Killing two birds with one stone: A Spatio-Temporal Prompt for the Inductive Traffic Extrapolation

Leilei Ding[1], Zhipeng Tang[1], Le Zhang[2 (✉)], Dazhong Shen[3], Chao Wang[1], Ziyang Tao[1], Jingbo Zhou[2], Yanyong Zhang[1], and Hui Xiong[4 (✉)]

University of Science and Technology of China, Hefei, China
{dingleilei, tangzhipeng, ziyangtao}@mail.ustc.edu.cn
{wangchaoai,yanyongz}@ustc.edu.cn
Business Intelligence Lab of Baidu Research, Beijing, China
zhoujingbo@outlook.com, zhangle0202@gmail.com
Nanjing University of Aeronautics and Astronautics, Nanjing, China
dazh.shen@gmail.com
Hong Kong University of Science and Technology (Guangzhou), Guangdong, China
xionghui@ust.hk

Abstract. Smart cities rely on sensor systems to collect data that support urban management. However, the high cost and frequent malfunctions of traffic sensors in certain areas lead to sparse data, which limits the performance of downstream tasks. This paper tackles these limitations through inductive spatio-temporal extrapolation, which forecasts time-series data for locations without sensors by leveraging data from surrounding sensor-equipped areas. We introduce a novel **S**patio-**T**emporal **P**rompt (STP) framework to address two primary challenges: spatial uncertainty and temporal dynamic. Spatial uncertainty arises from the inherent unpredictability of unseen locations during inference, while temporal dynamic refers to the evolving and complex correlations among nodes over time. Our STP leverages self-supervised training with randomly selected prompt nodes to effectively handle spatial uncertainty. Additionally, we employ a temporal prompt pool to capture dynamic temporal relationships. Extensive experiments on three real-world datasets demonstrate that STP significantly outperforms existing state-of-the-art models, showcasing its effectiveness in dealing with sparse sensor data.

Keywords: Prompt Learning · Traffic Spatio-Temporal Extrapolation

1 Introduction

Smart cities are fundamentally built on the deployment of diverse sensors, which empower city managers with deeper insights into urban dynamics, enhancing decision-making, and improve both the efficiency of urban systems and residents' quality of life [9,20,19]. For example, traffic data plays a very important role in vehicle navigation, route planning, and congestion management [10,13,12]. However, while the fine-grained data gathered by these sensors offers significant advantages for downstream tasks, the high-cost deployment of numerous traffic sensors is often impractical, and existing sensors also frequently experience malfunctions. Consequently, many analytical tasks must contend with sparse data, leading to suboptimal outcomes. This situation highlights the urgent need for methods to approximate data for locations without sensors in sensor-sparse areas.

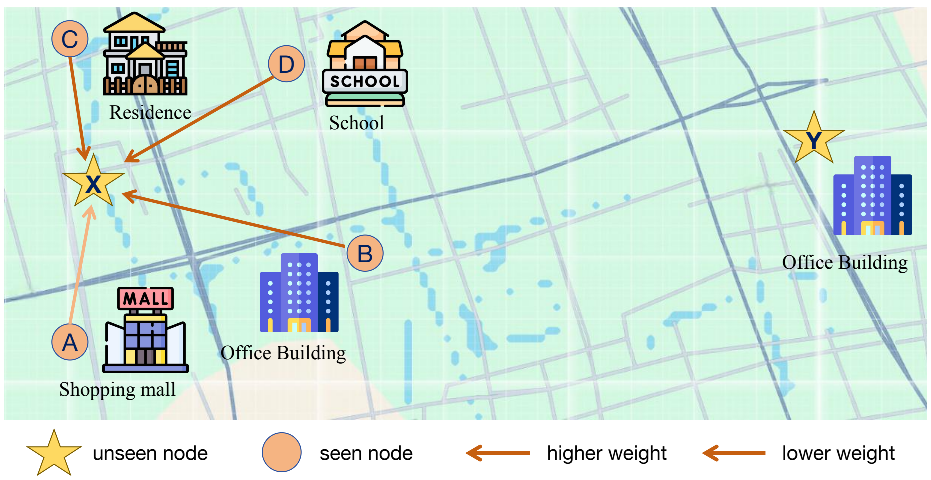

Fig. 1: An example of spatio-temporal extrapolation of traffic flow during peak commute hours on weekdays.

