APSIPA Transactions on Signal and Information Processing, 2024, 13, e10

Original Paper

POI-based Double-deck Graph Convolution Network for Traffic Forecasting

Xinhao Zhao[1], Zhe Wu[2*], Xinfeng Zhang[1], Zhiyuan Deng[1], Li Su[1,2,3*], Guorong Li[1,3], Yaowei Wang[2] and Qingming Huang[1,3]

[1] *University of Chinese Academy of Sciences, Beijing, China*
[2] *Peng Cheng Laboratory, Shenzhen, China*
[3] *Key Lab. of AI Safety, Chinese Academy of Sciences, Beijing, China*

ABSTRACT

Traffic flow forecasting, a vital task of multivariate time series prediction, has recently expanded to incorporate Points of Interest (POI) as an additional source of data. Rather than merely leveraging historical traffic flows, POIs facilitate the understanding of inherent geographical connections and potential functional interactions between nodes. However, traditional POI-based methods tend to use POI as static feature embeddings to compute functional similarity matrices, failing to consider the dynamic influence of node functionality on traffic patterns. This overlooks the reality that even regions with analogous POIs can exhibit fluctuating traffic flow trends, particularly over extended periods. In this paper, we propose the POI-based Double-deck Graph Convolution Network (PDGCN) for more nuanced traffic forecasting. To identify potential POI-based traffic patterns, we employ the spectral clustering method to group nodes with comparable POI functionalities into regions. We then devise a POI-based dynamic graph module with temporal convolution and attention mechanisms to trace the evolving relationships between traffic nodes. This novel design underpins

*Corresponding authors: Zhe Wu, wuzh02@pcl.ac.cn, and Li Su, suli@ucas.ac.cn.

Received 24 December 2023; revised 07 March 2024; accepted 25 March 2024
ISSN 2048-7703; DOI 10.1561/116.20230081

regional features. Experiments on two real datasets demonstrate that PDGCN effectively detects dynamic functional relationships between nodes and delivers superior prediction accuracy.

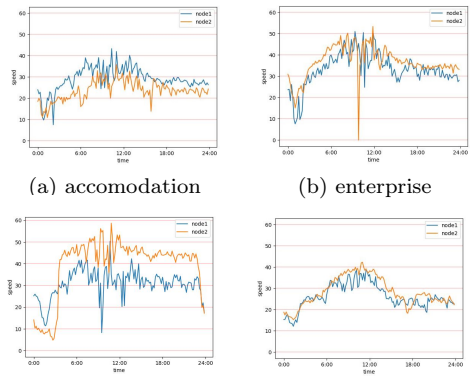
Keywords: traffic forecasting, dynamic graph construction, spatial attention mechanism.

1 Introduction

Traffic forecasting is a typical task of multivariate time series forecasting and has been widely applied in the modern Intelligent Transportation System (ITS). It can improve traffic conditions and develop effective control measures [3]. Traffic forecasting focuses on predicting future traffic features (e.g., flow, speed) on the road network. The early works apply traditional statistical methods [2, 26] to predict traffic flow. In recent decades, researchers have developed many Recurrent Neural Network (RNN) based [1, 18] and Graph Convolution Network (GCN) based methods [8, 28, 7, 9, 27] to model the spatial and temporal information of the traffic network.

Traffic flow originates from people's needs to move between POIs for daily requirements fulfillment. POIs commonly refer to geographically located entities with specific functionalities of interest to people, such as restaurants, shopping malls, residences, and so on. Therefore, POIs reflect the functionality of the traffic nodes and have a close relationship to traffic flow. Figure 1 shows that nodes with similar POI often have similar traffic patterns. Therefore, some methods employ POI to mine the inherent traffic patterns. For example, researchers apply POIs directly as geographical features of the traffic nodes [23, 10]. In addition, some works use POIs to explore the spatial relationship between nodes [19, 29, 31].

However, most current POI-based approaches treat POI as a static spatial feature to calculate the functional similarity between nodes, neglecting the dynamic effects of POI on inherent traffic patterns. For example, enterprise areas and residences may have heavy traffic during weekday rush hours, and shopping areas may have heavy traffic in the evening. The time-varying change can be shown in Figure 2. Moreover, some studies adopt geographic distance as a proxy for inter-nodal relations. They propose regional networks to facilitate traffic flow prediction, overlooking the analogous traffic patterns between nodes emanating from their functional similarity. Cities are naturally divided into regions owning to the functional homogeneity of nodes, reflecting their intrinsic geographical relation. For example, factories gather together to facilitate centralized treatment of pollution. The region division can be seen in Figure 2.



(a) accomodation (b) enterprise

(c) public facility (d) shopping

Figure 1: Different traffic patterns under different POIs of JiNan dataset.

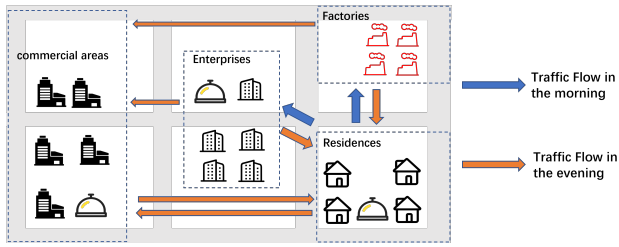


Figure 2: Example of aggregating road nodes with similar POIs into regions and the time-varying changes of traffic flow between regions.

Traffic nodes in the same region have similar traffic patterns. Mining potential traffic patterns in the region can help predict traffic flow at the node level.

