respectively.

$$\begin{aligned} \mathbf{u}_{t,n} &= \mathbf{x}_{(t-S+1:t),n}\mathbf{W}^u \\ \mathbf{m}_i &= p_i\mathbf{W}^m \end{aligned} \tag{3}$$

We compare the historical traffic flow representation $\mathbf{u}_{t,n}$ of node n with the traffic pattern memory vector m_i and obtain the similarity vector $\mathbf{w}_i$, which is used to obtain the integrated historical series representation $\mathbf{r}_{t,n}$ by a weighted sum of the traffic pattern set $\mathcal{P}$.

$$\begin{aligned} \mathbf{w}_i &= softmax(\mathbf{u}_{t,n}^T\mathbf{m}_i) \\ \mathbf{r}_{t,n} &= \sum_{i=1}^{N_p}\mathbf{w}_i(\mathbf{p}_i\mathbf{W}^c) \end{aligned} \tag{4}$$

where $\mathbf{W}^c$ is a learnable matrix. The feature of propagation delay from the short-term historical traffic flow of each time steps R_t can be obatined by concating the $\mathbf{r}_{t,n}$ along the node dimensions. R_t is been sent to graph learning to learn a dynamic graph during end to end learning.

Graph Learning

Traffic condition is highly dynamic and the correlations between each node are changed with time as well. It not only is influenced by current traffic flow but also affected by past short time traffic flow. To model the dynamic spatial dependency, Following DGCRN (Li et al. 2021), we employ a hyper-network to learn hidden representations from the traffic data to generate a dynamic graph to preserve hidden spatial dependency more effectively at each time step. At each time step, We combine the current traffic data $\mathbf{x}_t$ with the historical traffic flow representation R_t. This process explicitly models the time delay in spatial information propagation at each time step.

$$H_t = \mathbf{x}_t \parallel R_t \tag{5}$$

where $H_t \in \mathbb{R}^{B\times N\times d}$ is dynamic node feature, d is the feature dimension, B is the batch size, N is the number of nodes, $\parallel$ represents the concatenation operation. The H_t is fed into graph convolution module to update the node representation.

$$DF_t = \Theta_{\star G}(H_t) \tag{6}$$

where $\Theta_{\star G}$ represents the graph convolution, Θ denotes the learnable parameters. We utilize the pre-defined adjacency matrix A to conduct the message-passing process for dynamic node status. The output from graph convolution is called as dynamic filter DF_t. We employ element-wise multiplication between dynamic filter DF_t and the model parameters $\Theta \in \mathbb{R}^{N\times d_s}$ to dynamically adjust the correlation between each node to obtain dynamic node embedding. The formula of generating a graph is described as follows:

$$DE_1^t = tanh(\beta\langle DF_{t,1}, \Theta_1\rangle) \\ DE_2^t = tanh(\beta\langle DF_{t,2}, \Theta_2\rangle) \tag{7}$$

$$DA^t = ReLU(tanh(\beta(DE_1^t {DE_2^t}^T - DE_2^t {DE_1^t}^T))) \tag{8}$$

where Θ_1 and Θ_2 are learned parameters during end to end learning. β is a hyper-parameter for controlling the saturation rate of the activation function. $\langle\ ,\ \rangle$ is Hadamard product. We name $DE_1^t \in \mathbb{R}^{N\times d_s}$ as the source dynamic node embedding and $DE_2^t \in \mathbb{R}^{N\times d_s}$ as the target dynamic node embedding. By calculating the similarity between source dynamic node embedding DE_1^t and target dynamic node embedding DE_2^t, the dynamic adjacency matrix DA^t can be generated.

Dynamic Diffusion Convolutional Module

The Predefined graph reflects the stationary correlation between each node. Instead, a dynamic graph is changed with time to reflect the dynamic spatial correlation among traffic data. We combine the static graph and the dynamic graph in graph representation learning to detect hidden spatial dependency. The bidirectional diffusion convolution (Li et al. 2017) is embedded into graph convolution in graph representation learning. Especially, in the part of dynamic graph diffusion convolution, we consider the two-direction traffic flow influence in a dynamic graph at the same time to capture more influence from both the upstream and the downstream traffic.

