

Shared Spatio-temporal Attention Convolution Optimization Network for Traffic Prediction

Pengcheng Li¹, Changjiu Ke², Hongyu Tu², Houbing Zhang², and Xu Zhang^{1,*}

Abstract

The traffic flow in an urban area is affected by the date, weather, and regional traffic flow. The existing methods are weak to model the dynamic road network features, which results in inadequate long-term prediction performance. To solve the problems regarding insufficient capacity for dynamic modeling of road network structures and insufficient mining of dynamic spatio-temporal features. In this study, we propose a novel traffic flow prediction framework called shared spatio-temporal attention convolution optimization network (SSTACON). The shared spatio-temporal attention convolution layer shares a spatio-temporal attention structure, that is designed to extract dynamic spatio-temporal features from historical traffic conditions. Subsequently, the graph optimization module is used to model the dynamic road network structure. The experimental evaluation conducted on two datasets shows that the proposed method outperforms state-of-the-art methods at all time intervals.

Keywords

Optimization Graph, Shared Attention, Spatio-temporal Attention, Traffic Flow Forecasting

1. Introduction

Chongqing's traffic congestion ranks first during the rush-hour in China among the cities with more than four million cars introduced by the Amap and Baidu map. Traffic congestion may cause significant losses to cities. First, it increases residents' travel costs and wastes significant time. Second, it harms the environment and causes air pollution phenomena, such as haze. Third, traffic congestion may cause accidents and threaten people's lives. Accurate traffic flow prediction can help the government deal with these issues and distribute the traffic flow to alternative roads before traffic jams occur, which is crucial for urban development and human safety.

Traditional statistical methods have a limited ability to capture spatio-temporal correlations. For example, the historical average (HA) and autoregressive integrated moving average (ARIMA) models need to satisfy the stationarity hypothesis of each time series; however, they cannot function with random events. Recent research has proposed modeling spatial correlation with convolutional neural networks (CNN) and capturing temporal correlation with a recurrent neural network (RNN). In addition, some studies combined CNNs and RNNs to simulate spatio-temporal correlation and achieved satisfactory

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Manuscript received July 5, 2022; first revision September 27, 2022; second revision November 22, 2022; accepted December 27, 2022.

* Corresponding Author: Xu Zhang (zhangx@cqupt.edu.cn)

¹ Dept. of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China (1353276354@qq.com,

zhangx@cqupt.edu.cn)

² Chongqing Customs Technology Center, Chongqing Customs District, Chongqing, China (320110060@qq.com, 1123037746@qq.com, nibugailai@163.com)

performance. However, such methods have two limitations. First, they cannot deal with graph structures effectively. Second, they cannot model the dynamic spatial and temporal dependencies. The attention mechanism is efficient and flexible [1], which has achieved good performance in the natural language processing (NLP) field. In this study, we propose a layered shared spatio-temporal attention module to model the dynamic spatio-temporal dependency. Furthermore, we propose an optimization module that uses the optimized node embedding to learn and optimize the graph structures. Finally, we evaluate the proposed method against eight state-of-the-art methods on two real-world datasets.

The main contributions of this paper are summarized as follows:

- We propose a novel network framework for traffic flow prediction that captures the dynamic spatio-temporal dependence using a layered shared spatio-temporal attention convolution block.
- An optimized node embedding learning optimization graph is proposed to learn the graph structure, even if there is no predefined graph structure.
- The experimental results indicate that the proposed model performs better than existing methods, particularly in terms of long-term prediction performance.

2. Related Work

A deep neural network obtains the nonlinear modeling ability of data using the activation function, which is in accordance with traffic data characteristics. Recent studies have proposed to use graph convolution neural network (GCN) for traffic prediction, which outperform the traditional CNN/RNN-based methods. In the graph-based method, the nodes represent road segments that contain traffic information. An adjacency matrix is used to represent the connectivity between nodes. The following graph-based methods have achieved promising results. An attention-based spatial-temporal graph convolutional network (ASTGCN) divides the data into hours, days, and weeks to capture multi-scale features [2]. Diffusion convolutional recurrent neural network (DCRNN) use an encoder-decoder structure combined with a bidirectional static graph [3]. A graph multi-attention network (GMAN) proposes a self-attention mechanism and predefined graphs [4]. The fully connected gated graph architecture (FC-GAGA) passes the static graph through a residual full connection layer without using the dynamic attributes between nodes [5]. Graph WaveNet proposes an adaptive matrix for learning graphs with a significant improvement [6]. A multivariate time series graph neural network (MTGNN) proposes a graph learning layer to learn dynamic graphs for GCN [7]. A dynamic graph convolutional recurrent network (DGCRN) uses an encoder to learn the dynamic graph of each time step and curriculum learning strategy to train the model [8]. A novel neural controlled differential equation (STG-NCDE) is proposed for sequential data [9]. Towards physics-guided neural network (STDEN) assumes that the traffic flow on the road network is driven by a potential energy field [10]. Accordingly, the spatio-temporal dynamic process of the potential energy field is modeled as a differential equation network. However, these methods cannot integrate static and dynamic graphs to capture additional features.