This work proposes a POI-based Double-deck Graph Convolution Network (PDGCN) to capture the potential POI-based traffic patterns for traffic forecasting. We model the features of each node according to its POI distribution. Specifically, the density and imbalance of POI distribution will influence the functional similarity of nodes. We use these properties to calculate the functional similarity matrix and obtain the functional region division. Unlike the previous work that mining relationships between nodes, we explore the traffic flow patterns at both node and regional levels, which captures the hierarchical geographical relationship. The regional-level traffic features can assist node-level traffic flow prediction. Moreover, aiming to extract spatial information, we propose a POI-based dynamic graph structure inspired by tensor decomposition. We use the dynamic graph structure to capture varying spatial dependencies between traffic nodes. In summary, our contributions are summarized as follows:

- We propose a POI-based Double-deck Graph Convolution Network to predict the traffic flow. Unlike existing methods, we use the functional similarity matrix to construct a regional traffic network that assists in node-level prediction.
- A POI-based dynamic graph structure is implemented to model time-varying spatial correlation. Our method extracts spatial relationships on time slices from the dynamic graph structure.
- Currently, most traffic datasets lack POI as supplementary. We augment two existing datasets by collecting and integrating POI data proximal to each road node. We evaluate our proposed model on these two real-world datasets, and it has the most advanced prediction accuracy. The visualization of the POI relationships demonstrates their dynamic influences.

2 Related Work

2.1 Traffic Forecasting

Many recent studies use GCN to capture the spatial feature of the graph structure of the traffic network. Wu *et al.* [28] uses a learnable matrix to get the spatial relationships. Guo *et al.* [6] learns an optimized graph to reveal the latent relationship between nodes. More works focus on how to get spatio-temporal relationships. Han *et al.* [9] designs a dynamic graph construction to learn the spatial dependencies. Cirstea *et al.* [4] integrates the attention mechanism to generate a location-specific and time-varying model. Song *et al.* [21] explores the joint dependencies of data in space and time. Jiang *et al.* [12] applies spatial self-attention to capture the dynamic spatial dependencies. Ji *et al.* [11] proposes a physics-guided deep learning model to cast the physical mechanism of traffic flow. Jiang *et al.* [13] uses meta-graph learning as a graph structure on spatio-temporal data. Jin *et al.* [14] designs an automated dilated spatiotemporal graph module to capture the short and long term relationships. Liu *et al.* [17] takes the historical time steps as input and incurs the spatial information into the dependency embedding. Besides, some studies use potential geographic structure information to assist in traffic forecasting. Guo *et al.* [7] uses the geographical distance between nodes to get the regional traffic network. Sun *et al.* [22] uses the hypergraph of the traffic flow to capture the dynamic spatial-temporal feature.

2.2 POI-based Traffic Forecasting

POI is an inherent feature of transportation networks and can assist in traffic flow prediction [30]. Researchers use the distribution of POI to calculate the

similarity matrix between nodes. Some studies use the POI embedding to represent the geographical feature of the traffic network [23, 10]. They use the similarity functions to measure the differences between traffic nodes, e.g., cosine similarity [5, 16] and JS divergence [25]. Some researchers conduct more detailed modeling of POI. Lv uses Term Frequency-Inverse Document Frequency (TF-IDF) to calculate the functional relationship between nodes [19]. Zhang uses the TF-IDF equation to get the inherent region properties [29]. Zheng proposes the Inherent Influence Factor (IIF) to calculate the inherent influence of POIs on traffic flow [31].

Unlike current POI methods, we designs a novel framework to capture the potential spatial relationships. Instead of static POI modeling, we leverage trainable matrices and tensor decomposition to describe spatial relationships. Besides mining correlations between nodes, our model constructs regional traffic networks to assist node prediction.

3 Preliminaries

Traffic flow forecasting is a time series modeling problem. It uses historical traffic features (e.g., flow, speed) to predict future traffic features. The signal from node i at time t is denoted as $x_t^i \in \mathbb{R}^D$, where D is the number of the input features. The traffic data of all nodes at time t is denoted as $X_t = [x_t^1, x_t^2, ..., x_t^N] \in \mathbb{R}^{N \times D}$, where N is the number of the traffic nodes. The historical traffic data over T_1 time slices is defined as $X = [X_1, X_2, ..., X_{T_1}] \in \mathbb{R}^{N \times D \times T_1}$, and we predict the future traffic data $Y = [X_{T_1+1}, X_{T_1+2}, ..., X_{T_1+T_2}] \in \mathbb{R}^{N \times D \times T_2}$ over T_2 time slices. On the basis of traffic flow data, this article additionally uses POI data to assist in prediction. We use the matrix $POI \in \mathbb{R}^{N \times K}$ to represent the combination of the POIs of all nodes, where K is the number of POI categories. In this study, the traffic network can be modeled as a directed graph $G = (V, E, A)$, while $V \in \mathbb{R}^N$ is the road node sets; E is the set of edges; $A \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix of graph G. In this study, we use P_f and P_b to represent the forward and backward transition matrix of the traffic network, respectively.

