An Urban Traffic Knowledge Graph-Driven Spatial-Temporal Graph Convolutional Network for Traffic Flow Prediction

Chengbiao Yang
School of Computer Science and Engineering,
Southeast University
Nanjing, China
ycb_seu@hotmail.com

Guilin Qi
School of Computer Science and Engineering,
Southeast University
Nanjing, China
gqi@seu.edu.cn

ABSTRACT

Traffic flow prediction is a critical issue for researchers and practitioners in the field of transportation. Due to the high nonlinearity and complexity of traffic data, deep learning approaches have attracted much interest in recent years. However, existing studies seldom consider the topology of these urban roads and the connectivity of the monitor sensors. As we know, the real cause of the spread of traffic congestion is the connectivity of these road segments, rather than their spatial proximity. But it is challenging to model the topology of the urban traffic networks for traffic flow prediction. In this short research paper, we present an urban traffic knowledge graph-driven spatial-temporal graph convolutional networks for traffic flow prediction. We first construct an urban traffic knowledge graph that can represent the physical connectivity between roads and monitor sensors. Then, we use the urban traffic knowledge graph to improve the traffic flow networks. Finally, we combine the knowledge graph and traffic flow as the input of a spatial-temporal graph convolutional backbone networks. Experiments on two real-world traffic datasets verify the effectiveness of our approach.

KEYWORDS

traffic flow prediction, urban traffic knowledge graph, spatial-temporal graph convolutional networks, knowledge graph representation, topology of roads

1 Introduction

Traffic flow prediction has gained more and more attention with the rapid development of intelligent transportation systems (ITSs) [1]. Traffic flow prediction is an indispensable part of ITS, especially on the urban roads which has large traffic flows and frequently changing vehicle speed. As the great social and economic value, traffic flow prediction has become a popular research topic. The traditional methods often adopt statistical models (e.g. VAR [2]) and machine learning models (e.g. SVM [3]). The statistical models usually perform poorly in practice, and the machine learning models need careful feature engineering. With the development of deep learning, these traditional methods are rapidly eliminated.

Deep learning approaches have been widely and successfully applied to various traffic tasks nowadays. Significant progress has been made in the research of traffic flow prediction. Some researchers have used RNNs, such as LSTM [4, 5] and GRU [6] to model the temporal dependence of traffic flows. Some researchers have used CNNs [7] to model the spatial dependence of traffic flows. And some have used a combination of the two [8].

As traffic data are recorded via monitor sensors at fixed points in time and at fixed locations distributed in continuous space, urban traffic flow prediction is a typical problem of spatial-temporal data forecasting. In recent years, GNNs have emerged to better model the spatial and temporal dependences of roads and improve the prediction accuracy. Such models include the T-GCN [9], STGCN [11], DCRNN [15], ASTGCN [12], LSGCN [13], SeqGNN [14] and KST-GCN [10], among others. SeqGNN shows that the connectivity of roads is very important, rather than their spatial proximity. And KST-GCN shows that the knowledge of traffic can be helpful for the traffic prediction, but the external knowledge is difficult to obtain and the effect is not significant.

Our research shows that, it is very practical to make full use of the internal knowledge of the traffic networks, such as the topology of these urban roads and the connectivity of the monitor sensors. As we know, the real cause of the spread of traffic congestion is the connectivity of these roads within the road network topology. However, existing studies rarely consider this factor. Therefore, how to design an effective method to model the topology knowledge of the urban traffic networks is the main challenge. In this short research paper, the main contributions of this paper are summarized as follows:

1. We construct an urban traffic knowledge graph that can represent the physical connectivity between roads and monitor sensors. And use it to improve the traffic flow networks.

2. We provide a feasible network framework to model the topology knowledge of the urban traffic networks.
3. We evaluate our method on two real-world traffic datasets and the experimental results show the effectiveness of this method.

The rest of this paper is organized as follows: Section 2 will give a brief analysis of the traffic flow networks and the traffic flow prediction problem; Section 3 will introduce the urban traffic knowledge graph-driven spatial-temporal graph convolutional networks designed in this paper; Section 4 evaluates the performance of our model in terms of traffic flow prediction and presents the experimental results; Section 5 concludes the paper with some ideas for future research.

2 Preliminary

2.1 Traffic Flow Networks

In this study, we define a traffic flow network as an undirected graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{A}, \boldsymbol{X})$, where $\boldsymbol{V}$ is the set of nodes, indicating the monitor sensors; $\boldsymbol{E}$ is a set of edges, indicating the connectivity between the nodes; $\boldsymbol{A}$ denotes the adjacency matrix of graph $\boldsymbol{G}$; $\boldsymbol{X}$ is the history traffic flow data of $\boldsymbol{V}$.