In this study, we investigated the fundamental and widely applied problem, namely inductive spatio-temporal traffic extrapolation, which forecasts time-series traffic data for locations without sensors by leveraging information from surrounding sensor-equipped areas. Specifically, we can only use the time-series signals of **seen nodes** (locations equipped sensors) and their spatial relationships, to determine the time-series signal for **unseen nodes** (locations unequipped sensors). To tackle inductive extrapolation, we must pay careful attention to the spatio-temporal correlations among sensors. Spatio-temporal graph neural networks [22,23,18,16] have strong capabilities to capture spatio-temporal correlations and are often regarded as state-of-the-art models for spatio-temporal tasks, including inductive extrapolation.

Although these spatio-temporal extrapolation methods have achieved significant success, they share a subtle yet crucial limitation that severely restricts their performance. During training, a portion of the seen nodes is randomly masked in each batch, and the model is tasked with inferring these masked nodes. As training progresses, every node in the seen graph participates, which can lead to over-fitting to this seen spatial graph structure. This over-fitting reduces the model's ability to extrapolate to unseen nodes. During inference, the model must extrapolate unseen nodes, whose spatial locations are entirely uncertain. We refer to this challenge as **spatial uncertainty** for unseen nodes. However, due to the limitations of the masking-based training approach, these models struggle to effectively handle spatial uncertainty, thereby limiting their overall performance. Especially in the field of traffic [14], the influence of the surrounding nodes on the unseen nodes varies at different time points. For example, as shown in Fig. 1, during peak commute hours on weekdays, the traffic flow at node $\mathbf{X}$ may be more similar to that at nodes $\mathbf{B}, \mathbf{C}, \mathbf{D}$. However, on holidays, its traffic flow might resemble that at Node $\mathbf{A}$ more closely. We refer to this intricate and dynamic temporal correlation as **temporal dynamic**.

It is challenging to adapt the model to address spatial uncertainty and capture temporal dynamic. Inspired by graph prompts [11,15,7] and temporal prompts [21,1], we introduce a novel **S**patio-**T**emporal **P**rompt (**STP**) for the first time to address spatial uncertainty and temporal dynamic by inserting prompt nodes with temporal-aware learnable embeddings [4,2,3].

To enhance the model's ability to generalize to unseen nodes, we randomly insert empty prompt nodes into the spatial graph during training and predict their extrapolations using a Spatio-Temporal Graph Neural Network (STGNN) [8]. Since these prompt nodes lack ground truth labels, we employ a self-supervised framework: predictions from seen nodes to prompt nodes are cyclically used to reconstruct the original seen nodes, preventing overfitting and improving adaptability to unseen graphs. Additionally, temporal dynamics are captured by querying a prompt pool[21] with temporal representations of surrounding nodes, retrieving top-K learnable vectors to represent prompt nodes. This ensures temporal consistency, as similar temporal states retrieve analogous prompt vectors, embedding dynamic correlations between node sequences into the model.

In summary, our main contributions are summarized as follows:

- We propose a **S**patio-**T**emporal **P**rompt for the Inductive Traffic Extrapolation task. To the best of our knowledge, this is the first work that brings a spatio-temporal prompt to the spatio-temporal extrapolation.
- We have introduced a novel insight into existing spatio-temporal extrapolation methods, discovering the spatial uncertainty and temporal dynamic in this task, which were often overlooked by the previous methods. By addressing these issues, we have achieved significant performance improvements.
- We compared four different state-of-the-art models across three traffic datasets, and the experimental results indicate that the STP-STGNN outperforms state-of-the-art models, thereby validating the effectiveness of our approach.

2 Problem Formulation

Definition 1 (Spatial data). The spatial structure between the seen nodes is intuitively represented as a graph $\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)$, where $\mathcal{V}_s$ is the set of seen nodes and $\mathcal{E}_o$ is the set of edges between the seen nodes. Similarly, the spatial structure of the unseen nodes can be defined as $\mathcal{G}_u = (\mathcal{V}_u, \mathcal{E}_u)$, and the prompt graph is denoted as $\mathcal{G}_p = (\mathcal{V}_p, \mathcal{E}_p)$. Additionally, the spatial structure graph encompassing both the seen and unseen nodes is represented as $\mathcal{G}_{su} = (\mathcal{V}_{su}, \mathcal{E}_{su})$, while the graph comprising the seen and prompt nodes is denoted as $\mathcal{G}_{sp} = (\mathcal{V}_{sp}, \mathcal{E}_{sp})$.