3. Methodology of SSTACON

All road segments in an urban road network can be regarded as one graph. Specifically, let graph $G = (V, E, A)$. Here, V denotes a set of nodes representing different road segments and E denotes the set of

edges that indicate that the two road segments are reachable. $A \in R^{N \times N}$ denotes the weighted adjacency, where N represents the number of nodes in G . Subsequently, the value of the adjacency matrix is calculated using the distance between road segments. Each node v_i contains a series of observation values $[v_i^1, v_i^2, \dots, v_i^T]$, $v_i^t \in R^D$, where D is the number of features, such as time in days and speed. T is the sequence length. The graph signal composed of all nodes at time step t is denoted as $v^t \in R^{N \times D}$. Thus, the objective is to predict the series of future values $\hat{Y} = [v^{P+1}, v^{P+2}, \dots, v^{P+Q}]$ based on the observed values of all nodes up to time P which can be denoted as $X = [v^1, v^2, \dots, v^P]$. We need to learn a nonlinear function f to map G and X to \hat{Y} , which is represented as:

$$f(X, G) \rightarrow \hat{Y} \tag{1}$$

The structure of SSTACON is illustrated in Fig. 1, which consists of L shared spatio-temporal attention convolution (SSTAC) layers with B shared spatio-temporal attention convolution blocks and an optimization graph module in each layer. The SSTAC layer uses the optimized adjacency matrix and historical traffic data as inputs. There is a residual connection between each layer, and each SSTAC block has graph convolution layers and time dilation convolution layers. All blocks in each layer share the same spatial and temporal attention so that each layer can learn different spatio-temporal dependency patterns. The adjacency matrix learned by the graph optimization module is used as the input of the GCN. The key idea of the graph optimization module is to use better node embedding to obtain a better graph structure. The optimized graph module follows the settings reported in existing research [11]. To obtain good initial node embedding, we used the Nodevec algorithm to extract the embedded features. Nodevec is a network learning algorithm that aggregates the information of node neighborhoods in the network through depth-first and breadth-first search approaches, which are very consistent with the state of road traffic. When optimizing the graph, a cosine similarity mechanism is used to learn the correlation between nodes.

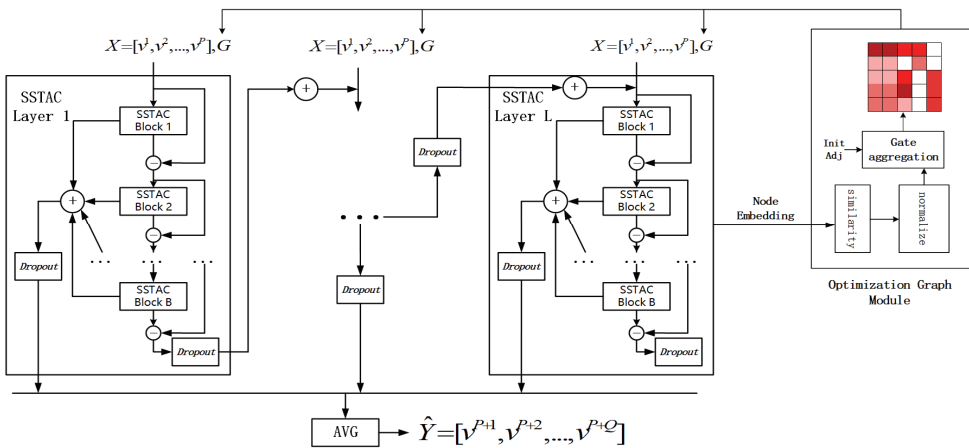


Fig. 1. Architecture of SSTACON.