4 Methodology

Figure 3(a) shows the architecture of PDGCN, consisting of a region network generation layer, L spatial-temporal layers, and an output layer. We calculate POI's IIF modeling to obtain a functional similarity matrix and use the spectral cluster method for functional region division, capturing hierarchical geographical traffic network relationships. Each layer has node-level and region-level ST blocks to extract features, with POI-based dynamic GCN in

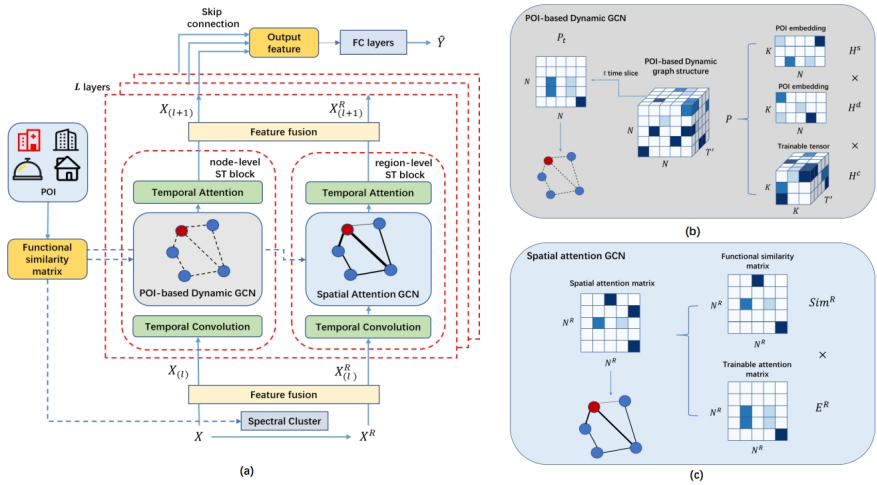


Figure 3: (a) The overall architecture of PDGCN. The framework processes the regional and node networks separately and mainly consists of two parallel parts: node-level spatiotemporal (ST) block and region-level ST block. (b) The structure of the POI-based dynamic GCN in the node-level ST block. It is used to capture the time-varying relationship between nodes. We use dashed lines in the graph to represent the connections between nodes. (c) The structure of the spatial attention GCN in the region-level ST block is used to mine spatial relationships between regions. The different thickness of edges in spatial attention GCN indicates the closeness of the relationship between nodes.

node-level ST block handling spatial correlations and spatial attention GCN in region-level ST block capturing spatial traffic features. After feature fusion, skip connection combines layer output and obtains prediction after several fully connected layers. We use mean absolute error (MAE) for training.

4.1 Traffic Node Functional Modeling

Traffic nodes sharing similar functionality have similar traffic patterns. To explore these potential traffic patterns, we use the POI distribution to model the traffic node functionality. We define the number of the k^{th} POI category of the n^{th} traffic node as $POI(n,k)$. We extend previous methods [31] to make it applicable to irregular graph networks. Node functionality is calculated using POI frequency Fre, density Den, and imbalance degree IBD. Frequency and density reflect POI aggregation. The imbalance degree uses Shannon entropy, weighting rarer POIs higher. Based on these three aspects, the functionality of the traffic node can be calculated as follows:

$$\begin{aligned} Fre(n,k) =& \frac{POI(n,k)}{\sum_k POI(n,k)}, \\ Den(n,k) =& \frac{POI(n,k)}{\sum_n POI(n,k)}, \\ IBD(n,k) =& 1 - \frac{-\sum_n Den(n,k) \times \log Den(n,k)}{\log N}, \\ IIF(n,k) =& Fre(n,k) \times Den(n,k) \times IBD(n,k). \end{aligned} \tag{1}$$

We merge the value of $IIF(n,k)$ to get the matrix $IIF(n) = [IIF(n,1), IIF(n,2), ..., IIF(n,K)] \in \mathbb{R}^K$ and $IIF = [IIF(1), IIF(2), ..., IIF(n)] \in \mathbb{R}^{N \times K}$. With the matrix IIF, we use the cosine similarity function to calculate the functionality matrix $Sim = cosine(IIF(i), IIF(j)) \in \mathbb{R}^{N \times N}$ between nodes.

4.2 POI-based Double-deck Graph Generating by Clustering

4.2.1 Region network generation

POI distribution naturally influences traffic flow. Nodes with similar POI distribution exhibit akin traffic patterns. To explore these patterns, we cluster nodes by POI distribution into regions and construct a regional traffic map. We use the matrix Sim as the adjacent matrix and calculate its Laplacian matrix $L_{sym} = I - Deg^{-\frac{1}{2}}\ Sim\ Deg^{-\frac{1}{2}}$ as the input of the spectral cluster, where I denotes the identity matrix, Deg denotes the degree matrix of Sim [24].

We could divide all the traffic nodes into N^R part, where N^R is the number of the traffic regions. After clustering, nodes with similar functions will belong to the same region. For example, all residential areas will belong to the same region. We can notice that the number of regions may not correspond one-to-one with the number of POI types. In other words, the classified regions may have multiple features at the same time, such as nodes with both mall and residential characteristics being classified into the same class. From the historical traffic data $X = [X_1, X_2, ..., X_{T_1}] \in \mathbb{R}^{N \times D \times T_1}$, we can generate the historical data of regions $X^R = [X_1^R, X_2^R, ..., X_{T_1}^R] \in \mathbb{R}^{N^R \times D^R \times T_1}$, where D^R is the feature number of the traffic regions.

4.2.2 Feature fusion block

Naturally, we can get the relational matrix $Bel \in \mathbb{R}^{N \times N^R}$ between nodes and regions:

$$Bel_{i,j} = \begin{cases} 1, \text{if the node } i \text{ is in the region } j; \\ 0, \text{else.} \end{cases} \tag{2}$$