$$\begin{aligned} g_\theta \star G(x) &= \sum_{k=0}^{K} \theta_{k,f} P_f^k x + \theta_{k,b} P_b^k x + \\ &\quad \theta_{k,df} \tilde{DA}_f^{t,k} x + \theta_{k,db} \tilde{DA}_b^{t,k} x, \\ \tilde{DA}_f^t &= \tilde{DA}^t / \sum_j (\tilde{DA}_{ij}^t), \\ \tilde{DA}_b^t &= (\tilde{DA}^t)^T / \sum_j ((\tilde{DA}^t)^T{}_{ij}), \\ \tilde{DA}^t &= DA^t + I \end{aligned} \tag{9}$$

where $P_f = D_o^{-1}A$ and $P_b = D_I^{-1}A^T$ represents the transition matrix. $D_o = diag(\mathbf{A1})$ and $D_I = diag(\mathbf{A}^T\mathbf{1})$ are the out-degree diagonal matrix and $\mathbf{1} \in \mathbb{R}^N$. P_f^k and P_b^k represents the power series of the transition matrix P_f, P_b. $\theta_{k,f}$ and $\theta_{k,b}$ are learnable parameters. $\tilde{DA}_f^t$ represents dynamic forward transition matrix. $\tilde{DA}_b^t$ represents dynamic backward transition matrix. $\theta_{k,f}$, $\theta_{k,b}$, $\theta_{k,df}$, $\theta_{k,db}$ are learnable parameters.

Temporal Dependency Module

Gated Recurrent Units (GRU) with fewer parameters can effectively tackle gradient vanishing and capture long time series. We utilize the RNN-based model to capture temporal dependency. To capture spatial and temporal correlation simultaneously, we replace matrix multiplications in GRU with the dynamic diffusion convolution and name it Graph Convolution Recurrent Unit (GCRU).

$$\begin{aligned} u^{(t)} &= \sigma(\Theta_{u\star G}(\mathbf{x}_t || H_{t-1})), \\ r^{(t)} &= \sigma(\Theta_{r\star G}(\mathbf{x}_t || H_{t-1})), \\ C^{(t)} &= tanh(\Theta_{C\star G}(\mathbf{x}_t || (r^{(t)} \odot H_{t-1}))), \\ H_t &= u^{(t)} \odot H_{t-1} + (1 - u^{(t)}) \odot C^{(t)} \end{aligned} \tag{10}$$

Where $\mathbf{x}_t$ denotes the traffic signal at time t, H_{t-1} denotes the output from GCRU at time $t-1$. $r^{(t)}$, $u^{(t)}$ are reset gate and update gate at time t. $\star G$ denotes the dynamic diffusion convolution which is defined in (9). Θ_r, Θ_u, Θ_h are learned parameters for the diffusion convolution layer.

To further explore global temporal dependency more effectively, we employ transformer (Vaswani et al. 2017) to learn the long-term temporal dependency finally.

Experiments

Table 1: The properties about PeMS04 and PeMS08 datasets.

Dataset	Nodes	Edges	Time Steps	Time Windows
PeMS04	307	680	16992	5min
PeMS08	170	548	17856	5min

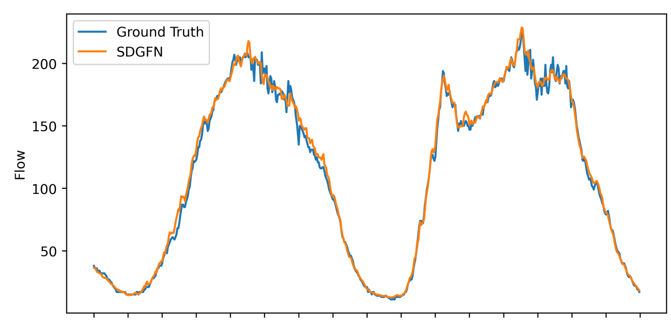

Figure 3: 15 minutes ahead traffic flow forecasting on node 200 of PeMS04

To verify the performance of SDGFN, we experiment on two public traffic datasets, PeMS04 and PeMS08. PeMS04 traffic data was collected from 307 sensors in the District04 from Jan 1st, 2018 to Feb 28th, 2018. PeMS08 records for two months of 170 sensors traffic flow in the District08, ranging from July 1st, 2018 to Aug 31st, 208. More details about PeMS04 and PeMS08 are presented in Table 1. We do the same data preprocessing and construct sensor graph as DCRNN (Li et al. 2017). We divide two datasets respectively, in which 70% of data is used as the training set, 20% of data as the testing set and the remaining data is used for the validation set.

Table 2: Comparison of the performance of SDGFN and baselines on PeMS04 and PeMS08 datasets.