3.1 Shared Spatio-temporal Attention Convolution Block

As shown in Fig. 2, each SSTAC block has the same structure, including the optimized GCN, time dilation convolution, shared temporal attention and shared spatial attention. Xb^{b-1} denotes the input of

the block b in the layer L . After the time dilation convolution, the time dimension is not equal to the input time dimension. Therefore, the full connection layer is used to modify the dimension. We express the output of shared spatial attention as $satt_l^b \in R^{T \times N \times N}$ and output of shared temporal attention as $tatt_l^b \in R^{N \times T \times T}$. Then, the hidden state is merged and passed through the full connection layer of two different output dimensions. The input of the next block is denoted as Xb_l^b , and part of the final output is denoted as Yf_l^b . To avoid model overfitting during the training process, we randomly selected some nodes with a probability of 0.4.

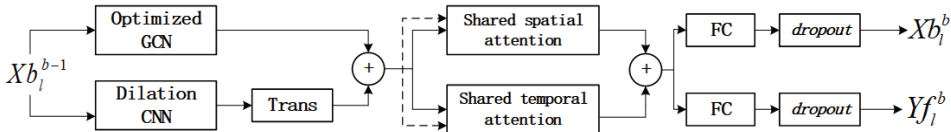


Fig. 2. Shared spatio-temporal attention convolution (SSTAC) block.

3.1.1 Optimized GCN

A GCN is a type of graph neural network that can aggregate the neighborhood information of a node. The formula is stated as follows:

$$g_\theta(L)X = \sum_{k=0}^{K-1} \theta_k T_k(A_o)X \quad (2)$$

where $\theta_k \in R^{hid \times hid}$ is a learnable parameter, hid is the hidden layer dimension, and $T_k(A_o)$ is a Chebyshev polynomial of order k . The topological structure of road networks is highly dynamic. The traffic condition of a road is influenced by its connected road segments. Therefore, instead of using the original definition of the scaled Laplacian matrix, $A_o \in R^{N \times N}$ is used, which is an optimized graph adjacency matrix. The optimization graph module is described in detail in Section 3.2.

3.1.2 Time dilation convolution & shared attention

Time dilation convolution (TDC). TDC uses four two-dimensional convolutions to extract the time features of different scales. When the dilation factor increases, it fills the time dimension of the data and increases the length of the sequence, which results in a large amount of noise data entering the network during training and reduces model performance. Thus, we limit the dilation factor to 1 and 2. We define the value of the dilation factor as the number of layers mod 2. Shared attention consists of shared spatial attention and shared temporal attention.

Shared spatial attention. The traffic conditions of road segments are influenced by the traffic conditions of other road segments, which denotes a mutual influence. To model dynamic global spatial features, we designed shared spatial attention to adaptively extract the correlation between road networks. To increase stability, we used a multi-head approach, which is formulated as follows:

$$\begin{aligned} h_l^b &= GCN(Xb_l^{b-1}) + TDC(Xb_l^{b-1}) \\ value &= cat(split(Vh_l^b + b_v)) \end{aligned} \quad (3)$$

where $h_i^b \in R^{P \times N \times hid}$ represents the sum of the output of GCN and TDC and *split* operation divides the input into multiple matrices for parallel calculation. $Q \in R^{hid \times hid}$, $K \in R^{hid \times hid}$, $V \in R^{hid \times hid}$ and $b_Q \in R^{hid}$, $b_H \in R^{hid}$, $b_V \in R^{hid}$, denote the parameters that can be learned. *Cat* operation concatenates multiple matrices into a certain dimension. Then, we use $query = cat(split(Qh_i^b + b_Q))$ and $key = cat(split(Kh_i^b + b_H))$ to calculate the attention score as follows:

$$\begin{aligned} score &= Softmax(matmul(query, key^T)/\sqrt{d}) \\ satt_i^b &= matmul(score, value) \end{aligned} \quad (4)$$

Specifically, the correlation between vertex v_i and vertex v_j is described in detail as follows:

$$score(v_i, v_j)' = \frac{query_{v_i} \bullet key_{v_j}^T}{\sqrt{d}}, \quad score(v_i, v_j) = \frac{exp(score(v_i, v_j)')}{\sum_{j=1}^N exp(score(v_i, v_j)')} \quad (5)$$

where \bullet represents the vector inner product operation and $d = hid/head$. In this study, we set $head = 8$. Specifically, each layer has different Q, K, and V values. The intermediate state outputs of all the SSTAC blocks in each layer pass through the same shared spatial and shared temporal attention, as denoted by the dotted line in Fig. 2.

