目录

1. Introduction

With the great development of modern cities, the rapid growth of population and the acceleration of urbanization has made transportation systems to an essential infrastructure. In the meantime, transportation systems are becoming more and more complex, which causes great pressure on urban traffic management. As a result, it is important to develop the Intelligent Transportation System (ITS)[1] for efficient traffic management.

Modern transportation systems contain road vehicles, railway transportation and a variety of newly emerged shared travel modes, including online ride-hailing, bike-sharing, etc.. In order to alleviate transportation related problems and manage the expanding transportation systems efficiently, traffic prediction or traffic forecasting is brought up for ITS by researchers in recent years. Traffic prediction is the process of analyzing urban road traffic conditions, including flow, speed and density, mining traffic patterns, and predicting road traffic trends. Traffic prediction can not only provide a scientific basis for traffic managers to perceive traffic congestion and limit vehicles in advance, but also provide a guarantee for travelers to choose proper travel routes and improve travel efficiency.

Traffic prediction is typically based on consideration of historical traffic state data. In the development of intelligent transportation systems, traffic states are detected by traffic sensors, bus and metro transactions logs, traffic surveillance cameras and GPS devices. However, traffic state data is hard to manage because it involves large data volumes with high dimensionality. Its typical characteristic is that it contains both spatial and temporal domains. Therefore, traffic prediction becomes a challenging topic because of spatial and temporal dependencies.

1. **Spatial dependency.** Urban road network has a topological structure that seriously affects the change of traffic state of each road. To be specific, the upstream traffic state influences the downstream roads for the reason like vehicle transfer.

2. **Temporal dependency.** Traffic state varies over time with periodicity. For example, in general, the traffic state over weekdays are similar to each other but has a huge difference with holidays, and vice versa. In detail, the traffic state at a specific moment is impacted by the previous moments or even hours.

Traditional time series prediction models (e.g., Moving Average (MA), Auto-regressive (AR), Auto-regressive Integrated Moving Average (ARIMA)) cannot handle such spatiotemporal prediction scenarios well. Therefore, to address the complex dependencies, deep learning methods have been introduced to this area.

Graph convolution networks (GCN)[2] becomes popular in recent years due to its ability to capture spatiotemporal dependencies efficiently. Many GCN-based models reached state-of-the-art performance, such as STGCN[3], DCRNN[4], Graph WaveNet[5] and AGCRN[6]. To represent road network, a graph is constructed where each node in the graph stands for a road segment or a traffic sensor. And edges means connectivity between road segments or sensors. As a result, spatial dependency can be extracted directly from the graph. Concerning temporal dependency, every node is linked with a feature vector that consists of traffic states at each moment. Several different methods were applied such as recurrent neural networks (RNN) and 1D convolutions. As mentioned above, in GCN-based models, spatial dependency is expressed only by the relationship among nodes in the graph. However, the traffic condition in real world is much more complicated. For example, the main roads in a city are often congested during peak hours. Although it is usually the shortest path to travel through main roads, commuters will probably prefer a father but clearer path. That is, the graph only shows the road connectivity which cannot represent the transfer preference by real drivers. Despite that it is impractical to collect all the traffic patterns, the trajectories reflect them well and thoroughly. In addition, when counting road flow or calculating road traffic speed, a trajectory is treated as discrete points, while the road transfer information naturally lies in the sequential order of the trajectory. Fortunately, such trajectories can be tracked by GPS devices and mobile apps with GPS service, and we have a completely raw GPS dataset which is copied directly from the logs of GPS devices in Shenzhen's taxis.

Based on these facts, we believe that the trajectories will give us the actual road transfer information. By analyzing road transfer, we propose a concept named **trajectory-based road correlation** that stands for the relevance or similarity among roads. With this, a better spatial dependency can be captured. Therefore, the focus of this paper is to design a general method to extract road correlation through trajectories and utilize it for state-of-the-art neural networks to predict traffic state.

