Traffic Prediction based on Spatio-Temporal Modeling and Alternative Data Mining

Chenbo Xi*
xichb@shanghaitech.edu.cn
ShanghaiTech University
Shanghai, China

Yucong Chen*
chenyc@shanghaitech.edu.cn
ShanghaiTech University
Shanghai, China

Yirui Yuan*
yuanyr@shanghaitech.edu.cn
ShanghaiTech University
Shanghai, China

Boyang Liu*
liuby1@shanghaitech.edu.cn
ShanghaiTech University
Shanghai, China

Abstract

Accurate and real-time traffic prediction is of great significance for urban traffic planning and traffic management. However, this problem is challenging due to the constraints of urban road network topology and the laws of dynamic changes over time, i.e., spatial dependence and temporal dependence. Although previous work has made great efforts in learning the temporal dynamics and spatial dependence of traffic, some of the current models still fail to encode the complex traffic transition patterns, or ignore the complex urban area planning and fail to distinguish the zoning of cities with different characteristics. Based on the current cutting-edge research in this field, we try to construct a spatio-temporal based multidimensional data network and combine it with alternative data like POI(i.e. point of interest) data to try to predict urban traffic flow, expecting to achieve better results.

Keywords: Alternative data mining, traffic prediction, spatio-temporal dependency modeling, Point of Interest

1 Introduction

ST prediction is an important task for learning systems operating in dynamic environments, which has a wide range of applications. In this paper, we study an important task: traffic prediction for road networks, which is a core component of intelligent transportation systems. Traffic prediction is a key part of an advanced traffic management system and is an important component for achieving traffic planning and traffic management.[5] Traffic forecasting is the process of analyzing the traffic conditions of urban roads,it not only provides a scientific basis for traffic managers to sense traffic congestion and restrict vehicles in advance, but also provides a guarantee for urban travelers to choose suitable travel routes and improve travel efficiency. However, traffic forecasting has been a challenging task due to its complex spatial and temporal dependencies.

(1) **Time Dependence**. Traffic volume changes dynamically over time, mainly in terms of periodicity and trend. For example, traffic volumes can be influenced by traffic conditions in the previous moment or even longer. At the same time, traffic volumes are extremely similar within a specific cycle, measured in weeks.

(2) **Spatial Dependence**. Traffic volume changes are governed by the topology of the urban road network. Traffic conditions on upstream roads influence traffic conditions on downstream roads through the transmission effect, and traffic conditions on downstream roads influence traffic conditions upstream through the feedback effect.

To solve the above problems, we propose a new traffic prediction method which combine innovative network structure with the alternative data such as POI (i.e., point of interest) data and weather data to try to predict urban traffic flow, expecting to achieve better results.

2 Related Work

In recent years, many achievements have emerged in the field of traffic prediction due to the rise of industries such as self-driving techniques and smart cities. Recently, there are few surveys that review this topic, such as [10] and [9].

Recent state-of-the-art methods mainly use deep learning frameworks, [11] is a representative work of traffic prediction by combining **GCN** and **GRU** [2] as the **T-GCN** to modeling spatial and temporal dependency respectively. However, most of these methods do not make full use of the alternative data like holidays and map queries. For example, [3] focuses on developing a dynamic graph convolution method to solve the spatial topology from the road networks, and [6] explored the the application of diffusion convolution.

There are also several studies that take the auxiliary information into account such as [1, 13]. [7] considered the physical property of roads and social events like holidays as well as the topology of roads. Some enterprises use the query data in their map applications as auxiliary information, such as [8]. However, the number of works that consider those alternative data is small and their utilization of datasets is

*All four authors contributed equally to this research.

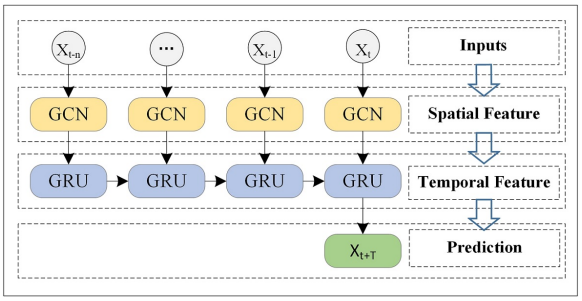


Figure 1. Overview[12] of T-GCN. The input is a series of X by time. The network takes the input with period length of n, and then obtains the spatial features by GCNs. After that, it feeds them into GRU [2] to get the temporal features.

inefficient. Also, the private datasets owned by enterprises won't be accessed publicly.