In practice, a node may represent a monitor sensor located at the corresponding road of the traffic networks, as shown in Figure 1(a). Each node detects traffic flow with the same sampling frequency, that is, the features of nodes change at each time slice, as shown in Figure 1(b). Most of the existing studies use straight-line distance stands for the connectivity between two nodes, ignoring the importance of the road network connectivity. We will introduce the construction method of $\boldsymbol{E}$ in detail in the section 3.2.

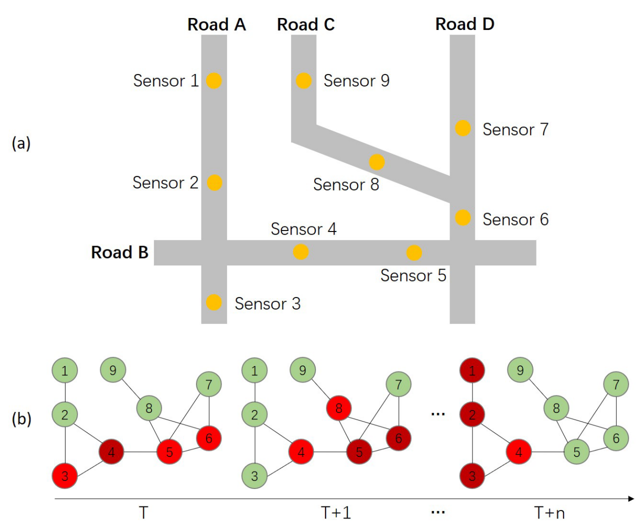

Figure 1: (a) A road network with 9 monitor sensors. (b) The traffic flow networks over time form *T* to *T+n*, where the nodes stand for the monitor sensors. The color of the nodes represents the traffic flow detected by the monitor sensors, which red means heavy traffic and green means little traffic.

2.2 Traffic Flow Prediction

For a traffic flow network $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{A}, \boldsymbol{X})$, let $\boldsymbol{x}_t^i$ represents the traffic flow value of the node $\boldsymbol{i}$ at time step $\boldsymbol{t}$, and $\boldsymbol{x}_t$ represents the traffic flow values of $\boldsymbol{V}$ at time step $\boldsymbol{t}$. Given history traffic flow data $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k)^{\boldsymbol{T}}$, and the urban traffic knowledge graph $\boldsymbol{KG}$, the purpose of traffic flow prediction is to predict the traffic flow values of all nodes in future time steps, namely $\boldsymbol{Y}$. The problem can be considered as learning the function $\boldsymbol{f}$, which denoted in Equation 1. For more details on urban traffic knowledge graph $\boldsymbol{KG}$, refer to section3.1.

$$Y = f(G, KG) \tag{1}$$

3 Proposed Method

3.1 Urban Traffic Knowledge Graph

We first define a simple ontology of the urban traffic knowledge graph, as shown in Figure 2. We define 2 classes and 4 relationships. The 2 classes are easy to understand, but the relationships need to be explained.

The relationship ***intersect*** means that two roads are directly connected or intersected. The relationship ***located*** means that a monitor sensor is located on a road. The relationship ***next_to*** means two sensors are located on the same road and next to each other. The relationship ***near*** means two sensors are located on different roads and next to each other. These four relationships can subtly represent the spatial knowledge of the urban road networks.

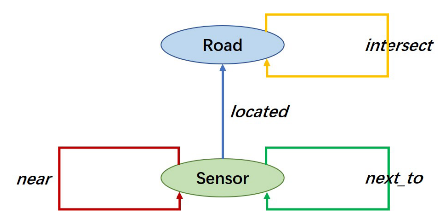

Figure 2: The ontology of urban traffic knowledge graph.

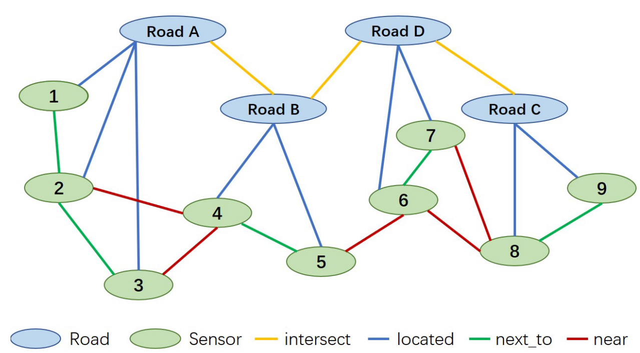

Figure 3: The urban traffic knowledge graph of the road networks in Figure 1(a).