The feature fusion block combines the features between nodes and regions. We use a dynamic transfer matrix based on the attention mechanism [7] with the following form:

$$\begin{aligned} E^d =& norm(V\sigma((F)^T U_1)U_2((F^R U_3)^T) + b)), \\ Bel^d =& \sigma(E^d) * Bel, \end{aligned} \tag{3}$$

where $V, b \in \mathbb{R}^{T_1^l \times T_1^l}$, $U_1 \in \mathbb{R}^N$, $U_2 \in \mathbb{R}^{C_l \times N}$, $U_3 \in \mathbb{R}^{C_l}$ are learnable parameters, C_l is the number of channels of the input data in the l^{th} layer, T_1^l is the length of the temporal dimension in the l^{th} layer. F, F^R are the input of the node feature and the region feature. $norm$ indicates a normalization operation. With the attention matrix Bel^d, we can get the output feature F_{out} for nodes and F_{out}^R for regions as follows:

$$F_{out} = Concat(F, Bel^d * F^R), F_{out}^R = F^R. \tag{4}$$

4.3 POI-based Dynamic Graph Structure

The adjacency matrix in graph convolution denotes spatial relationships between nodes. The existing methods mainly utilize static POI distribution modeling. However, the functional similarity between nodes changes over time. For example, the traffic flow between the industrial and residential areas peaks in the morning and evening. We can use the trainable matrix to mine the dynamic relationship between nodes using POI distribution. With traffic periodicity, we can use the same trainable functional similarity matrix at each slice of the day. The complexity of the algorithm will be $O(T^{'} \times N^2)$, where $T^{'}$ is the number of the time slice in one day. However, this method has a large number of training parameters. In this paper, we design a POI-based dynamic graph structure for time-varying functional similarity. Travel is purposeful between initial and destination POIs. Therefore, we model dynamic POI similarity and combine tensors to get node similarity.

We use a trainable tensor $H^c \in \mathbb{R}^{K \times K \times T^{'}}$ to model the temporal relationship between POIs. Extract each time segment, we can obtain the relationship matrix $H^{c(t)} \in \mathbb{R}^{K \times K}$ between POIs, where $t \in \{1, 2, ..., T^{'}\}$. We use two static matrices $H^s = H^d = IIF \in \mathbb{R}^{N \times K}$ to represent the POI distribution of the starting point and destination, respectively. With the above three tensors, we can generate the tensor $P \in \mathbb{R}^{N \times N \times T^{'}}$ to describe the dynamic functional similarity between nodes. The structure of the dynamic graph can be seen in Figure 3(b). The complexity of the algorithm is $O(T^{'} \times K^2)$. The math forms are as follows:

$$P_{i,j,t} = softmax(\sum_{q=0}^{N} \sum_{r=0}^{N} H_{q,r,t}^c H_{i,q}^s H_{j,r}^d). \tag{5}$$

4.4 The Node-level ST Block

The node-level ST block consists of Temporal Convolution, POI-based dynamic GCN and Temporal Attention. The input of the node-level ST block in the l^{th} layer is $X_{(l)}$. The output of three parts are $X^1_{(l)}$, $X^2_{(l)}$, $X^3_{(l)}$ separately. The structures of the three parts are as follows:

4.4.1 Temporal Convolution

Temporal convolution is aimed at extracting temporal features. We use the dilated causal convolution to expand the receptive field of the convolution [28]. The dilated causal convolution can help generate the common features for both nodes and regions. The input of the temporal convolution is $X_{(l)}$. To distinguish the data between node and region, We divide it into two parts and activate them separately. The math forms are as follows:

$$\begin{aligned}(\alpha_1, \alpha_2) =& split(W \star X_{(l)}),\\ X^1_{(l)} =& tanh(\alpha_1) \odot \sigma(\alpha_2),\end{aligned} \tag{6}$$

where $\odot$ denotes an element-wise multiplication operator, σ is a sigmoid activation function, $\star$ is the dilated convolution operation and W is learnable parameters of convolution filters.

4.4.2 POI-based Dynamic GCN

The adjacency matrix represents spatial relationships between nodes. Existing methods mainly employ static matrices for spatial relations. In this section, we utilize the POI-based dynamic graph structure proposed above to capture the time-varying spatial dependencies between nodes.

Given input $X^1_{(l)}$, we can get the time slice t of the data by division. Through the tensor $P \in \mathbb{R}^{N \times N \times T'}$ calculated by POI above, we substitute t into the time dimension and obtain the tensor $P_t \in \mathbb{R}^{N \times N}$, which represents the POI-based adjacency matrix on time slice t. The math forms of the dynamic GCN are as follows:

$$X^2_{(l)} = \sum_{m=0}^{M} (P^m_f X^1_{(l)} W_{k_1} + P^m_b X^1_{(l)} W_{k_2} + P^m_t X^1_{(l)} W_{k_3}), \tag{7}$$

where W_{k_1},W_{k_2},W_{k_3} is learnable matrix, P_f is the forward transition matrix of the traffic network, P_b is the backward transition matrix of the traffic network, P^m_f represents the power series of the transition matrix.

4.4.3 Temporal Attention

We use an attention mechanism to capture the different correlations between nodes [8]. The input of the temporal convolution is $X^2_{(l)}$. The math forms are as follows:

$$\begin{aligned} E =& V\sigma((X)^T U_1)U_2((XU_3)^T) + b), \\ X^3_{(l)} =& softmax(E) \times X^2_{(l)}, \end{aligned} \tag{8}$$

where $V, b \in \mathbb{R}^{T_1^l \times T_1^l}$, $U_1 \in \mathbb{R}^N$, $U_2 \in \mathbb{R}^{C_l \times N}$, $U_3 \in \mathbb{R}^{C_l}$ are learnable parameters. X is the traffic data. C_l is the number of channels of the input data in the l^{th} layer, T_1^l is the length of the temporal dimension in the l^{th} layer.

4.5 The Region-level ST Block

The region-level ST block consists of Temporal Convolution, Spatial Attention convolution, and Temporal Attention. The input of the block in the l^{th} layer is $X^R_{(l)}$. The output of three parts are $X^{R,1}_{(l)}$, $X^{R,2}_{(l)}$, $X^{R,3}_{(l)}$ separately. After clustering, nodes with similar functions form the regions, so the relationship between regions can be directly modeled using functional similarity. We use the dynamic GCN to capture spatial traffic features.