Definition 2 (Temporal Data). $x_{i,t} \in \mathbb{R}^C$ represents C types of data recorded at the time step t for the node i. Given the P time steps as window, $\mathrm{X}_i = (\mathrm{x}_{i,1}, \mathrm{x}_{i,2}, \cdots, \mathrm{x}_{i,P})^\top \in \mathbb{R}^{P \times C}$ is a time series at the node i, while the temporal data of all N nodes is donates as $\mathcal{X} = (\mathrm{X}_1, \mathrm{X}_2, \cdots, \mathrm{X}_N)^\top \in \mathbb{R}^{N \times P \times C}$. Specifically, we represent the temporal data for the seen, unseen, and prompt nodes as $\mathcal{X}_s$ $\mathcal{X}_u$ and $\mathcal{X}_p$, respectively.

Definition 3 (Inductive extrapolation). The predicted extrapolations of N_u unseen nodes are denoted as $\hat{\mathcal{Y}} = \left(\hat{\mathrm{Y}}_1, \hat{\mathrm{Y}}_2, \cdots, \hat{\mathrm{Y}_{N_u}}\right)^\top \in \mathbb{R}^{N_u \times P \times C}$. Inductive extrapolation utilizes $\mathcal{G}_s$ and $\mathcal{X}_s$ to train the model. After training, the model use $\mathcal{G}_{su}$, $\mathcal{X}_s$ to predict the extrapolations of unseen nodes.

3 Method

In this section, we will provide further details about the spatio-temporal Graph Neural Network (STGNN) enhanced with Spatio-Temporal Prompt (STP).

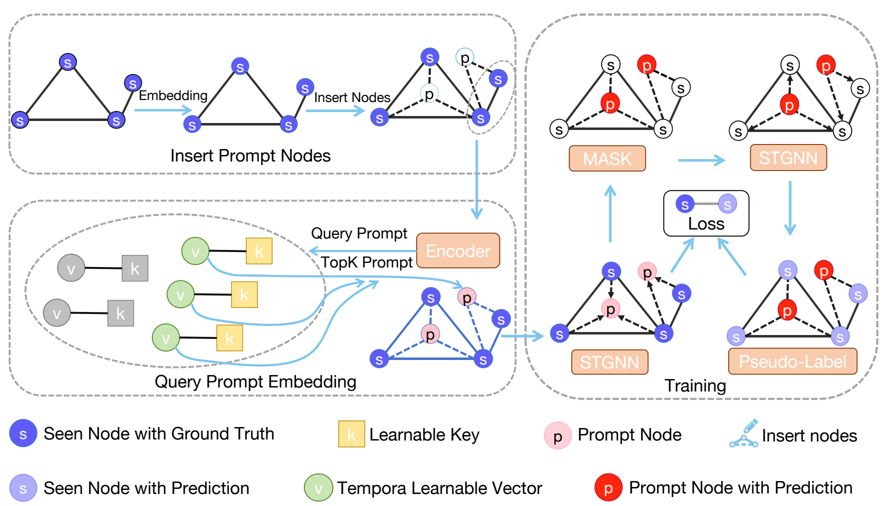

Fig. 2: Overview of STP. STP randomly inserts prompt nodes into the seen graph and retrieves embeddings from neighboring nodes. It then selects the top-K temporal embeddings from a pool to represent each prompt node's temporal features. In semi-supervised training, the seen nodes' temporal embeddings predict the prompt nodes' extrapolated values, and the masked temporal values of the seen nodes are inferred from these predictions and compared with the ground truth.

3.1 Model Overview

In this section, we will provide an overview of STP, including the form of prompts, the insertion of prompt nodes, querying the embeddings of prompt nodes, and finally conducting unsupervised training. The framework diagram is shown in Fig. 2.

3.2 Insert Prompt Nodes

In each training batch, the random insertion enables the model to adapt to different spatial structures rather than simply converging to the seen graph. Additionally, to increase the randomness of the connections for the inserted nodes, we apply random pruning to the connections between the prompt nodes and the seen nodes.

3.3 Query Prompt Embedding

Inspired by [21], we use temporal prompts to represent temporal dynamic. Intuitively, if the temporal representation embeddings of a prompt node's neighboring nodes are similar, then these neighboring nodes have similar temporal relevance to the prompt node. Therefore, we retrieve the corresponding temporal prompts from the prompt pool based on the temporal representation embeddings of the prompt node's neighboring nodes, and use them as the temporal representation for the spatio-temporal prompt nodes. This approach ensures that prompt nodes with similar neighboring nodes have similar temporal representations, signaling to the model the temporal correlation between them.