Datasets	Model	HA	SVR	FC-LSTM	DCRNN	STGCN	STFGNN	SDGCN	SDGFN
PeMS04	MAE	36.54	29.34	27.89	22.737	21.758	19.830	20.42	**19.781**
	MAPE	26.48	19.91	21.86	14.751	13.874	**13.021**	14.99	13.672
	RMSE	53.16	44.51	41.07	36.575	34.769	**31.870**	34.20	31.899
PeMS08	MAE	30.15	24.15	23.10	18.185	17.838	16.636	17.01	**16.157**
	MAPE	19.09	15.30	18.56	11.235	11.211	**10.547**	10.94	10.647
	RMSE	44.20	36.42	34.07	28.176	27.122	26.206	26.61	**25.073**

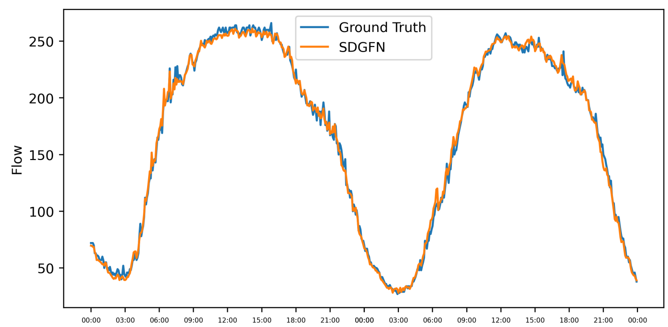

Figure 4: 15 minutes ahead traffic flow forecasting on node 50 of PeMS08

Experimental Settings

We conduct our experiments under the Linux environment with one Intel(R) Xeon(R) Sliver 4210 CPU @ 2.20GHZ and one NVIDIA Geforce RTX3090 GPU cards. In our model, we set the batch size to 64 and the hidden dimension d to 64 for PeMS04 and PeMS08 datasets. In GCRU, the depth of dynamic graph diffusion recurrent convolution layer is set to 1, and the depth of graph diffusion convolution K is 2. We set the number of attention heads in the transformer layer as 2. The Adam optimization with an initial learning rate of 0.001 is utilized to train the model. We utilize the fully connected network in output layer to map the feature to the traffic flow predictions. The maximum epoch is set to 150. To avoid overfitting, we employ early stopping during training.

Experimental Results

Table 2 illustrates the average performance of SDGFN and seven baselines. SDGFN outperforms both the temporal model (HA, SVR, FC-LSTM) and the spatio-temporal model(DCRNN (Li et al. 2017), STGCN (Yu, Yin, and Zhu 2017), STFGNN (Li and Zhu 2021), SDGCN (Li et al. 2023)). We observe the following experimental phenomenons.

- Benefits from the dynamic graph, which can be learned during end-to-end learning, the performance of SDGCN and SDGFN work better than other spatio-temporal models (STGCN, DCRNN). The dynamic graph at each time step reflects the current traffic topology structure more effectively and preserves hidden spatial dependency which the static graph doesn't have which is helpful in traffic flow forecasting. However, SDGCN has a slightly weaker relationship in modeling dynamic spatiotemporal changes compared with SDGFN. Probably the time delay feature transformation module effectively extracts the influence of the past short-term traffic flow on the current traffic flow to generate dynamic graph.
- Compared with STFGNN, the experiment indicates that SDGFN performs similarly to STFGNN. STFGNN and SDGFN all have improved on the generated graph, static graph instead. SDGFN generates the dynamic graph at each time step, combing with the static graph to dual graph diffusion convolution. The dynamic graph provides more real-time node correlation to help the diffusion convolution to capture spatial dependencies more effectively. STFGNN constructs several static graphs to model the traffic node correlation from different aspect, instead, which can also achieve the effect of dynamic graphs to a certain extent.
- Further, we visualize the 15 minutes ahead forecasting result and the real result in Figure 3, 4. SDGFN outputs the smooth prediction in PeMS04 and PeMS08 and accurately predicts the trend of traffic flow.

Conclusion

In this paper, we propose a spatial dynamic graph diffusion convolution network to traffic flow forecasting. We propose a graph learning network which employ an hyper-network with time-delay feature transformation module to generate dynamic graphs without any prior knowledge to preserve hidden spatial dependency. Moreover, we propose making dynamic Graph diffusion convolution embed into the gated recurrent unit to capture spatial-temporal dependency simultaneously. Further, the transformer in SDGFN is to enhance the model capacity to explore long-term temporal dependency. Extensive experiments on two public traffic datasets demonstrate that the superior performance of SDGFN on traffic flow forecasting.