4.5.1 Spatial Attention GCN

In node functional modeling above, we calculate the matrix $Sim \in \mathbb{R}^{N \times N}$ to represent the functional similarity between nodes. We use the same method to calculate the $Sim^R \in \mathbb{R}^{N^R \times N^R}$. The input of the region feature is $X^{R,1}_{(l)}$. The superscript R represents the region data. As the relation between regions is changeable, we use the spatial attention mechanism to capture the dynamic relationships (similar to Equation 3). The structure of the spatial attention graph can be seen in Figure 3(c). The math forms are as follows:

$$\begin{aligned} E^R &= norm(V\sigma((X^R)^T U_1)U_2((X^R U_3)^T) + b)), \\ X^{R,2}_{(l)} &= \sum_{m=0}^{M} ((P_f^R)^m X^{R,1}_{(l)} W_{k_1} + (P_b^R)^m X^{R,1}_{(l)} W_{k_2} \\ &\quad + (Sim^R * E^R)^m X^{R,1}_{(l)} W_{k_3}). \end{aligned} \tag{9}$$

5 Experiments

5.1 Datasets

Currently, most datasets of traffic flow provide only traffic volume or velocity data, lacking supplementary information such as POI. This study scraped POI data proximal to each road node from Gaode AMAP Inside (https://lbs.amap.com/) and systematically organized said data. Two traffic speed datasets used in our experiments are collected by Didi Chuxing GAIA Initiative (https://gaia.didichuxing.com) in JiNan and XiAn cities of China. These datasets contain the average speed of road segments sampled per 10 minutes. There are 561 and 792 road segments (nodes) for the JiNan and XiAn datasets. The total sample number of the two datasets is 52286 each. We use the preprocessed dataset for experiments [7]. We split the data with a ratio of 6:2:2 at two datasets into training sets, validation sets, and test sets. We divide all POIs into 23 classes (listed in Table 1). There are 72722 and 117865 POIs on JiNan and XiAn datasets. The resulting dataset will be made publicly available. The specific information of two datasets are summarized in Table 2.

Table 1: POI category taxonomy.

ID	Category	ID	Category
01	Auto Service	13	Government Organization
02	Auto Dealers	14	Science/Culture Education Service
03	Auto Repair	15	Transportation Service
04	Motorcycle Service	16	Finance & Insurance Service
05	Food & Beverages	17	Enterprises
06	Shopping	18	Road Furniture
07	Daily Life Service	19	Place Name & Address
08	Sports & Recreation	20	Public Facility
09	Medical Service	21	Incidents and Events
10	Accommodation Service	22	Indoor facilities
11	Tourist Attraction	23	Pass facilities
12	Commercial House		

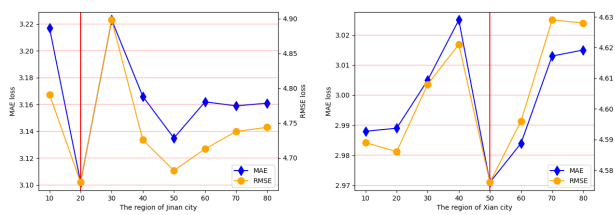
Table 2: Datasets description and statistics.

Datasets	#Nodes	#POINums	#TimeSteps	#MissingRatio
JiNan	561	72722	52309	6.511%
XiAn	792	117865	52309	5.014%

5.2 Experiments Settings

For the PDGCN experiment, the model's layer number is $L = 2$. We set the regional feature sizes $D^R = 2$ with the average and minimum speed in the

region. We set the diffusion step of GCN $M = 2$, the input channel of GCN $c_{in} = 32$, the output channel of GCN $c_{out} = 512$ and the hidden representation $h = 32$. We set the number of regions $N^R = 20$ in the Jinan dataset and $N^R = 50$ in the Xian dataset. We conduct experiments using different N^R and select the parameter with the best performance on the validation set. The comparison result can be seen in Figure 4. Due to the two datasets collecting data every 10 minutes, 144 data are obtained in one day, so we set the time slice number $T^{'} = 144$. For other baselines, in GCN-based methods including GWNET [28], MTGNN [27], HGCN [7], DMSTGCN [9], DeepSTD [31], We set the diffusion step of GCN $M = 2$, the input channel of GCN $c_{in} = 32$, the output channel of GCN $c_{out} = 512$ and the hidden representation $h = 32$. In ST-WA [4], we set the hidden representation $h = 32$. We use a 3 layer fully connected network with 32 neurons for the encoder. We use a 3 layer fully-connected network with 16, 32, and 5 neurons for the decoder. We use 2 fully connected layers, each with 512 neurons for the predictor. When computing the attentions, we utilize multi-head attention with a total of 8 heads. In MegaCRN [13], each RNN layer in encoder and decoder has 64 units and the memory bank has 20 meta-nodes with 64-dimension. For all methods, we set the input time interval $T_1 = 12$ and the forecasting time interval $T_2 = 12$. We use Adam optimizer [15] for the training with an initial learning rate of 0.001. The batch size is 64. We run the model for 70 epochs. The model is trained in Pytorch 1.12.1 environment with one GeForce RTX 3090 GPU.



(a) Jinan (b) Xian

Figure 4: MAE and RMSE loss of 1 hour forecasting with different N^R on the two datasets.

5.3 Comparison with Baselines

The proposed method is compared with the several baselines on the Jinan and Xian datasets as below: Historical Average (HA), LSTM [20], GWNET [28], MTGNN [27], HGCN [7], DMSTGCN [9] (We only use the traffic speed as input), MegaCRN [13], ST-WA [4], DeepSTD [31] (Since the DeepSTD

method is not open source, we conduct experiments in the GWNET using POI models).