To summarize, in this paper, we propose a procedure to learn trajectory-based road correlation via GPS data and use it to improve traffic state prediction.

The contribution of our paper is:

- We build a road-network-based trajectory dataset upon completely raw GPS data.
- We proposed a procedure to learn road correlation through trajectories.
- We refine traffic state prediction by utilizing the trajectory-based road correlation.

2. Related Work

Public Traffic Datasets. There are several public traffic datasets which are frequently used for traffic prediction. They can be briefly categorized into three classes by spatial domain, which are **grid-based**, **sensor-based** and **road-network-based**.

For grid-based datasets, there are *TaxiBJ*[7] that consists of the taxi in and out flow data in Beijing, and *TaxiNYC* for taxis in New York City published by the New York City Taxi and Limousine Commission (TLC). For sensor-based datasets, *METR-LA*[4] and *PEMS-BAY* are the most widely used datasets in urban traffic prediction area. In detail, *PEMS-BAY* is collected from 325 sensors all over the San Jose bay area every 5 minutes. The traffic sensors can directly record the traffic flow of each road, which makes the dataset easy to handle and process. And for road-network-based datasets, *Didi GAIA*'s open data has a good quality but they are seldom applied to build a model. It is GPS data containing taxi locations with timestamp that collected by *Didi* company's mobile app, which is similar to ours. In conclusion, as suggested by Jiang and Luo[8], most traffic prediction models are built upon traffic sensor datasets, while road-network-based datasets are mainly used for test. Therefore, we need to make better use of it.

Road Network Modeling. Road network is the basic component of urban traffic system. To make use of the spatial information inside it, many approaches have been proposed. Statistical models are used to represent road network. For recent traffic prediction articles, Li et al.[9] model road transition as a Markov Process over road network and use a first order Markov matrix to represent it. The growth of deep learning models makes it possible to model more complex road network and learn road characteristics efficiently. In basic GCN[2], the authors use adjacency matrix to calculate Graph Laplacian Matrix in order to represent the whole graph. Lately, Wu et al.[10] proposed a hierarchical graph neural networks to capture both structural and functional characteristics of road network through several pre-defined attributes of each road. Wu et al.[5] use graph convolution to learn a new adjacency matrix of sensor graph, which is quite related to our work. To conclude, the two methods mentioned above need prior knowledge or history traffic state of roads. In contrast, our work is to learn a representation of each road to model its spatial characteristics only by trajectories.

Traffic Prediction Models. Early attempts use traditional time series forecasting model including ARIMA[11] and VAR[12], as well as machine learning techniques like k-NN[13] and SVM[14]. As mentioned in section 1, these models cannot capture the spatiotemporal dependency well. Many state-of-the-art deep neural networks have been proposed in the last several years. Yu et al.[3] proposed two different convolution blocks to capture spatial and temporal dependencies separately. Li et al.[4] take advantage of seq2seq[15] architecture and perform diffusion convolution on the graph. From our observation, few existing work leverage trajectories in traffic prediction. Hui et al.[16] extract the temporal features of roads by convolution with recent, daily-periodic and weekly-periodic traffic state data. Then they perform feature smoothing by propagating features through trajectories. On the contrary, our work attempts to combine the spatial representation that learned from trajectories into traffic state prediction models.

3. Preliminaries

In this section, we will introduce the notations used in this paper and problem definitions in our task.

3.1 Notations

Table 1 Notations

Notation	Definition
n_r	#roads
$\mathcal{R}$	road set
r	a single road in $\mathcal{R}$
$\mathcal{E}$	edge set
A	adjacency matrix
$\mathcal{G}$	road network graph
$\mathcal{T}$	trajectory set
T	a trajectory in $\mathcal{T}$
ts	timestamp
s	speed
E	road embedding matrix
$\mathbf{e}_i$	embedding vector for road r_i
d_r	dimension of embedding vectors
C	road correlation matrix
t	time interval
n_t	#time intervals
X	traffic state matrix
$\mathbf{x}_t$	traffic state vector at time interval t

The above table 1 gives the notations and their definitions.