3 Problem Statement

Based on the cutting-edge research of scholars in this field, we would like to explore the impact of multidimensional data on the accuracy of traffic prediction, which is defined by the traffic speed of each road section at a time. By mining the existing data set, we hope to integrate the above multidimensional data to build an innovative network model, combine it with the alternative data and test it on a specific data set to validate our network prediction performance, including the design of model parameters, analysis of prediction results, perturbation analysis and model interpretation.

4 Methodology

4.1 Spatio-temporal Based Multidimensional Data Network

We can take a typical traffic prediction as a regression problem that has a specific interpretation of its dependencies.

We generate a road-road adjacent matrix containing the spatial information and a t-road speed matrix, containing the temporal information. We used the generic T-GCN [12] framework to help us build the overall workflow. The T-CGN combines GCN into a time series model as GRU (Figure 1).

As adding the alternative data, we refer to the method of [1] building a attention-based T-GCN to automatically select the information needed. In experiments, we concate the alternative data on the channel level of the original data, and feed them into our models. Besides, for the input data in training, we add a Gaussian noise as the data augmentation to increase the robustness of our training process.

In the experiment, we found that this kind of data is characterized by a highly similar input and output, which is consistent with the application ideas of residual networks[4]. To this end, we add a residual connection from thhe input and the final fully-connected layer. Therefore we constructed a refined model called Res-A3T-GCN.

4.2 Alternative Data Combination

We load POI data from Gaode map API. For every road, the data contains several most significant POIs in a fixed range, which is decided by the user search index. Then we count the frequecy of each type of POI, and set the type with maximum counts as the POI type of this road. Since the POI does not change over time, we can generate a type-road 1× n matrix.

5 MapReduce

MapReduce is a parallel programming model and methodology. We use it to process our data to improve the efficiency of our method.

5.1 Scenario

We apply MapReduce to the part of the spatio-temporal based multidimensional data network construction. We use the sparese adjacency matrix $AdjMat$ from the dataset to construct the topological sturcture of the road. Such adjacency marix has the size of N^2, where N represents the number of roads. If road p and road q are connected, the (p, q) item in $AdjMat$ would be 1, otherwise it would be 0. We then transfer it to a dense matrix $EdgeMat$ which store the intersection relations between roads by record a road pair which represents the intersection of two roads.

In our ordinary processing flow, we traverse the sparse matrix $AdjMat$ to construct the dense matrix $EdgeMat$. However, when dealing with more roads, the size of the adjacency matrix may become extremely huge, which would lead to a low efficiency of the process. In order to improve the efficiency of our method, we use MapReduce to transfer the $AdjMat$ to the $EdgeMat$.

5.2 Logic

Here are two parts of our MapReduce method: Mapper and Reducer. With the help of the API of Hadoop, we use $STDIN$ and $STDOUT$ to transmit data from the Mapper to the Reducer. The Map function traverse the row in $AdjMat$, detect which roads are connected with the current road, and print the road pair data to the Reduce funtion. The Reduce function collect the data from the Mapper and aggregate them to build the $EdgeMat$.As shown in Figure 2, we pass the $AdjMat$ as an input to HDFS, and call the Mapper function and the Reducer function to get the output data, then apply post processing to get the $EdgeMat$.

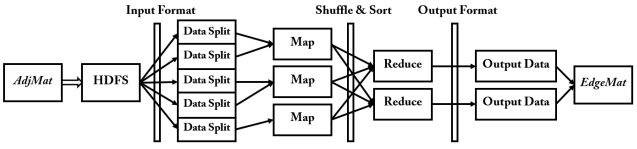


Figure 2. Pipeline of MapReduce

5.3 Version & Environment

We deploy the Hadoop on Windows11 with JDK8, Hadoop-3.1.3 and apache-hadoop 3.1.0 winutils. We use the hadoop-streaming-3.1.3.jar to run the MapReduce code. You can find more details about how to run it in our Readme.md file.