Temporal Prompt Pool. The temporal prompt pool consists of L key-value pairs, where the key is a learnable vector used for similarity retrieval, and the value is a learnable temporal prompt embedding designed to indicate temporal correlations. The temporal prompt pool can flexibly update its parameters (both keys and values) during the training process. The prompt pool is denoted as $(\mathbf{Key}s, \mathbf{Val}s)$, consisting of a set of learnable keys $\boldsymbol{k}$ and temporal prompts $\boldsymbol{P}$.

$$(\mathbf{Key}s, \mathbf{Val}s) = \{(\boldsymbol{k}_i, \boldsymbol{P}_i)\}_{i=1}^{L}, \tag{1}$$

Query Temporal Prompt. After inserting the prompt nodes, we can obtain all the first-order neighbor seen nodes of prompt node i. We apply a pooling function (max pooling, average pooling, or max-average pooling) to the embeddings of the first-order neighbors of node i, thus obtaining the embeddings $\boldsymbol{h}_i$ for prompt node i as follows.

$$\boldsymbol{h}_i = Pooling(\boldsymbol{h}_{neighbors\_of\_i}) \tag{2}$$

We use the embeddings $\boldsymbol{h}_i$ of the prompt node i to retrieve the top$\mathbf{K}$ most similar keys from the temporal prompt pool via Maximum Inner Product Search (MIPS). The top$\mathbf{K}$ selected keys of prompt node i can be denoted as:

$$\mathbf{K}_{\text{topk-i}} = \underset{\{j\}_{j=1}^{N}}{\text{argmin}} MIPS \sum_{j=1}^{K} (\boldsymbol{h}_i, \boldsymbol{k}_j), \tag{3}$$

To allow the learnable keys vector to update its parameters during training, we define the $\mathcal{L}_{keys}$ as:

$$\mathcal{L}_{keys} = \beta \sum_{i} \sum_{\text{K}_{\text{topk-i}}} MIPS (\boldsymbol{h}_i, \boldsymbol{k}_j) \tag{4}$$

3.4 Semi-supervised Learning

To address spatial uncertainty, we aim for the model to accurately predict the extrapolation values of the randomly inserted prompt nodes. However, these random prompt nodes do not have ground truth. To resolve this issue, we adopt a semi-supervised learning approach. Intuitively, if the model provides more accurate predictions for the seen nodes, it will also exhibit a stronger capability to capture the spatio-temporal relationships surrounding the randomly inserted prompt nodes.

Specifically, as shown in Fig. 2, the model first temporal predicts the values for the prompt nodes, then masks the temporal values of the seen nodes. Subsequently, the model uses the predicted prompt node values to infer the temporal values of the seen nodes, and the loss function is computed based on the difference between the predicted and ground truth values of the seen nodes.

The overall loss function can be defined in two parts: the MAE loss calculated from the predicted values and the ground truth, and the optimization loss of the learnable keys vectors. This can be expressed as:

$$\mathcal{L} = \mathcal{L}_{MAE} + \beta \mathcal{L}_{keys} \tag{5}$$

4 Experiments

In this section, we present the experimental setting, the overall performance, and an ablation study.

4.1 Experimental Settings

Datasets. We conduct experiments using three popular real-world traffic spatio-temporal datasets. **METR-LA** consists of traffic speed data collected from 207 sensors across Los Angeles County over a four-month period (from March 1, 2012, to June 30, 2012). **PEMS-BAY** is a traffic dataset from 325 sensors in the Bay Area, San Francisco, recorded every 5 minutes from January 1 to June 30, 2016. **SEA-LOOP** contains traffic data collected from 323 inductive loop detectors on Seattle freeways.

Baslines. We select the top 4 state-of-the-art and most popular neural networks baselines for comparison. IGNNK[17], INCREASE[24], LSJSTN[6] and STGNP[5] are described in Section 4.2.

Evaluation. We evaluate the model performance using three widely-used metrics, Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RSME). Additionally, our dataset processing follows the procedure in GRIN, where the data is split into training, validation, and test sets with a ratio of 7:1:2. Additionally, we set the learning rate to 0.0002. Following popular experimental setups, we set the missing rate to 50%.