Based on the ontology, we can extract the urban traffic knowledge graph at a low cost. First, we extract all road and sensor entities, in which the road entity is identified by name and the sensor entity is identified by number. Secondly, we get the longitudes and latitudes of the roads from the Open Government Data, and then get the geographic information about the sensors from the datasets. Thirdly, we construct the relationships ***intersect*** and ***located*** by calculating longitudes and latitudes. Fourthly, we construct the relationship ***next_to*** by calculating the adjacency of sensors road by road. Fifthly, we construct the relationship ***near*** by searching the sensors near the corners. A constructed knowledge graph is shown in Figure 3.

3.2 Knowledge Based Traffic Flow Networks

For the traffic flow network $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{A}, \boldsymbol{X})$, we should build $\boldsymbol{E}$ first, and then we can get $\boldsymbol{A}$ with simple processing. As the edges indicate the connectivity between the nodes, we can use the distance between nodes to judge whether they should have edges. And the best way is to calculate the routing distance, rather than straight-line distance, as shown in Figure 4(a). But the cost of calculating the routing distance between all nodes is very high, so we use the urban traffic knowledge graph to increase efficiency, the steps are as follow:

1. Select all the relationships of type ***next_to*** and ***near*** from the urban traffic knowledge graph.
2. Calculate the routing distance of the selected relationships.
3. If the routing distance is less than the threshold, add the corresponding edge to the traffic flow network.

Through this method, we filter out the edge between sensor 7 and sensor 8, as shown in Figure 4(b), and get the final traffic flow network, as shown in Figure 4(c).

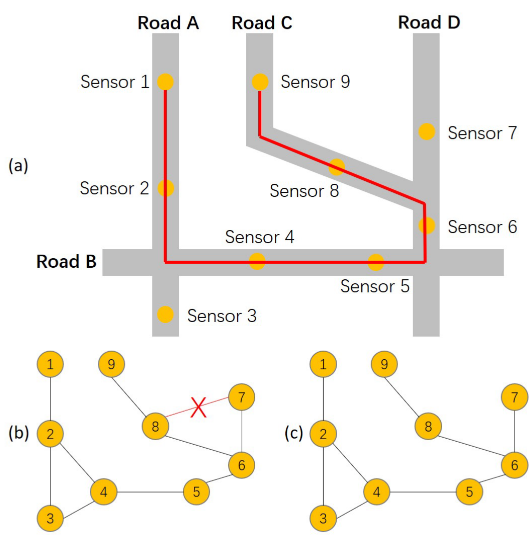

Figure 4: (a) The route distance between sensor 1 and sensor 9 is much greater than the straight-line distance. (b) Calculate the edges that can be removed. (c) The traffic flow network of the road networks in Figure 1(a).

3.3 Knowledge graph-driven spatial-temporal graph convolutional networks

Figure 5 presents the overall framework of the knowledge graph-driven spatial-temporal graph convolutional networks. We provide a feasible solution for modeling the topology knowledge of the urban traffic networks.

To begin with, we derive the embedding of the entities with a knowledge graph representation learning model, here we use TransD [16]. Then, we get the history traffic flow data $\boldsymbol{X}$ and adjacency matrix $\boldsymbol{A}$ from the traffic flow network $\boldsymbol{G}$. And then, we combine the knowledge graph embeddings and traffic flow as the input of a graph convolutional networks. Furthermore, we use the recurrent neural networks to capture the temporal dependency. Finally, we use a fully connected layer to generate the predicted data.

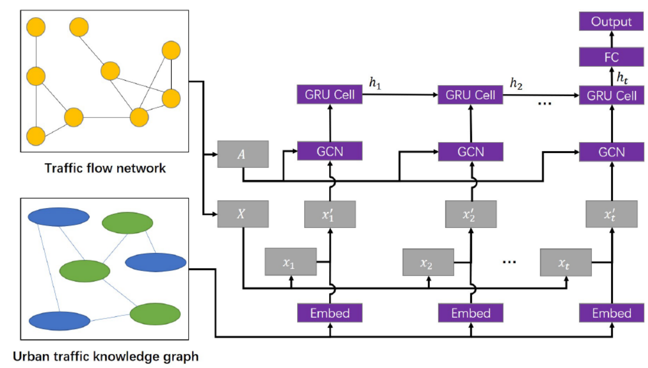

Figure 5: The network Architecture. FC: Full connected layers.