Tables 3 and 4 show the performance of the proposed model and the baselines on two datasets for 30, 60, and 120 minutes predictions. On the Jinan dataset, our method performs best in horizons 3 and 6, while horizon 12 is close to HGCN. For Xian, PDGCN performs better in all horizons. Overall, deep learning models surpass traditional statistics like HA. RNNs like LSTM underperform due to the inability to learn spatial relationships. Our method surpasses GCN-based methods, including DMSTGCN and HGCN. The lack of traffic speed data potentially impairs the performance of DMSTGCN. Compared to HGCN, we introduce POI to enhance spatial node correlations. Compared to recent methods, including ST-WA and MegaCRN, our method performs better. Compared to other POI methods, our method performs better, which means the dynamic GCN and the spatial attention GCN can capture the spatio-temporal feature better.

Table 3: Performance comparison on the JiNan datasets. The superscript * denotes the method that uses POI.

Method	Year	horizon 3			horizon 6			horizon 12		
		MAE	MAPE(%)	RMSE	MAE	MAPE(%)	RMSE	MAE	MAPE(%)	RMSE
HA	-	5.690	20.02	7.600	5.690	20.02	7.600	5.690	20.02	7.600
LSTM[20]	-	3.196	12.83	4.830	3.661	14.90	5.466	4.290	17.52	6.255
GWNET[28]	IJCAI19	2.933	11.47	4.404	3.194	12.47	4.740	3.450	13.64	5.151
MTGNN[27]	KDD20	3.013	11.83	4.499	3.284	12.94	4.848	3.637	14.30	5.276
HGCN[7]	AAAI21	2.887	11.49	4.370	3.112	12.50	4.680	**3.412**	**13.62**	**5.041**
DMSTGCN[9]	KDD21	2.906	11.62	4.394	3.210	12.91	4.790	3.577	14.60	5.268
ST-WA[4]	ICDE22	3.620	13.41	6.270	3.830	14.26	6.440	4.080	15.24	6.550
MegaCRN[13]	AAAI23	2.940	11.67	4.458	3.172	12.65	4.775	3.475	13.74	5.142
DeepSTD* [31]	TITS19	2.919	11.59	4.411	3.145	12.60	4.716	3.443	13.63	5.076
PDGCN*	-	**2.877**	**11.42**	**4.358**	**3.103**	**12.43**	**4.665**	**3.412**	**13.61**	5.057

Table 4: Performance comparison on the Xian datasets. The superscript * denotes the method that uses POI.

Method	Year	horizon 3			horizon 6			horizon 12		
		MAE	MAPE(%)	RMSE	MAE	MAPE(%)	RMSE	MAE	MAPE(%)	RMSE
HA	-	6.020	21.80	8.160	6.020	21.80	8.160	6.020	21.80	8.160
LSTM[20]	-	3.151	11.92	4.816	3.692	14.22	5.506	4.510	17.31	6.509
GWNET[28]	IJCAI19	2.771	10.51	4.305	3.038	11.70	4.671	3.349	13.01	5.082
MTGNN[27]	KDD20	2.812	10.64	4.314	3.082	11.92	4.674	3.486	13.70	5.175
HGCN[7]	AAAI21	2.751	10.52	4.275	2.982	11.62	4.576	3.251	12.71	4.910
DMSTGCN[9]	KDD21	2.801	10.67	4.329	3.101	12.00	4.712	3.516	13.73	5.220
ST-WA[4]	ICDE22	3.300	12.02	5.900	3.460	12.91	6.020	3.680	13.97	6.150
MegaCRN[13]	AAAI23	2.753	10.46	4.283	3.019	11.75	4.633	3.471	13.67	5.181
DeepSTD*[31]	TITS19	2.780	10.50	4.293	3.043	11.93	4.652	3.393	13.40	5.095
PDGCN*	-	**2.743**	**10.36**	**4.268**	**2.966**	**11.41**	**4.569**	**3.214**	**12.47**	**4.874**

5.4 Experiments Analysis

5.4.1 Evaluation of hyper-parameter

N^R denotes the number of regions, which impacts the scale of the regional traffic network. As shown in Figure 4, the performance on two datasets first fluctuates and then stabilizes at a certain level. Further increases will make the regional feature closer to the node feature, which loses its heterogeneity. From the results, we stack $N^R = 20$ on the Jinan dataset and $N^R = 50$ on the Xian dataset.

5.4.2 Evaluation of the Dynamic GCN and the Spatial attention GCN

We conduct an ablation study on the Dynamic GCN (D) and the Spatial Attention GCN (SA) to verify the performance of the proposed model. We test the network with the different block combinations. Table 5 shows the different variants of our model on the Jinan and Xian datasets.

Table 5: Results of ablation study of the Dynamic GCN (D) and the Spatial attention GCN (SA) on the Jinan and Xian datasets.

Dataset	Components		MAE		
	D	SA	horizon 3	horizon 6	horizon 12
Jinan			2.910	3.142	3.460
	✓		2.895	3.137	3.441
		✓	2.896	3.126	3.425
	✓	✓	**2.878**	**3.102**	**3.412**
Xian			2.764	3.004	3.326
	✓		2.765	2.999	3.291
		✓	2.751	2.979	3.266
	✓	✓	**2.743**	**2.966**	**3.214**

It can be concluded that PDGCN has the best performance compared with the models removing specific components. Concretely, Models with only Dynamic GCN or Spatial Attention GCN surpass the baseline without either, indicating the effectiveness of Dynamic GCN and Spatial Attention GCN. Moreover, PDGCN excels over variants with single modules, illustrating that the interaction of the dynamic GCN and the Spatial Attention GCN benefits traffic prediction.