5.4 Performance Enhancement

We test the performance of MapReduce on two dataset, sz_taxi.csv(49KB) and Q-traffic.csv(315.1MB) on Surface go pro 2 with Windows 11. As shown in Table 1, although MapReduce does not perform as well as ordinary method(i.e., for loop) on the smaller sz_taxi datasets, it shows great computing efficiency when faced with a larger dataset, consuming only 3.54% of the time of the ordinary method. Based on the results, we can conclude that MapReduce can make a incredible performance enhancement when dealing with big dataset.

Dataset	For Loop	MapReduce
SZ-Taxi dataset	0.23s	0.31s
Q-Traffic dataset	1270.4s	45.6s

Table 1. Performance comparison between ordinary methods and MapReduce

6 Experiments

6.1 Datasets

We chose two public data sets in the field of traffic prediction for experiments. They are often used in the research communities of spatio-temporal data.
To see the introduction of our datasets, please check the appendix.

6.1.1 POI. Since Q-Traffic dataset does not have proper poi data. We get the poi information from Gaode map api according to the latitude-longitude range corresponding to different compressibilty. The poi information contains the position and type of every poi. The order of pois around a road is decided by the weights of every type of poi according to the search index. Then we count every type of poi around the road and regard the type with maximum counts as the poi type of this road. Since the poi information does not change over time. The poi matrix is the same form as sz-taxi.

6.2 Data Visualization

Since we only have the location of the velocimeter on each road, suppose here's a node, we use the average of location of all roads one of whose end is this node as the location of this node. Then we generate the road network graph with networkx, according to the latitude and longitude of every node and the topological structure of roads. Fig.4 shows the spatial structure of roads in Beijing with compressibility as 10. Fig.3 shows the distribution of points of interest.

Figure 3. POI data visualization

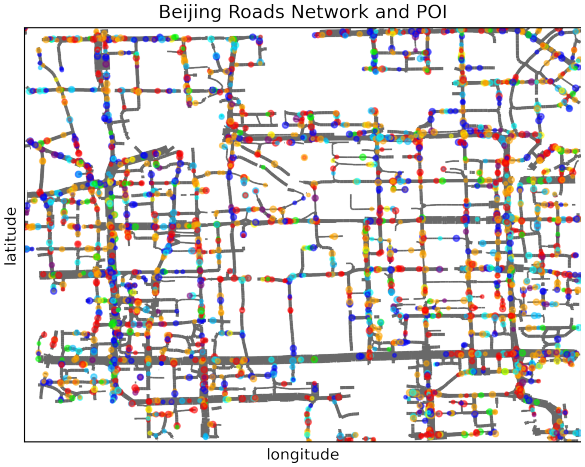


Figure 4. The grey roads are the distribution of roads in Beijing. The circles with different colors and sizes are pois. Different colors represent different types of poi and the size represents the significance weight of the poi. The larger the size, the more the significance is.

6.3 Evaluation Metrics

To evaluate the prediction performance of our model, we use four metrics to evaluate the difference between the real traffic information Y and the prediction $\hat{Y}$, which are matrices of shape $(I \times J)$ and each row represents a road expanding along the column at different timestamps. The formulations are as follows:

- Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{IJ}\sum_{i,j}\left(Y_{ij} - \hat{Y}_{ij}\right)^2} \tag{1}$$

- Mean Absolute Error (MAE)

$$MAE = \frac{1}{IJ}\sum_{i,j}\left|Y_{ij} - \hat{Y}_{ij}\right| \tag{2}$$

Table 2. Performance comparison between different methods and alternative data on SZ-Taxi. The values in red are the best compared to others, and the values in blue are the second best.

Methods	Alternative Data	RMSE ↓	MAE ↓	Accuracy ↑	R2 ↑
GRU	poi & weather	5.6273	4.2043	0.6080	0.7093
	poi	5.5826	4.1257	0.6111	0.7139
	weather	5.8682	4.3557	0.5912	0.6839
	NA	5.9267	4.3997	0.5871	0.6775
A3T-GCN	poi & weather	4.1368	2.7979	0.7118	0.8429
	poi	4.1371	2.7955	0.7118	0.8429
	weather	4.1375	2.8079	0.7118	0.8428
	NA	4.1375	**2.7906**	0.7118	0.8428
Res-A3T-GCN	poi & weather	**4.1326**	2.7998	**0.7121**	**0.8432**
	poi	4.1352	2.8193	0.7119	0.8430
	weather	**4.1302**	**2.7898**	**0.7123**	**0.8434**
	NA	4.1402	2.8551	0.7116	0.8426