We use some special treatments to combine the knowledge embeddings $\boldsymbol{X_{KG}}$, and traffic flow features $\boldsymbol{X_t}$ observed at time $\boldsymbol{t}$, as shown in Figure 6. The output is the updated traffic flow features fused with topology knowledge of the urban traffic networks at time $\boldsymbol{t}$, which we denoted as $\boldsymbol{X'_t}$, as shown in Equation 2. And $\boldsymbol{w}$ is linear transformations, $\boldsymbol{b}$ is bias constants.

$$X'_t = Relu(x_t w + b) \tag{2}$$

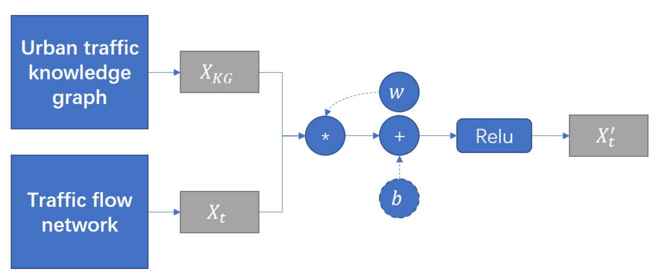

Figure 6: The network architecture for combining the knowledge graph embeddings and traffic flow as the input of a graph convolutional networks.

The inputs of the GCN are the updated road section features $\boldsymbol{X}_t'$ and the adjacency matrix $\boldsymbol{A}$. The GCN uses graph spectral theory to capture the topological relations and features of the traffic network and obtain the representation vector of each sensor. Here we use STGCN [11] and ASTGCN [12] as the backbone for the experiments.

We use the recurrent neural networks to capture the temporal dependency. State $\boldsymbol{h}_t$ represents the output at time $\boldsymbol{t}$. Finally, $\boldsymbol{h}_t$ is fed into a fully connected layer to generate the predicted future traffic flow $\widehat{\boldsymbol{Y}}$. The objective of the training process is to minimize the error between the predicted traffic flow $\widehat{\boldsymbol{Y}}$ and real traffic flow $\boldsymbol{Y}$. So the loss function is formulated as:

$$loss = \left\|\widehat{Y} - Y\right\| + \delta L_{reg} \tag{3}$$

4 Experiment

4.1 Dataset Description

We validate our model on two highway traffic datasets PeMSD4 and PeMSD8 from California. The datasets are collected by the Caltrans Performance Measurement System [17] in real-time every 30 seconds. The traffic data are aggregated into every 5-minute interval from the raw data. The system has more than 39,000 detectors deployed on the highway in the major metropolitan areas in California. Geographic information about the sensor stations is recorded in the datasets. And we get the map date of California's roads from the California Open Data[1] website.

PeMSD4. It refers to the traffic data in San Francisco Bay Area, containing 3848 sensors on 29 roads. The time span of this dataset is from January to February in 2018. We choose data on the first 50 days as the training set, and the remains as the test set.

PeMSD8. It is the traffic data in San Bernardino from July to August in 2016, which contains 1979 sensors on 8 roads. The data on the first 50 days are used as the training set and the data on the last 12 days are the test set.

4.2 Data Preprocessing

We remove some redundant sensors to ensure every sensor is on the road networks. Finally, there are 3812 sensors in the PeMSD4 and 1955 sensors in the PeMSD8. The traffic data is aggregated every 5 minutes, so each sensor contains 288 data of points per day. And the missing values are filled by linear interpolation.

We build the urban traffic knowledge graphs and traffic flow networks for the datasets PeMSD4 and PeMSD8. The knowledge graph of PeMSD4 has 3841 entities and 8065 relationships. The knowledge graph of PeMSD8 has 1963 entities and 4318 relationships. The traffic flow network of PeMSD4 has 3812 nodes and 4213 edges. The traffic flow network of PeMSD8 has 1955 nodes and 2311 edges.

[1] https://data.ca.gov/

4.3 Experiment Results

We evaluate the proposed method on two representative spatial-temporal graph convolutional networks, STGCN [11] and ASTGCN [12], and name their corresponding models of our approach as KG-STGCN and KG-ASTGCN, respectively. We compare KG-STGCN, KG-ASTGCN, STGCN and ASTGCN on PeMSD4 and PeMSD8. Table 1 shows the average results of traffic flow prediction performance over the next one hour. It can be seen from Table 1 that our KG-STGCN and KG-ASTGCN achieve the better performance than STGCN and ASTGCN in both two datasets in terms of all evaluation metrics.