5.4.3 Evaluation of different cluster methods

Functional similarity matrices from various POI modeling and similarity functions are input to generate regional traffic. "POI embedding", "tf_idf" and "IIF" indicate that we adopt different POI modeling methods: POI embedding, "tf_idf" [29] and "IIF". "cos", "js" and "euci" indicate cosine similarity, Jensen-Shannon divergence and euclidean distance as similarity metrics.

Table 6 shows the different cluster methods for different horizons on two datasets. Compared to the no-cluster method, PDGCN performs better in all horizons, which proves the effectiveness of the POI-based cluster method. We can observe that "POI embedding + cos" outperforms "POI embedding + euci" and "POI embedding + js", which indicates that cosine similarity is more beneficial in extracting node functional information. The last three experiments show that IIF modeling performs better in horizon 3 and 6 on two datasets, demonstrating its efficacy in modeling POI distribution.

Table 6: Results of ablation study of different cluster methods on the Jinan and Xian datasets."POI embedding", "tf_idf" and "IIF" indicate different POI modeling methods: POI embedding, "tf_idf" [29] and "IIF". "cos", "js" and "euci" mean cosine similarity, Jensen-Shannon divergence and euclidean distance in the study.

Dataset	Method	MAE		
		horizon 3	horizon 6	horizon 12
Jinan	no cluster	2.890	3.150	3.430
	POI embedding + js	2.908	3.134	3.404
	POI embedding + euci	2.940	3.182	3.449
	POI embedding + cos	2.905	3.125	**3.399**
	tf_idf + cos	2.915	3.165	3.476
	IIF + cos (PDGCN)	**2.878**	**3.102**	3.412
Xian	no cluster	2.764	3.004	3.326
	POI embedding + js	2.769	3.001	3.304
	POI embedding + euci	2.764	2.993	3.267
	POI embedding + cos	2.754	2.975	3.218
	tf_idf + cos	2.754	2.974	3.261
	IIF + cos (PDGCN)	**2.743**	**2.966**	**3.214**

5.4.4 Case Study

We learn POI dynamics via dynamic graph convolution and use linear combinations to generate node dependencies. We select a case from the dataset Jinan to confirm the rationality of the dynamic graph convolution model. Historical average traffic speed from two different nodes in one day is illustrated in Figure 5(a). We extract the main POIs from the two nodes (node1: Transportation Service, node2: Finance & Insurance Service) and get the

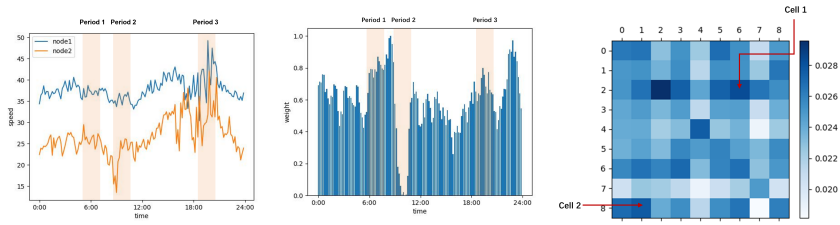


Figure 5: Illustration of the learned dynamic graph. (a) the average traffic speed of two nodes in one day. (b) the weight between two POIs in one day. In periods 1 and 3, the high inter-node weight indicates similar traffic patterns. In period 2, the weight is low in the learnable dynamic graph, while the traffic pattern is different. (c) The matrix is the part of the spatial attention matrix E^R for chosen regions. A deeper color indicates closer node relationships. The coordinate 0-8 means the main POI of nine selected regions: Auto Service, Food, Shopping, Daily Life Service, Sports, Medical Service, Tourist Attraction, Commercial House, and Enterprises. Cell 1 shows a close shopping-tourist relationship. Cell 2 reflects enterprise-food proximity.

weight between two POIs after normalization in Figure 5(b). In period 1, the average speed of the two nodes is relatively stable, which means the traffic pattern between nodes is similar. The weight between the two nodes is also high. In period 2, the speed on node 1 keeps stable, while the speed on node 2 decreases to the minimum. In this period, the weight is low in the dynamic graph. Period 3 has a rising speed and high inter-node weight.

The study uses the spatial attention model to describe the relationships between regions. Functional similarity provides regions grouping nodes with similar POI distribution, reflecting inherent patterns. The main POI in each region can reflect the inherent traffic pattern. We select nine regions from the Jinan dataset to confirm the rationality of the spatial attention model. The average spatial attention matrix between 9 regions is shown in Figure 5(c). The coordinate 0-8 means the main POI of nine selected regions. The deeper color indicates closer relationships. Cell 1 shows a close shopping-tourist relationship. Cell 2 reflects enterprise-food proximity.

6 Conclusions

This paper proposes a graph convolution network named PDGCN for traffic forecasting. We use the functional similarity matrix to construct a regional traffic network that assists in the node-level prediction. The proposed model uses dynamic GCN and spatiotemporal attention to capture the time-varying correlations between nodes. Experimental results on two real-world datasets

demonstrate that PDGCN outperforms state-of-the-art methods. Except for POI, other external factors such as weather and holidays can also impact traffic patterns. In future work, we will add other external factors to the existing network structure.

Acknowledgements

This work was supported in part by the major project of PCL under Grant PCL2023A08, in part by National Natural Science Foundation of China (61931008, 62272438), and Fundamental Research Funds for the Central Universities (E2ET1104).

References

[1] A. F. M. Agarap, "A neural network architecture combining gated recurrent unit (GRU) and support vector machine (SVM) for intrusion detection in network traffic data", in *Proceedings of the 2018 10th international conference on machine learning and computing*, 2018, 26–30.