- Accuracy:

$$Accuracy = 1 - \frac{\|Y - \hat{Y}\|_F}{\|Y\|_F} \tag{3}$$

- Coefficient of Determination (R^2):

$$R^2 = 1 - \frac{1}{I}\sum_i \frac{\sum_j (Y_{ij} - \hat{Y}_{ij})^2}{\sum_j (Y_{ij} - \bar{Y}_i)^2}$$
$$\text{where } \bar{Y}_i = \frac{1}{J}\sum_j Y_{ij} \tag{4}$$

The Eq.1 and Eq.2 measure the error between predictions and groundtruth from two aspects. The Eq.3 defines matchness between the two matrices, and Eq.4 focuses on the correlation between the time series.

6.4 Experimental Results

We utilized the available datasets and computing resources, and then designed a series of experiments to prove our conjectures. Our experiments mainly focuses on two perspectives: the performance comparison of our refined methods with baselines and SOTA methods; The analysis on influences from alternative data.

We choose the standard GRU as the baseline model for our experiments, as it is a small but powerful model for temporal data. As it only considers the temporal informantion in data, we use its as the baseline to test the gain from modelling spatial information.

For SOTA methods, there are many previous outcomes leading in different sub-areas or utilizing different functionalities, as we listed in related works. Here we choose [1] as one of them to be a reference.

Although some of the results are published in previous papers before, we redo all the experiments about the baseline and SOTA as we are able to set the proper hyper-parameters for all methods in a fair competetion then.

6.4.1 Performance Comparison on SZ-Taxi. This sets of experiments verifies the performance of GRU, A3T-GCN and our refined method Res-A3T-GCN under different settings of alternative data, as in Table 2. We can find that our refined method Res-A3T-GCN outperforms baseline significantly. This shows the importance of spatial information for the traffic predictions. Besides, our refined model surpasses SOTA from all perspectives also. For the alternative data, we found it works differently in different methods. Most of the time the poi & weather data help the model to learn better, but in our refined model we found that we use the weather without poi reaches the best performance. This phenomenon needs to be further studied, and we speculate that improving the integration and coding of data can make breakthroughs in this regard. We selected a road to visualize our result as in Figure 5

6.4.2 Performance Comparison on Q-Traffic. Similarly, we conduct experiments for GRU, A3T-GCN and our Res-A3T-GCN under different alternative data. For all the 3 versions of Q-Traffic, the our refined method with poi data achieves the best performance. We selected a road to visualize our result as in Figure 6

As we plot in Figure 8, we show that our model with proper alternative data correctly estimate the speed distribution over

Table 3. Performance comparison between different methods on Q-Traffics, best values are **bolded**.

Datasets	Methods	RMSE ↓	MAE ↓	Accuracy ↑	R2 ↑
Q-Traffic10	GRU	4.5241	3.1521	0.8626	0.8293
	A3T-GCN	3.6857	2.4108	0.8881	0.8867
	Res-A3T-GCN	**3.5842**	**2.3484**	**0.8912**	**0.8929**
Q-Traffic25	GRU	3.9060	2.7632	0.8621	0.7793
	GRU-poi	3.9229	2.7658	0.8615	0.7774
	A3T-GCN	3.3605	2.2402	0.8814	0.8366
	A3T-GCN-poi	3.3559	2.2403	0.8815	0.8371
	Res-A3T-GCN	3.1105	2.0915	0.8902	0.8600
	Res-A3T-GCN-poi	**3.0985**	**2.0692**	**0.8906**	**0.8611**
Q-Traffic50	GRU	4.0555	2.8330	0.8608	0.8175
	A3T-GCN	3.5980	2.4262	0.8765	0.8563
	Res-A3T-GCN	**3.2944**	**2.1789**	**0.8869**	**0.8796**

