Graph network models and algorithms using in smart traffic - Project

Congjie AN[1], Haiwei FU[2], Nan CHEN[3], and Yangfan ZHANGLIN[4]

[1] congjie.an@student-cs.fr
[2] haiwei.fu@student-cs.fr
[3] nan.chen@student-cs.fr
[4] yangfan.zhanglin@student-cs.fr

Abstract. In this project, we participated in the Alibaba Tianchi cloud data competition and found a dataset that conforms to the topology model of the graph network structure: the Intelligent Transportation Dataset. We explored and studied the dataset, processed and modeled it according to the requirements of the competition. Firstly, due to the large size of the dataset, we took a small part of it and used a network structure model to study the dataset and output some results. Then, we hope to further explore this dataset to achieve better prediction results, while also exploring the similarities and differences between traditional methods and pure network structure methods on topological structure datasets. Therefore, we borrowed the idea of a participant's data preprocessing and established an XGBoosting model for prediction. Finally, we also explored the performance of some GNN model structures, which serve as a reference for us on a more widely used traffic dataset, further learning the development of network structure model problems under deep learning.

Keywords: Smart Transportation · Network Science · Topology Dataset

1 Introduction

With the opening of the era of mobile Internet, every travellers has become a contributor to traffic information. Super-large scale location data is processed and integrated into the cloud to generate traffic information for the whole period of the city without blind areas.

2 Database and Problem Definition

The transportation network dataset includes information about links, including unique identification numbers, lengths, widths, and road types. Links are arranged in multiple directions, and their upstream and downstream relationships are stored in a separate table. The dataset also includes historical travel time data for 132 links, collected at a 2-minute interval, covering the entire day for four months from April to June 2017 and the entire month of July 2016. Additionally, travel time data for the same 132 links during three time([8:00 - 9:00), [15:00-16:00), [18:00- 19:00])periods of each day in July 2017 is provided. The dataset aims to predict the average travel time for each two-minute interval during March to June 2016 and July 2017, respectively, using the provided training data.

3 Related Work

An accurate traffic prediction in metropolitan circumstance is of great importance to the administration department. traffic flow forecast is the precondition for traffic management measures like traffic planning, route guidance, and traffic control. Traffic prediction also helps to prevent public or traffic accidents from happening (Zheng et al 2014) through noticing administrators in advance, and the emergency response plans can be deployed promptly.

4 China traffic part

4.1 Consideration

Through reviewing the information, we believe that the more critical aspects of this competition are:

1) The analysis of historical data revealed that the distribution of traffic flow is mainly cyclical in terms of weeks, so the modelling of the model needs to make good use of this cyclicality to construct features;

2) The competition specifically gives real-time travel times for the hour before the corresponding forecast period, which is a reflection of the actual situation on that day and contains relatively large amounts of information;

3) The 2min time granularity of the prediction results, this time granularity is relatively fine, how to reflect the overall homogeneity of the prediction period of one hour and how to reflect the connection and difference of each 2min time slice within the prediction hour, need to grasp well

4) The question mentions the fluctuation of traffic time, whether it can reflect the prediction when congestion occurs on the road.

4.2 Data processing

The data is first subjected to some processing including removal of missing values, normalisation, sorting.

Next, each element in a column in the data frame is split into a row to form a new data frame. Finally, the nodes and edges are transformed into a graph using and visualised. Then, we process the data into hourly average traffic flows and each data has a corresponding timestamp.Lastly, a graph between the roads is constructed, by traversing the paths and converting each road connection into an edge, represented by the edge-index array. At the same time the node-feature array is created and contains information about the width and length of each node.

4.3 Modeling

In the model part, we apply the Gated Graph Neural Network (GGNN). First of all, we define the Linkage Network to enrich the properties a graph of the road network can present. Linkage which is newly introduced can include and present the significant property called propagation pattern, which actually shows the internal mechanism of the traffic variation.

After that, GGNN is proposed to mine and learn this propagation pattern and make the prediction and synchronously. GGNN contains a propagation module to propagate the hidden states along the linkage network just as the traffic flow spreading along the road network. Considering that the propagation of traffic flow directly affects the variation of traffic, GGNN can easily generate the prediction results with the already learned patterns. Then we integrate the LSTM method for time-series forecasting. LSTM can capture the complex temporal dependencies of the traffic data, which is essential for accurate traffic prediction. Lastly, we use a linear layer to get the final prediction.