3.2 Problem Definition

This section gives the definitions[9] of the concepts and tasks occurred in this paper.

Definition 1 (Road Network Graph) *The road network can be represented by a directed graph $\mathcal{G} = (\mathcal{R}, \mathcal{E}, A)$, where $\mathcal{R} = \{r_1, r_2, \ldots, r_{n_r}\}$ is a finite set of roads that each r_i stands for a real road in the road network. $\mathcal{E}$ is the set of directed edges where $(r_i, r_j) \in \mathcal{E}$ indicates that there is a directed edge from r_i to r_j, i.e. r_j is the downstream road in the road network. $A \in [0, 1]^{n_r \times n_r}$ is the adjacency matrix whose entry A_{ij} is a binary value that indicates whether there exists an edge $(r_i, r_j) \in \mathcal{E}$.*

Definition 2 (Trajectory) *Given a road network graph $\mathcal{G} = (\mathcal{R}, \mathcal{E}, A)$, a trajectory $T = [(r_1, s_1, ts_1), (r_2, s_2, ts_2), \ldots, (r_l, s_l, ts_l)]$ is a sequence of (road, speed, timestamp) tuples. Each tuple (r_i, s_i, ts_i) specifies that the vehicle is driving on r_i with speed s_i at timestamp ts_i. Besides, $\forall i = 1, 2, \ldots, l-1$, $r_i \neq r_{i+1}$ and $(r_i, r_{i+1}) \in \mathcal{E}$.*

Definition 3 (Traffic State) *Traffic state stands for the traffic flow or speed of a road during a particular time interval. Traffic flow is defined as the number of vehicles passing* *by the road, and traffic speed is the average speed of these vehicles. For a road graph $\mathcal{G} = (\mathcal{R}, \mathcal{E}, A)$, we use $X \in \mathbb{R}^{n_r \times n_t}$ to record the traffic state of each time interval. For time interval t, $\mathbf{x}_t = X_{:,t} \in \mathbb{R}^{n_r}$ represents the traffic state of all roads during t.*

Problem 1 (Road Correlation) *Given a road network graph $\mathcal{G} = (\mathcal{R}, \mathcal{E}, A)$, find a road correlation function Cor which takes two roads as input and returns a real number $0 \leqslant Cor(r_i, r_j) \leqslant 1$ to quantify the spatial dependency between two roads r_i and r_j. The value is bigger if the two roads have a stronger dependency, e.g. r_i is the only way to r_j. The road correlation matrix C stores all the correlation values s.t. $C_{ij} = Cor(r_i, r_j)$.*

Problem 2 (Traffic State Prediction) *Given a road network graph $\mathcal{G} = (\mathcal{R}, \mathcal{E}, A)$, find a function f and its parameter set Θ s.t. given historical traffic states $\{\mathbf{x}_{t-\tau_{in}+1}, \mathbf{x}_{t-\tau_{in}}, \dots, \mathbf{x}_t\}$ for an input window τ_{in}, f estimates the most likely traffic states $\{\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \dots, \mathbf{x}_{t+\tau_{out}}\}$ for an output window τ_{out}.*

$$\hat{X}_{:,t+1:t+\tau_{out}} = f_\Theta(X_{:,t-\tau_{in}+1:t-1}) = \underset{X_{:,t+1:t+\tau_{out}}}{\arg\max}\ p(X_{:,t+1:t+\tau_{out}} | X_{:,t-\tau_{in}+1:t-1}) \tag{1}$$

4. Dataset

This section introduces how we build the whole dataset from raw data.
TODO: 这里插一张总体流程图

4.1 Data Description

Our data is taken from the records of GPS devices on the taxis in Shenzhen. A brief description is as the following:

- **Region:** Shenzhen
- **Time Range:** June 2020
- **Content:** Taxi GPS records
  - License number
  - Longitude and latitude
  - Speed
  - Timestamp
- **Size:** Over 2,500,000,000 rows

A small part of data is shown as an example in figure 1.