Shared temporal attention. The structures of shared temporal and shared spatial attention are the same. However, the main difference is that, when calculating attention, here the temporal axis is used instead of the spatial axis.

3.2 Optimization Graph Module

The optimization graph module adaptively learns the optimized graph adjacency matrix in a data-driven manner, which can effectively capture the dynamic spatial characteristics of the road network. The core idea is to optimize the adjacency matrix of the graph using optimized node feature vectors. In this study, we encapsulate the node feature vectors as network parameters so that the node feature vectors are trained and updated together with other network parameters. Then, we use the weighted cosine similarity to calculate the similarity between nodes as follows:

$$s_{i,j}^p = CosSimilarity(W_p \odot vec_i, W_p \odot vec_j), \quad s_{i,j} = \frac{1}{m} \sum_{p=1}^m s_{i,j}^p \quad (6)$$

where $W_p \in R^{dim}$ is a weighted parameter vector that can be learned, $vec_i, vec_j \in R^{dim}$ represents a feature vector of nodes, \odot is the Hadamard product, and *CosSimilarity* calculates cosine similarity of the two vectors. $s_{i,j}^p$ represents the p^{th} similarity vector. Specifically, we use m weight parameter vectors to calculate m independent similarity vectors, use their average as the final similarity, and obtain a similarity matrix $s_{i,j} \in R^{N \times N}$.

We adopt the Nodevec algorithm on the predefined adjacency matrix to obtain a good initial node feature. p in Nodevec is set to 1 and q is set to 2. Fig. 1 shows that the similarity matrix is input into a double-gated aggregation mechanism that controls how much of the learned graph structure can be updated to the current graph. Therefore, the state of part of the original graph structure can be retained, and deep hidden relationships can be learned based on the following formula:

$$A_s = \frac{S_{i,j}}{\sum_j S_{i,j}}, A_o = \mu(\lambda A + (1 - \lambda)A_s) + (1 - \mu)A^{(1)} \quad (7)$$

where A_s is obtained by line normalization of the similarity matrix, A is a predefined graph, and $A^{(1)}$ is calculated by cosine similarity and normalization when the model is initialized. λ and μ are hyper parameters that control the proportion of the currently optimized graph structure in the graph adjacency matrix.

The embedded features of the nodes are updated along during the back-propagation process. The obtained optimized node features are used to calculate an optimized graph suitable for the next traffic flow forecast in the optimized graph module through weighted cosine similarity and double-gated aggregation. This process continues with model training and keeps learning appropriate graphs that align with the dynamic traffic flow in reality.

4. Experiments

We evaluate the proposed method using eight state-of-the-art methods on two public datasets, namely, METR-LA and PEMS-BAY. The detailed experimental setup is as follows. There are three SSTAC layers with four blocks in each layer. Each block has two outputs: the hidden size of Xb_t^b is 32, and hidden state size of Yf_t^b is 64. The order of the optimized GCN is set to 2. The values of λ and μ are both 0.9, and the number of weighted parameter vectors $m = 8$. The initial learning rate is 0.001, which is reduced to one-tenth of the current learning rate every 10 epochs. The number of training epochs is 100. The batch sizes for the PEMS-BAY and METR-LA datasets are set as 16. We use MAE as the loss function of our model and AdamW optimizer for training. The proposed model is implemented using PyTorch 1.4.0 on a server with a 32-G memory NVIDIA Telsa V100. We repeated the experiment five times and reported the average values of the evaluation metrics. The early stopping is set to 20 epochs.