4.2 Overall Performance

Model	METR-LA207			PEMS BAY325			SEA_LOOP323		
	MAE	MAPE	RSME	MAE	MAPE	RSME	MAE	MAPE	RSME
IGNNK	6.966	0.208	10.405	5.602	0.144	9.705	<u>4.542</u>	0.150	7.244
INCREASE	<u>6.243</u>	<u>0.184</u>	**9.831**	3.841	<u>0.088</u>	<u>6.606</u>	5.973	0.210	9.585
LSJSTN	7.151	0.201	10.624	3.890	0.089	6.737	4.691	<u>0.129</u>	<u>7.185</u>
STGNP	7.591	0.188	13.146	<u>3.812</u>	0.091	6.932	4.738	0.157	7.530
STP	**6.126**	**0.179**	<u>9.923</u>	**3.676**	**0.085**	**6.556**	**4.230**	**0.123**	**6.934**
Improv.	+1.9%	+2.7%	−1.0%	+3.6%	+3.4%	+0.8%	+6.9%	+2.3%	+3.5%

Table 1: Performance comparison of our STP on the three traffic datasets. The numbers following the dataset represent the total number of nodes in the dataset. Values in bold denote the best results. Values under lined denote the state-of-the-ar results. The improvement represents the enhancement of STP compared to the current best model.

Overall performance comparison. As shown in Table 1, our proposed STP model consistently outperforms all other models across three of the most popular traffic datasets. IGNNK, the first to apply graph spatio-temporal neural networks for inductive extrapolation tasks, offers robustness and stable performance due to its simple design. However, a notable performance gap exists between IGNNK and STP in most cases. INCREASE leverages a multi-relational attention representation learning mechanism, but the lack of point of interest (POI) attribute information in our datasets limits its performance. LSJSTN's complex design reduces its robustness, resulting in significant performance fluctuations across different datasets. STGNP, the latest extrapolation algorithm based on neural processes, is both simple and interpretable; however, its limited neural network fitting capability causes it to underperform relative to STP. STP achieves the highest MAE improvement on the SEA-LOOP dataset, with an improvement of +6.9%, and the smallest on the METR-LA207 dataset, with an improvement of 1.9%. The extensive performance improvements of STP can be largely attributed to its improved capacity to manage spatial uncertainty and temporal dynamic, which aligns closely with the requirements of inductive extrapolation tasks.

4.3 Ablation Study.

To validate the effectiveness of the spatio-temporal prompt, we perform an ablation study on the STP model. As shown in Table 2, the spatio-temporal prompt plays a decisive role in enhancing the model's performance. These results are sufficient to demonstrate the effectiveness of our approach.

Ablation Experiment	MAE	MAPE	RMSE
STGNN with STP	16.502	0.394	31.002
STGNN without STP	17.391	0.454	31.610

Table 2: Ablation Study Results

5 Conclusion

This paper addresses the challenge of inductive spatio-temporal extrapolation in smart cities. We introduce a novel Spatio-Temporal Prompt (STP) to handle two key challenges: spatial uncertainty and temporal dynamic. The STP approach uses self-supervised training with random prompt nodes and a temporal prompt pool, significantly improving the model's ability to predict values for unseen nodes and adapt to dynamic temporal correlations. Experimental results across multiple datasets demonstrate that STP outperforms existing state-of-the-art models.

6 Acknowledgements

This work was supported in part by the National Key R&D Program of China (Grant No. 2023YFF0725001), the National Natural Science Foundation of China (Grant No. 92370204, No.92370204), the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2023B1515120057), Nansha Postdoctoral Research Project, Guangzhou-HKUST (GZ) Joint Funding Program (Grant No. 2023A03J0008), and the Education Bureau of Guangzhou Municipality.