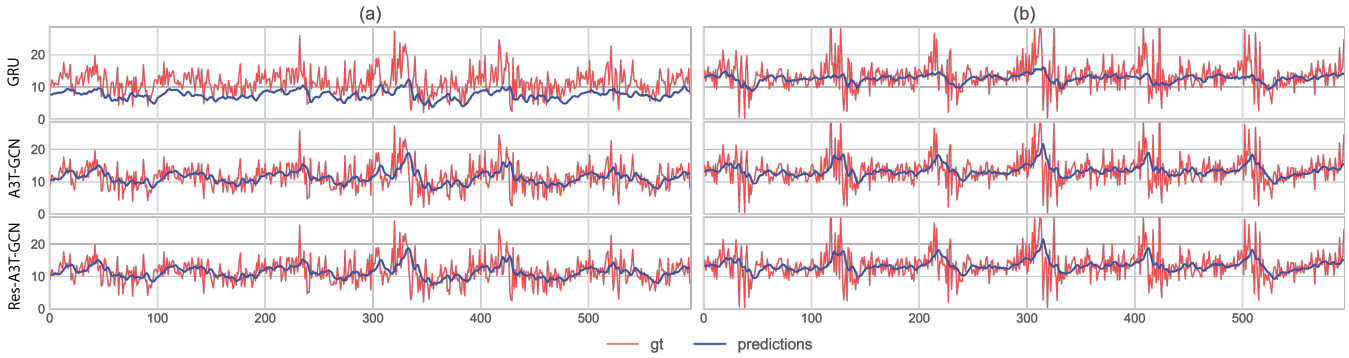


Figure 5. The predictions from three methods using both poi and weather information on two roads (col. a and b respectively). The x-axis represents the time and y-axis is speed. We can see that our method outperforms others.

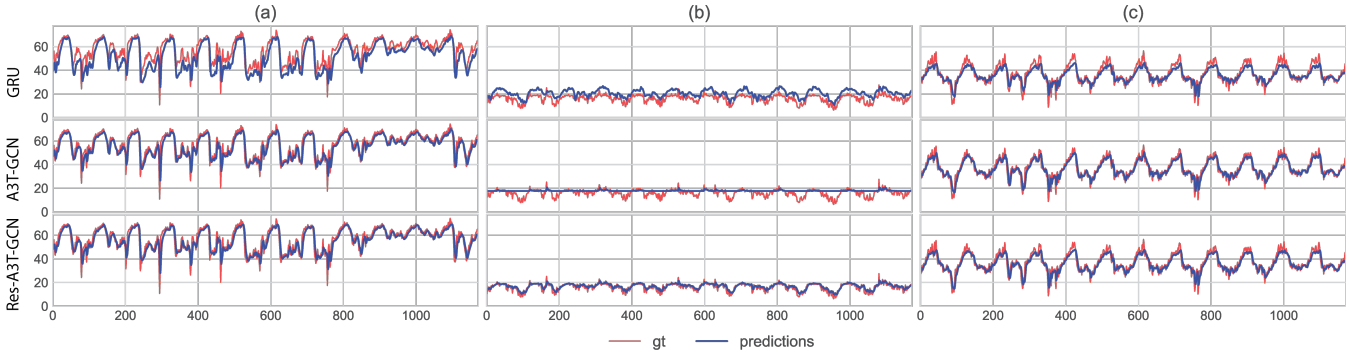


Figure 6. The predictions from three methods using both poi and weather information on three roads of three datasets (col. a, b, c; Q-Traffic10, 25, 50 respectively). The x-axis represents the time and y-axis is speed. We can see that our method outperforms others.

the dataset. In Figure 7, we plot the prediction results of our model at a certain moment, indicating that it is applicable to the real world functionalities.

7 Conclusion

We design and implement a traffic predicting system that combine spatio-temporal based multidimensional data network and combine it with alternative data like POI and weather. The experiments on SZ-Taxi and Q-Traffics datasets show that our model can achieve the highest accuracy compared with GRU and the SOTA traffic prediction method(i.e., A3T-GCN). We have also conducted analysis and experiments for combining alternative data with GRU and A3T-GCN, as a result, the methods combining alternative data were all better performing than the original methods. Limited by time and computing resources, we cannot further explore how to embed more alternative data into our traffic predicting pipeline to achieve better performance, but our results prove that the combination of alternative data and the traditional traffic predicting method is feasible and effective.

References

[1] Jiandong Bai, Jiawei Zhu, Yujiao Song, Ling Zhao, Zhixiang Hou, Ronghua Du, and Haifeng Li. 2021. A3T-GCN: Attention Temporal Graph Convolutional Network for Traffic Forecasting. *ISPRS International Journal of Geo-Information* 10, 7 (July 2021), 485.