Table1: Average performance comparison of different approaches on PeMSD4 and PeMSD8.

Model	PeMSD4		PeMSD8	
	RMSE	MAE	RMSE	MAE
STGCN	33.16	22.83	24.78	15.32
ASTGCN	34.71	23.26	26.86	17.12
KG-STGCN	**32.87**	**22.65**	**24.23**	**14.89**
KG-ASTGCN	**34.56**	**23.02**	**25.74**	**16.25**

5 Conclusion

In this paper, we present an urban traffic knowledge graph-driven spatial-temporal graph convolutional networks for traffic flow prediction. Our method tries to make full use of the internal knowledge of the traffic networks, such as the topology of these urban roads and the connectivity of the monitor sensors. Firstly, we construct the urban traffic knowledge graph, which can represent the topology and connectivity of the roads and monitor sensors. Then we can derive the embeddings of the nodes with a knowledge graph representation learning model. Secondly, we improve the traffic flow networks with the connectivity of the roads. Finally, we combine the knowledge graph embeddings and traffic flow as the input of a graph convolutional networks. This study provides a feasible solution for modeling the topology knowledge of the urban traffic networks. However, the urban traffic flow is affected by many external factors, like weather and social events. In the future, we will take some external influencing factors into account to further improve the forecasting accuracy.

REFERENCES

[1] Zhang, J., Wang, F. Y., Wang, K., Lin, W. H., Xu, X., & Chen, C. (2011). Data-driven intelligent transportation systems: A survey. IEEE Transactions on Intelligent Transportation Systems, 12(4), 1624-1639.
[2] Zivot, E., & Wang, J. (2006). Vector autoregressive models for multivariate time series. Modeling financial time series with S-PLUS®, 385-429.
[3] Jeong, Y. S., Byon, Y. J., Castro-Neto, M. M., & Easa, S. M. (2013). Supervised weighting-online learning algorithm for short-term traffic flow prediction. IEEE Transactions on Intelligent Transportation Systems, 14(4), 1700-1707.
[4] Ma, X., Tao, Z., Wang, Y., Yu, H., & Wang, Y. (2015). Long short-term memory neural network for traffic speed prediction using remote microwave

sensor data. Transportation Research Part C: Emerging Technologies, 54, 187-197.

[5] Tian, Y., & Pan, L. (2015, December). Predicting short-term traffic flow by long short-term memory recurrent neural network. In 2015 IEEE international conference on smart city/SocialCom/SustainCom (SmartCity) (pp. 153-158). IEEE.

[6] Fu, R., Zhang, Z., & Li, L. (2016, November). Using LSTM and GRU neural network methods for traffic flow prediction. In 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC) (pp. 324-328). IEEE.

[7] Chen, M., Yu, X., & Liu, Y. (2018). PCNN: Deep convolutional networks for short-term traffic congestion prediction. IEEE Transactions on Intelligent Transportation Systems, 19(11), 3550-3559.

[8] Yu, H., Wu, Z., Wang, S., Wang, Y., & Ma, X. (2017). Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. Sensors, 17(7), 1501.

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

[10] Zhu, J., Han, X., Deng, H., Tao, C., Zhao, L., Wang, P., ... & Li, H. (2022). KST-GCN: A knowledge-driven spatial-temporal graph convolutional network for traffic forecasting. IEEE Transactions on Intelligent Transportation Systems.

[11] Yu, B., Yin, H., & Zhu, Z. (2017). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. arXiv preprint arXiv:1709.04875. Date:June

[12] Guo, S., Lin, Y., Feng, N., Song, C., & Wan, H. (2019, July). Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 922-929). Copyright Year:2018

[13] Huang, R., Huang, C., Liu, Y., Dai, G., & Kong, W. (2020, July). LSGCN: Long Short-Term Traffic Prediction with Graph Convolutional Networks. In IJCAI (pp. 2355-2361).

[14] Xie, Z., Lv, W., Huang, S., Lu, Z., Du, B., & Huang, R. (2019). Sequential graph neural network for urban road traffic speed prediction. IEEE Access, 8, 63349-63358.

[15] Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv preprint arXiv:1707.01926.

[16] Ji, G., He, S., Xu, L., Liu, K., & Zhao, J. (2015, July). Knowledge graph embedding via dynamic mapping matrix. In Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers) (pp. 687-696). Copyright Statement:rightsretained

[17] Chen, C., Petty, K., Skabardonis, A., Varaiya, P., & Jia, Z. (2001). Freeway performance measurement system: mining loop detector data. Transportation Research Record, 1748(1), 96-102.