4.1 Experiment Evaluation

Table 1 shows the traffic forecasting results for different methods of 15, 30, and 60 minutes. SSTACON achieves state-of-the-art performance in all datasets and time intervals. DCRNN [3] uses a distance-based graph that is time-invariant and inconsistent with the reality of the peak period. GMAN [4] entirely depends on static graphs, which ignore the dynamic properties of road networks. MTGNN [7] and Graph WaveNet benefit from the adaptive graph, but their adaptive graphs remain static and cannot capture the dynamic spatial dependence. FC-GAGA [5] does not use road network topology information. Therefore, its performance is inferior to those of other graph-based methods. It is difficult to model the long-term dependence of DGCRN [8] because it uses an encoder-decoder system. PM-MemNet [10] assumes that the traffic flow is driven by a potential energy field, but this assumption is inconsistent with the actual situation. Owing to the limited representation ability of the model, the ASTGCN [2] does not perform well on the two datasets. Our method uses shared attention to learn the global dependency between roads and uses a double-gated mechanism to combine static and dynamic graphs. However, the proposed SSTACON method requires considerable training time. It requires 4.5, 9.2, 2, and 1.5 times of the training time required by GMAN, MTGNN, DGCRN, and DCRNN, respectively, on the PEMS dataset, while requiring more GPU memory.

Table 1. Performance of different baselines on METR-LA and PEMS- BAY datasets

Model	METR / PEMS								
	15 min			30 min			60 min		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
DCRNN	2.77/1.38	5.38/2.95	7.30/2.90	3.15/1.74	6.45/3.97	8.80/3.90	3.60/2.07	7.60/4.74	10.50/4.90
WaveNet	2.69/1.30	5.15/2.74	6.90/2.73	3.07/1.63	6.22/3.70	8.37/3.67	3.53/1.95	7.37/4.52	10.01/4.63
ASTGCN	4.86/1.52	9.27/3.13	9.21/3.22	5.43/2.01	10.61/4.27	10.1/4.48	6.51/2.61	12.52/5.42	11.64/6.00
GMAN	2.80/1.34	5.55/2.91	7.41/2.86	3.12/1.63	6.49/3.76	8.73/3.68	3.44/1.86	7.35/4.32	10.07/4.37
MTGNN	2.69/1.32	5.18/2.79	6.86/2.77	3.05/1.65	6.17/3.74	8.19/3.69	3.49/1.94	7.23/4.49	9.87/4.53
FC-GAGA	2.70/1.33	5.24/2.82	7.01/2.82	3.04/1.66	6.19/3.75	8.31/3.71	3.45/1.93	7.19/4.40	9.88/4.48
DGCRN	2.62/1.28	5.01/2.69	6.63/2.66	2.99/1.59	6.05/3.63	8.02/3.55	3.44/1.89	7.19/4.42	9.73/4.43
PM-MemNet	2.66/-	5.28/-	7.06/-	3.02/-	6.28/-	8.49/-	3.40/-	7.24/-	9.88/-
SSTACON	2.49/1.20	4.45/2.21	6.55/2.58	2.83/1.46	5.19/2.81	7.75/3.24	3.30/1.80	6.11/3.50	9.49/4.22

The best values are shown in bold.
RMSE=root mean square error.

4.2 Ablation Analysis

We conduct an ablation study to verify the effectiveness of each component of the proposed model. *No optimized GCN* denotes removing the optimized GCN; *No TDC* denotes removing the time dilation convolution component and using only spatial model correlation; *No shared att* denotes removing the shared attention in the shared spatio-temporal attention convolution layer; and *No OGM* denotes removing the optimization graph module and using a predefined graph for GCN.

As shown in Table 2, the optimized GCN module can effectively improve the performance because it generates a graph adjacency matrix suitable for the current situation based on dynamic topology information and part of the original topology information. Accordingly, it can better describe the real-time changes in the road network. Modeling the time dependence can significantly improve the prediction performance because it can capture the temporal dependence between road segments. The proposed shared attention effectively captures the global dependence on road segments. The optimization graph module significantly affects the mean absolute error (MAE) and mean absolute percentage error (MAPE) because it can learn the dynamic graph while retaining the original road topology information. In summary, the shared attention collaboration with the optimization graph module can capture dynamic spatio-temporal correlations and obtain better performance.

Table 2. Ablation analysis on METR-LA and PEMS-BAY datasets

Model	METR / PEMS								
	15 min			30 min			60 min		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
No optim GCN	2.63/1.29	4.64/2.29	6.84/2.86	3.00/1.55	5.43/2.89	8.10/3.52	3.53/1.90	6.44/3.61	10.01/4.52
No TDC	2.61/1.32	4.57/2.37	7.39/3.07	2.94/1.56	5.28/2.92	8.54/3.68	3.43/1.89	6.22/3.58	10.28/4.59
No shared att	2.50/1.22	4.46/2.24	6.61/2.67	2.84/1.48	5.20/2.85	7.82/3.34	3.32/1.83	6.14/3.58	9.56/4.35
No OGM	2.51/1.22	4.45/2.23	6.73/2.70	2.84/1.47	5.17/2.81	7.91/3.34	3.31/1.80	6.08/3.49	9.62/4.28
SSTACON	2.49/1.20	4.45/2.21	6.55/2.58	2.83/1.46	5.19/2.81	7.75/3.24	3.30/1.80	6.11/3.50	9.49/4.22

The best values are shown in bold.
RMSE=root mean square error.