[2] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[3] Liangzhe Han, Bowen Du, Leilei Sun, Yanjie Fu, Yisheng Lv, and Hui Xiong. 2021. 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* (Virtual Event, Singapore) *(KDD '21)*. Association for Computing Machinery, New York, NY, USA, 547–555. https://doi.org/10.1145/3447548.3467275

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[5] M. Li and Z. Zhu. 2020. Spatial-Temporal Fusion Graph Neural Networks for Traffic Flow Forecasting. (2020).

[6] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. (2018).

[7] Binbing Liao, Jingqing Zhang, Chao Wu, Douglas McIlwraith, Tong Chen, Shengwen Yang, Yike Guo, and Fei Wu. 2018. Deep Sequence Learning with Auxiliary Information for Traffic Prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 537–546.

[8] Hao Liu, Yongxin Tong, Panpan Zhang, Xinjiang Lu, Jianguo Duan, and Hui Xiong. 2019. Hydra: A Personalized and Context-Aware Multi-Modal Transportation Recommendation System. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA) *(KDD '19)*. Association for Computing Machinery, New York, NY, USA, 2314–2324. https://doi.org/10.1145/3292500.3330660

[9] Eleni I. Vlahogianni, Matthew G. Karlaftis, and John C. Golias. 2014. Short-term traffic forecasting: Where we are and where we're going. *Transportation Research Part C: Emerging Technologies* 43 (2014), 3–19. https://doi.org/10.1016/j.trc.2014.01.005 Special Issue on Short-term Traffic Flow Forecasting.

[10] Xueyan Yin, Genze Wu, Jinze Wei, Yanming Shen, Heng Qi, and Baocai Yin. 2021. Deep Learning on Traffic Prediction: Methods, Analysis and Future Directions. *IEEE Transactions on Intelligent Transportation Systems* (2021), 1–17. https://doi.org/10.1109/TITS.2021.3054840

[11] Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. 2020. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems* 21, 9 (Sept. 2020), 3848–3858.

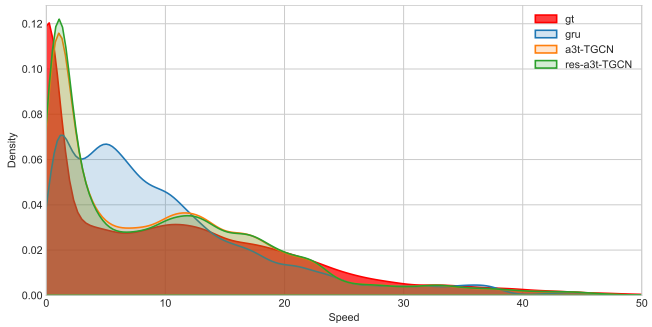
[12] Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. 2020. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Trans. Intell. Transp. Syst.* (2020).

[13] Jiawei Zhu, Qiongjie Wang, Chao Tao, Hanhan Deng, Ling Zhao, and Haifeng Li. 2021. AST-GCN: Attribute-Augmented Spatiotemporal Graph Convolutional Network for Traffic Forecasting. *IEEE Access* 9 (2021), 35973–35983.

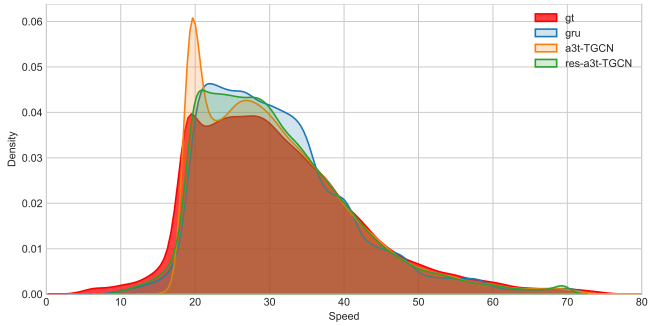
A Appendix-Datasets Introduction & Visualization

A.0.1 SZ-Taxi. This dataset is firstly published by [11] and it is publicly available at https://www.github.com/lehaifeng/T-GCN. It contains the taxi trajectory records in Shenzhen from January 1 to January 3, 2015. As it coincides with the New Year's holiday, the data during this period are more irregular than those on weekdays, this feature can help us to better verify the generalization of algorithms. A 156*156 adjacency matrix is used to model the connection of 156 main road sections in the Luohu District. The traffic speed time series of selected sections are calculated and organized into a feature matrix, with road sections indexed in the rows and timestamps in the columns. In addition, SZ-Taxi integrates good alternative data. SZ-POI and SZ-Weather [13] are two auxiliary datasets along with SZ-Taxi as our alternative data here. The poi information divided the functional sections surrounding selected main roads into nine catagories: catering services, enterprises, shop- ping services, transportation facilities, education services, living services, medical services, accommodations, and others. Since the poi will not change in a limited period, the data size is 156*1. The SZ-Weather is similar to the feature matrix that describes the traffic speed, but its entries record the weather catagories (sunny, cloudy, fog, light rain, and heavy rain) instead of continous speed values.