[2] S. R. Chandra and H. Al-Deek, "Predictions of freeway traffic speeds and volumes using vector autoregressive models", in *Journal of Intelligent Transportation Systems*, Vol. 13, No. 2, 2009, 53–72.

[3] Y.-y. Chen, Y. Lv, Z. Li, *et al.*, "Long short-term memory model for traffic congestion prediction with online open data", in *IEEE 19th International Conference on Intelligent Transportation Systems*, 2016, 132–7.

[4] R.-G. Cirstea, B. Yang, C. Guo, *et al.*, "Towards spatio-temporal aware traffic time series forecasting", in *2022 IEEE 38th International Conference on Data Engineering*, 2022, 2900–13.

[5] X. Geng, Y. Li, L. Wang, *et al.*, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting", in *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33, No. 01, 2019, 3656–63.

[6] K. Guo, Y. Hu, Z. Qian, *et al.*, "Optimized graph convolution recurrent neural network for traffic prediction", in *IEEE Transactions on Intelligent Transportation Systems*, Vol. 22, No. 2, 2020, 1138–49.

[7] K. Guo, Y. Hu, Y. Sun, *et al.*, "Hierarchical graph convolution network for traffic forecasting", in *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35, No. 1, 2021, 151–9.

[8] S. Guo, Y. Lin, N. Feng, *et al.*, "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, 922–9.

[9] L. Han, B. Du, L. Sun, *et al.*, "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, 547–55.

[10] S. He and K. G. Shin, "Towards fine-grained flow forecasting: A graph attention approach for bike sharing systems", in *Proceedings of The Web Conference 2020*, 2020, 88–98.

[11] J. Ji, J. Wang, Z. Jiang, *et al.*, "STDEN: Towards physics-guided neural networks for traffic flow prediction", in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36, No. 4, 2022, 4048–56.

[12] J. Jiang, C. Han, W. X. Zhao, *et al.*, "PDFormer: Propagation Delay-aware Dynamic Long-range Transformer for Traffic Flow Prediction", in *arXiv preprint arXiv:2301.07945*, 2023.

[13] R. Jiang, Z. Wang, J. Yong, *et al.*, "Spatio-temporal meta-graph learning for traffic forecasting", in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, No. 7, 2023, 8078–86.

[14] G. Jin, F. Li, J. Zhang, *et al.*, "Automated dilated spatio-temporal synchronous graph modeling for traffic prediction", in *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", in *arXiv preprint arXiv:1412.6980*, 2014.

[16] Z. Li, N. D. Sergin, H. Yan, *et al.*, "Tensor completion for weakly-dependent data on graph for metro passenger flow prediction", in *proceedings of the AAAI conference on artificial intelligence*, Vol. 34, No. 04, 2020, 4804–10.

[17] D. Liu, J. Wang, S. Shang, *et al.*, "Msdr: Multi-step dependency relation networks for spatial temporal forecasting", in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, 1042–50.

[18] Y. Liu, H. Zheng, X. Feng, *et al.*, "Short-term traffic flow prediction with Conv-LSTM", in *9th International Conference on Wireless Communications and Signal Processing*, 2017, 1–6.

[19] M. Lv, Z. Hong, L. Chen, *et al.*, "Temporal multi-graph convolutional network for traffic flow prediction", in *IEEE Transactions on Intelligent Transportation Systems*, Vol. 22, No. 6, 2020, 3337–48.

[20] L. S.-T. Memory, "Long short-term memory", in *Neural computation*, Vol. 9, No. 8, 2010, 1735–80.

[21] J. Song, J. Son, D.-h. Seo, *et al.*, "ST-GAT: A Spatio-Temporal Graph Attention Network for Accurate Traffic Speed Prediction", in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, 4500–4.

[22] Y. Sun, X. Jiang, Y. Hu, *et al.*, "Dual dynamic spatial-temporal graph convolution network for traffic prediction", in *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 12, 2022, 23680–93.

[23] Y. Sun, G. Jiang, S. K. Lam, *et al.*, "Predicting Traffic Congestion Evolution: A Deep Meta Learning Approach.", in *Proceedings of the International Conference on International Joint Conferences on Artificial Intelligence.* 2021, 3031–7.

[24] U. Von Luxburg, "A tutorial on spectral clustering", in *Statistics and computing*, Vol. 17, 2007, 395–416.

[25] B. Wang, Y. Lin, S. Guo, *et al.*, "GSNet: learning spatial-temporal correlations from geographical and semantic aspects for traffic accident risk forecasting", in *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35, No. 5, 2021, 4402–9.

[26] C.-H. Wu, J.-M. Ho, and D.-T. Lee, "Travel-time prediction with support vector regression", in *IEEE transactions on intelligent transportation systems*, Vol. 5, No. 4, 2004, 276–81.

[27] Z. Wu, S. Pan, G. Long, *et al.*, "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, 753–63.

[28] Z. Wu, S. Pan, G. Long, *et al.*, "Graph wavenet for deep spatial-temporal graph modeling", in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019, 1907–13.

[29] M. Zhang, T. Li, Y. Li, *et al.*, "Multi-view joint graph representation learning for urban region embedding", in *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 2021, 4431–7.

[30] T. Zhang, L. Sun, L. Yao, *et al.*, "Impact analysis of land use on traffic congestion using real-time traffic and POI", in *Journal of Advanced Transportation*, Vol. 2017, 2017.

[31] C. Zheng, X. Fan, C. Wen, *et al.*, "DeepSTD: Mining spatio-temporal disturbances of multiple context factors for citywide traffic flow prediction", in *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 9, 2019, 3744–55.