5. Conclusion

In this study, we proposed a novel traffic flow prediction method that uses weighted cosine similarity to calculate the similarity matrix between nodes and a double-gated aggregation mechanism to aggregate the proportion of the initial and currently learned graphs. In addition, we propose using shared attention to simulate the complex spatio-temporal correlation of traffic flow. Different convolution kernels are used in the time dilation convolution, which can capture the time correlation of different scales. Experimental evaluations conducted on two real-world datasets demonstrate that SSTACON outperforms state-of-the-art methods in both long- and short-term forecasting. In the future, dynamically adjusting the fusion ratio of the optimized graph and reducing the training time will be important.

Acknowledgement

This paper is supported by the Key Cooperation Project of Chongqing Municipal Education Commission (No. HZ2021008).

References

- [1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need" *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998-6008, 2017.
- [2] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 922-929, 2019. <https://doi.org/10.1609/aaai.v33i01.3301922>
- [3] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: data-driven traffic forecasting," in *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, 2018.
- [4] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, Macao, China, 2019, pp. 1907-1913.
- [5] C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: a graph multi-attention network for traffic prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 1, pp. 1234-1241, 2020. <https://doi.org/10.1609/aaai.v34i01.5477>
- [6] Y. Chen, L. Wu, and M. Zaki, "Iterative deep graph learning for graph neural networks: better and robust node embeddings," *Advances in Neural Information Processing Systems*, vol. 33, pp. 19314-19326, 2020.
- [7] 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*, Virtual Event, 2020, pp. 753-763.
- [8] B. N. Oreshkin, A. Amini, L. Coyle, and M. Coates, "FC-GAGA: fully connected gated graph architecture for spatio-temporal traffic forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 10, pp. 9233-9241, 2021. <https://doi.org/10.1609/aaai.v35i10.17114>
- [9] 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)*, 2022. <https://doi.org/10.1145/3532611>

- [10] J. Choi, H. Choi, J. Hwang, and N. Park, "Graph neural controlled differential equations for traffic forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 6, pp. 6367-6374, 2022. <https://doi.org/10.1609/aaai.v36i6.20587>
- [11] J. Ji, J. Wang, Z. Jiang, J. Jiang, and H. Zhang, "STDEN: towards physics-guided neural networks for traffic flow prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, pp. 4048-4056, 2022. <https://doi.org/10.1609/aaai.v36i4.20322>



Pengcheng Li <https://orcid.org/0000-0002-6911-3033>

He received B.S. degrees in Computer Science and Technology from Chongqing University of Posts and Telecommunications (CQUPT), Chongqing, China in 2020. He is currently pursuing an M.S. degree in the Department of Computer Science and Technology, CQUPT. His current research interests include traffic prediction.



Changjiu Ke <https://orcid.org/0000-0002-1066-1158>

He received his degree in Chongqing Technology and Business University (Yuzhou University), Chongqing, China, in 1997. He is currently an engineer at Chongqing Customs Technology Center. His research interests include calibration related research.



Hongyu Tu <https://orcid.org/0000-0003-3475-3388>

She received M.S. degrees in MBA from Chongqing University, Chongqing, China in 2009. She is currently a senior engineer at Chongqing Customs Technology Center. Her current research interests include risk prediction and related research.



Houbing Zhang <https://orcid.org/0000-0002-5312-7713>

He received his M.S. degree at Inner Mongolia University of Technology, Inner Mongolia, China, in 2008. He is currently a senior engineer at Chongqing Customs Technology Center. His current research interests include risk evaluation and prediction in related area.



Xu Zhang <https://orcid.org/0000-0002-7051-2736>

He received his Ph.D. in Computer and Information Engineering at Inha University, Incheon, Korea, in 2013. He is currently an associate professor at Chongqing University of Posts and Telecommunications. His research interests include urban computing, intelligent transportation system, and few-shot learning.