A.0.2 Q-Traffic. This dataset is proposed by [7] and it is publicly available at https://github.com/JingqingZ/BaiduTraffic. It contains three sub-datasets, and we choose to use RoadNetwork sub-dataset and TrafficSpeed sub-dataset. The datasets are much more original and complex than sz-taxi dataset, which includes the whole of Beijing and contains 45158 roads. In order to improve efficiency, we simplify the dataset according to the latitude and longitude of roads and some attributes of roads. First, we remove roads whose length is smaller than 0.01. Than we calculate the distribution of location of roads and get the minimum, maximum, average of them. Then we choose the compressibility from [10, 25, 50] to compress the dataset. We view the minimum to maximum as the original range and compress the range centered on

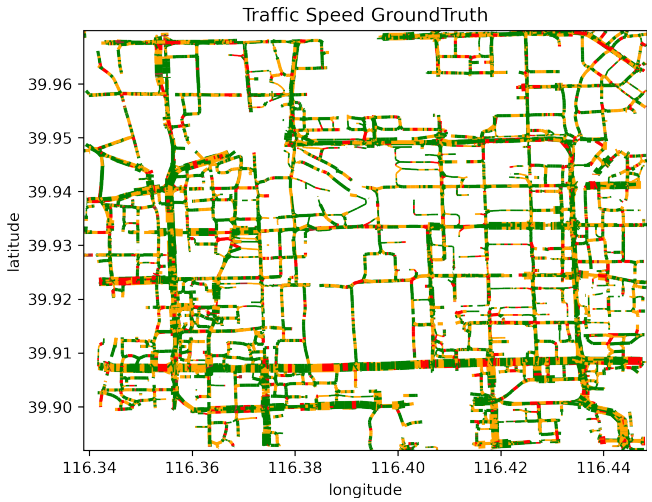


(a) Speed distributions of SZ-Taxi.

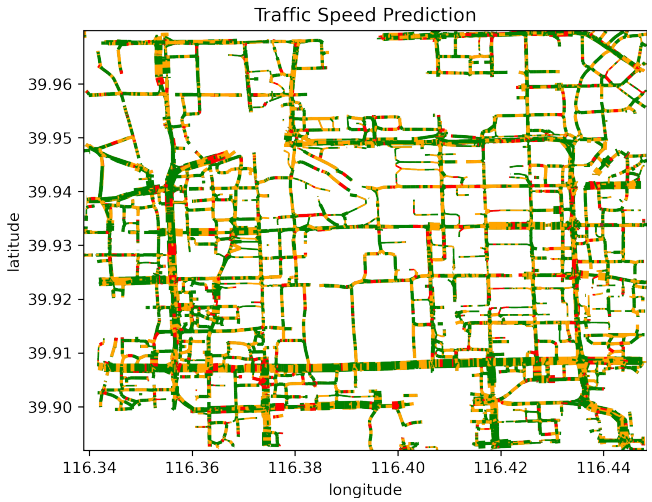


(b) Speed distributions of Q-Traffic.

Figure 8. Speed distributions in ground-truth and the predictions.



(a) GT: brown: spd $<$ 10, red: 10 $\leq$ spd $<$ 20, orange: 20 $\leq$ spd $<$ 30, green: spd $\geq$ 30



(b) Prediction: brown: spd $<$ 10, red: 10 $\leq$ spd $<$ 20, orange: 20 $\leq$ spd $<$ 30, green: spd $\geq$ 30

Figure 7. Speed visualization on map of Q-Traffic, at a moment.

the mean latitude and longitude in proportion to the compressibility. Then we generate the adjacent matrix from the road-(start_node, end_node) information in RoadNetwork dataset and generate the feature matrix from TrafficSpeed dataset.