References

Chen, C.; Li, K.; Teo, S. G.; Zou, X.; Wang, K.; Wang, J.; and Zeng, Z. 2019. Gated residual recurrent graph neural networks for traffic prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 485–492.

Dou, H.; Tan, J.; Wei, H.; Wang, F.; Yang, J.; Ma, X.-G.; Wang, J.; and Zhou, T. 2022. Transfer inhibitory potency prediction to binary classification: A model only needs a small training set. *Computer Methods and Programs in Biomedicine*, 215: 106633.

Fang, W.; Zhuo, W.; Yan, J.; Song, Y.; Jiang, D.; and Zhou, T. 2022. Attention meets long short-term memory: A deep learning network for traffic flow forecasting. *Physica A: Statistical Mechanics and its Applications*, 587: 126485.

Feng, A.; and Tassiulas, L. 2022. Adaptive Graph Spatial-Temporal Transformer Network for Traffic Forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 3933–3937.

Guo, S.; Lin, Y.; Wan, H.; Li, X.; and Cong, G. 2021. Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 34(11): 5415–5428.

Hossain, M. Z.; Sohel, F.; Shiratuddin, M. F.; and Laga, H. 2019. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CsUR)*, 51(6): 1–36.

Huang, B.; Tan, G.; Dou, H.; Cui, Z.; Song, Y.; and Zhou, T. 2022. Mutual gain adaptive network for segmenting brain stroke lesions. *Applied Soft Computing*, 129: 109568.

Huang, R.; Huang, C.; Liu, Y.; Dai, G.; and Kong, W. 2020. LSGCN: Long Short-Term Traffic Prediction with Graph Convolutional Networks. In *IJCAI*, volume 7, 2355–2361.

Jiang, J.; Han, C.; Zhao, W. X.; and Wang, J. 2023. PDFormer: Propagation Delay-aware Dynamic Long-range Transformer for Traffic Flow Prediction. In *AAAI*. AAAI Press.

Kong, X.; Zhang, J.; Wei, X.; Xing, W.; and Lu, W. 2022. Adaptive spatial-temporal graph attention networks for traffic flow forecasting. *Applied Intelligence*, 1–17.

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

Li, H.; Yang, S.; Song, Y.; Luo, Y.; Li, J.; and Zhou, T. 2023. Spatial dynamic graph convolutional network for traffic flow forecasting. *Applied Intelligence*, 53(12): 14986–14998.

Li, M.; and Zhu, Z. 2021. Spatial-temporal fusion graph neural networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 4189–4196.

Li, Y.; Yu, R.; Shahabi, C.; and Liu, Y. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

Liu, G.; and Guo, J. 2019. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing*, 337: 325–338.

Lu, H.; Ge, Z.; Song, Y.; Jiang, D.; Zhou, T.; and Qin, J. 2021. A temporal-aware lstm enhanced by loss-switch mechanism for traffic flow forecasting. *Neurocomputing*, 427: 169–178.

Paparrizos, J.; and Gravano, L. 2015. k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD international conference on management of data*, 1855–1870.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, X.; Ma, Y.; Wang, Y.; Jin, W.; Wang, X.; Tang, J.; Jia, C.; and Yu, J. 2020. Traffic flow prediction via spatial temporal graph neural network. In *Proceedings of the web conference 2020*, 1082–1092.

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

Yan, H.; Ma, X.; and Pu, Z. 2021. Learning dynamic and hierarchical traffic spatiotemporal features with transformer. *IEEE Transactions on Intelligent Transportation Systems*, 23(11): 22386–22399.

Young, T.; Hazarika, D.; Poria, S.; and Cambria, E. 2018. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13(3): 55–75.

Yu, B.; Yin, H.; and Zhu, Z. 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*.

Zhao, L.; Song, Y.; Zhang, C.; Liu, Y.; Wang, P.; Lin, T.; Deng, M.; and Li, H. 2019. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 21(9): 3848–3858.

Zheng, C.; Fan, X.; Wang, C.; and Qi, J. 2020a. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 1234–1241.

Zheng, L.; Guo, N.; Chen, W.; Yu, J.; and Jiang, D. 2020b. Sentiment-guided sequential recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 1957–1960.

Zhou, T.; Dou, H.; Tan, J.; Song, Y.; Wang, F.; and Wang, J. 2022. Small dataset solves big problem: An outlier-insensitive binary classifier for inhibitory potency prediction. *Knowledge-Based Systems*, 251: 109242.