Unlike the open datasets that can be applied to deep learning models without the need of data cleaning and completion, this raw dataset contains lots of abnormal values, which should be cleaned and re-organized carefully.

	sys_time	license_number	lng	lat	gps_time	EMPTY1	speed	direction	car_status	alarm_status	EMPTY2	EMPTY3	license_color	recorder_speed	mileage
0	2020-06-01 00:00:01	粤BD[redacted]	113.98681	22.529696	2020-05-31 23:59:48	NaN	0	40	0	0	NaN	NaN	蓝色	0	2081590
1	2020-06-01 00:00:01	粤BD[redacted]	113.96201	22.536120	2020-05-31 23:59:49	NaN	0	0	0	0	NaN	NaN	蓝色	0	686220
2	2020-06-01 00:00:01	粤BD[redacted]	114.04288	22.598593	2020-05-31 22:22:57	NaN	0	173	0	0	NaN	NaN	蓝色	0	1894000
3	2020-06-01 00:00:01	粤BD[redacted]	0.00000	0.000000	2020-05-31 23:59:49	NaN	0	0	0	32	NaN	NaN	蓝色	0	2484210
4	2020-06-01 00:00:01	粤BW[redacted]	0.00000	0.000000	2000-01-01 00:00:00	NaN	0	0	0	0	NaN	NaN	蓝色	0	0
...	...	[redacted]	...	...	...	...	...	...	...	...	...	...	...	...	...
9999995	2020-06-01 02:28:27	粤BD[redacted]	113.92077	22.611652	2020-06-01 02:27:05	NaN	56	90	0	0	NaN	NaN	蓝色	56	3069870
9999996	2020-06-01 02:28:27	粤BD[redacted]	114.13057	22.610834	2020-06-01 02:28:16	NaN	0	53	512	0	NaN	NaN	蓝色	0	4341550
9999997	2020-06-01 02:28:27	粤BD[redacted]	113.81205	22.622503	2020-06-01 02:23:26	NaN	29	178	0	0	NaN	NaN	蓝色	29	0
9999998	2020-06-01 02:28:27	粤BD[redacted]	113.98769	22.590467	2020-06-01 02:28:15	NaN	51	123	0	0	NaN	NaN	蓝色	51	922650
9999999	2020-06-01 02:28:27	粤BD[redacted]	113.25477	23.175537	2020-05-29 14:41:16	NaN	40	175	0	0	NaN	NaN	蓝色	40	3113800

Figure 1 Shenzhen taxi GPS raw data

4.2 Data Cleaning

Data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data[17]. There are many kinds of bad records that should be deleted or modified. To summarize, we categorize them as the following classes.

1. **Duplicate Rows.** A considerable large part of the raw data are duplicate. The reason is when a GPS device is transmitting data to server, it will send several copies in order to avoid packet loss under poor Internet connection. As a result, they are completely same rows, and thus can be removed safely, leaving only the foremost one.

2. **Corrupted Timestamp.** This is a sort of abnormal record. Since our time range is June 2020, all the timestamps that not in here should be deleted. In detail, there are two kinds of them: 1) records in May 31^{st} or July 1^{st}. This is caused by the equipments' lack of accuracy. 2) 2000-01-01. And this is caused by data loss, thus, it is filled by a default value.

3. **Abnormal Location.** The latitude and longitude of some records are zero, which is resulted by the data loss during transmission. These dirty values should be deleted.

4. **Zero Speed.** Stationary taxis are still transmitting their location information to the server if the GPS device is on, leading to a big portion of zero speed records. They are useless owing to that trajectories are a series of moving locations. Therefore, under normal circumstances, it is better to remove them. However, things are not that

happy in our data. There are four kinds of zero speed records relating to the change of location, i.e. latitude and longitude, and they should be treated differently. Details are provided in the next subsection.