References

1. Cao, D., Jia, F., Arik, S.O., Pfister, T., Zheng, Y., Ye, W., Liu, Y.: Tempo: Prompt-based generative pre-trained transformer for time series forecasting. arXiv preprint arXiv:2310.04948 (2023)
2. Chen, L., Qin, C., Sun, Y., Song, X., Xu, T., Zhu, H., Xiong, H.: Collaboration-aware hybrid learning for knowledge development prediction. In: Proceedings of the ACM on Web Conference 2024. pp. 3976–3985 (2024)
3. Chen, L., Sun, Y., Zhang, S., Ye, Y., Wu, W., Xiong, H.: Tackling uncertain correspondences for multi-modal entity alignment. In: Proceedings of the 38th Conference on Neural Information Processing Systems (2024)
4. Chen, L., Tong, P., Jin, Z., Sun, Y., Ye, J., Xiong, H.: Plan-on-graph: Self-correcting adaptive planning of large language model on knowledge graphs. In: Proceedings of the 38th Conference on Neural Information Processing Systems (2024)
5. Hu, J., Liang, Y., Fan, Z., Chen, H., Zheng, Y., Zimmermann, R.: Graph neural processes for spatio-temporal extrapolation. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 752–763 (2023)
6. Hu, J., Liang, Y., Fan, Z., Liu, L., Yin, Y., Zimmermann, R.: Decoupling long- and short-term patterns in spatiotemporal inference. IEEE Transactions on Neural Networks and Learning Systems (2023)
7. Jiang, W., Wu, W., Zhang, L., Yuan, Z., Xiang, J., Zhou, J., Xiong, H.: Killing two birds with one stone: Cross-modal reinforced prompting for graph and language tasks. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 1301–1312 (2024)
8. Jin, G., Liang, Y., Fang, Y., Shao, Z., Huang, J., Zhang, J., Zheng, Y.: Spatio-temporal graph neural networks for predictive learning in urban computing: A survey. IEEE Transactions on Knowledge and Data Engineering (2023)
9. Li, S., Zhou, J., Liu, J., Xu, T., Chen, E., Xiong, H.: Multi-temporal relationship inference in urban areas. In: Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining. pp. 1316–1327 (2023)
10. Li, S., Zhou, J., Xu, T., Liu, H., Lu, X., Xiong, H.: Competitive analysis for points of interest. In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 1265–1274 (2020)
11. Sun, M., Zhou, K., He, X., Wang, Y., Wang, X.: Gppt: Graph pre-training and prompt tuning to generalize graph neural networks. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 1717–1727 (2022)
12. Sun, Q., Zha, R., Zhang, L., Zhou, J., Mei, Y., Li, Z., Xiong, H.: Crosslight: Offline-to-online reinforcement learning for cross-city traffic signal control. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 2765–2774 (2024)
13. Sun, Q., Zhang, L., Yu, H., Zhang, W., Mei, Y., Xiong, H.: Hierarchical reinforcement learning for dynamic autonomous vehicle navigation at intelligent intersections. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 4852–4861 (2023)
14. Sun, Q., Zhang, L., Zhou, J., Zha, R., Mei, Y., Tian, C., Xiong, H.: Spatio-temporal sequence modeling for traffic signal control. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. pp. 4076–4080 (2024)

15. Sun, X., Cheng, H., Li, J., Liu, B., Guan, J.: All in one: Multi-task prompting for graph neural networks. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 2120–2131 (2023)
16. Wang, T., Fan, Q., Wang, C., Yang, L., Ding, L., Yuan, N.J., Xiong, H.: Flagvne: A flexible and generalizable reinforcement learning framework for network resource allocation. arXiv preprint arXiv:2404.12633 (2024)
17. Wu, Y., Zhuang, D., Labbe, A., Sun, L.: Inductive graph neural networks for spatiotemporal kriging. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 4478–4485 (2021)
18. Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., Zhang, C.: 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. pp. 753–763 (2020)
19. Xiao, C., Zhou, J., Huang, J., Zhu, H., Xu, T., Dou, D., Xiong, H.: A contextual master-slave framework on urban region graph for urban village detection. In: 2023 IEEE 39th International Conference on Data Engineering (ICDE). pp. 736–748. IEEE (2023)
20. Xiao, C., Zhou, J., Huang, J., Zhuo, A., Liu, J., Xiong, H., Dou, D.: C-watcher: A framework for early detection of high-risk neighborhoods ahead of covid-19 outbreak. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 4892–4900 (2021)
21. Xue, S., Wang, Y., Chu, Z., Shi, X., Jiang, C., Hao, H., Jiang, G., Feng, X., Zhang, J., Zhou, J.: Prompt-augmented temporal point process for streaming event sequence. Advances in Neural Information Processing Systems **36**, 18885–18905 (2023)
22. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv preprint arXiv:1709.04875 (2017)
23. Zheng, C., Fan, X., Wang, C., Qi, J.: Gman: A graph multi-attention network for traffic prediction. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 1234–1241 (2020)
24. Zheng, C., Fan, X., Wang, C., Qi, J., Chen, C., Chen, L.: Increase: Inductive graph representation learning for spatio-temporal kriging. In: Proceedings of the ACM Web Conference 2023. pp. 673–683 (2023)