5. **Irrelevant Attributes.** As shown in figure 1 above, the raw data consists of several columns. The information that have no contribution to trajectories needs to be removed, leaving only latitude, longitude, speed and timestamp.

We take the data of June 1^{st} as a case study to give an illustration of our data cleaning procedure. In total, there are 97,453,725 rows.

Table 2 Data Cleaning Procedure on June 1^{st}

Action	Deleted Percentage	#Remaining Rows
Drop duplicate	51.73%	47,042,104
Drop abnormal values	1.19%	45,874,548
Drop zero speed	16.84%	29,458,603
Remaining Percentage	30.22%	

4.3 Data Processing

5. Methodology

This is methodology.

6. Experiments

This is experiments.

7. Conclusion and Future Work

This is conclusion.

参考文献

[1] ZHANG J, WANG F Y, WANG K, Data-driven intelligent transportation systems: A survey. IEEE Transactions on Intelligent Transportation Systems, 2011, 12(4): 1624-1639.

[2] KIPF T N WELLING M. Semi-supervised classification with graph convolutional networks. ArXiv preprint arXiv:1609.02907, 2016.

[3] YU B, YIN H, ZHU Z. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting//Proceedings of the 27th International Joint Conference on Artificial Intelligence. [S.l. : s.n.], 2018: 3634-3640.

[4] LI Y, YU R, SHAHABI C, Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting//International Conference on Learning Representations. [S.l. : s.n.], 2018.

[5] WU Z, PAN S, LONG G, Graph wavenet for deep spatial-temporal graph modeling. ArXiv preprint arXiv:1906.00121, 2019.

[6] BAI L, YAO L, LI C, Adaptive graph convolutional recurrent network for traffic forecasting. Advances in Neural Information Processing Systems, 2020, 33: 17804-17815.

[7] ZHANG J, ZHENG Y, QI D. Deep spatio-temporal residual networks for citywide crowd flows prediction//Thirty-first AAAI conference on artificial intelligence. [S.l. : s.n.], 2017.

[8] JIANG W LUO J. Graph neural network for traffic forecasting: A survey. ArXiv preprint arXiv:2101.11174, 2021.

[9] LI M, TONG P, LI M, Traffic flow prediction with vehicle trajectories//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 35: 1. [S.l. : s.n.], 2021: 294-302.

[10] WU N, ZHAO X W, WANG J, Learning effective road network representation with hierarchical graph neural networks//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. [S.l. : s.n.], 2020: 6-14.

[11] WILLIAMS B M, DURVASULA P K, BROWN D E. Urban freeway traffic flow prediction: application of seasonal autoregressive integrated moving average and exponential smoothing models. Transportation Research Record, 1998, 1644(1): 132-141.

[12] CHANDRA S R AL-DEEK H. Predictions of freeway traffic speeds and volumes using vector autoregressive models. Journal of Intelligent Transportation Systems, 2009, 13(2): 53-72.

[13] DAVIS G A NIHAN N L. Nonparametric regression and short-term freeway traffic forecasting. Journal of Transportation Engineering, 1991, 117(2): 178-188.

[14] VANAJAKSHI L RILETT L R. A comparison of the performance of artificial neural networks and support vector machines for the prediction of traffic speed//IEEE Intelligent Vehicles Symposium, 2004. [S.l. : s.n.], 2004: 194-199.

[15] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks. Advances in neural information processing systems, 2014, 27.

[16] HUI B, YAN D, CHEN H, Trajnet: A trajectory-based deep learning model for traffic prediction//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. [S.l. : s.n.], 2021: 716-724.

[17] WU S. A review on coarse warranty data and analysis. Reliability Engineering & System Safety, 2013, 114: 1-11.

致谢

感谢广东省深圳市南山区学苑大道 1088 号南方科技大学工学院南楼 552B 崔氏集团实验室