5 Applying XGboosting on Smart Traffic project

5.1 Consideration

In previous models, we attempted to use graph neural networks directly to model and solve the problem of intelligent transportation prediction.

5.2 Data Exploration

The traditional research steps of machine learning projects, we examined some properties of the dataset. In the given dataset on the attributes of traffic flow on links, we explored the distribution of the fourth column, which represents the average travel time in seconds, and also the distribution after applying a logarithmic transformation. We found that the distribution after applying the logarithmic transformation is approximately normal.

We wanted to observe the average traffic volume of all roads over time on a weekly basis. The graph below shows the changes in traffic volume over time on the same day of the week.

5.3 Outlier Handling

In real-world traffic, we often face extreme situations that are not caused by daily tides or road location and attributes, such as large accidents, traffic jams caused by accidents, temporary traffic controls, etc. We hope to remove these outliers to achieve data smoothing. By using the ”cast_log_outliers” method, we remove outliers greater than the 0.95 percentile value and less than the 0.05 percentile value, so that the model we build on this dataset has better robustness.

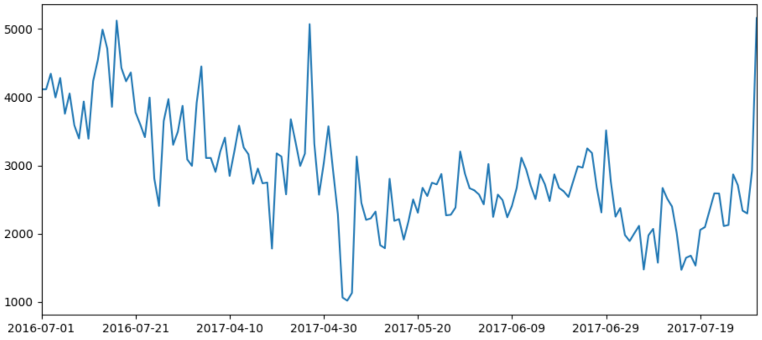

Fig. 1. the distribution of the missing values.

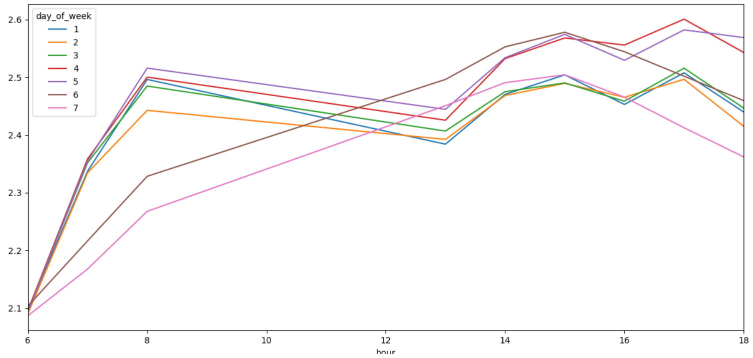

Fig. 2. the flow distribution with the time in days.

5.4 Missing Value Handling

why As mentioned earlier, the provided dataset for the competition is not complete, and missing values are prevalent. Traditional methods of filling missing values often involve manual processing, such as using the median or mean for interpolation.

Finding Missing Values We need to process the table to obtain all missing values. By constructing a new data frame containing all two-minute time intervals within a complete day, filling all other information with null values, and taking the Cartesian product with all link IDs, we can obtain a complete table of information filled with null values. Then, by merging it with the existing data frame, we can fill in the missing information with null values.

Implementation of Missing Value Handling Method We used spline interpolation to handle missing values in our study. In the implementation process, we filled in missing values for each road segment based on the trend of date and time. Specifically, we first grouped the data by time, and then further grouped it by date to calculate the average travel time for each date. We then filled in missing values with the daily average value or the linear regression fit of the time trend. Next, we grouped the data by time again and then further grouped it by minute to calculate the average travel time for each minute. We filled in missing values with the average value of each minute or the result of spline function fitting.

5.5 Feature Extraction

After missing value processing, we can begin training the model by extracting features. First, we need to extract the basic attribute features of the road segments linked in the network structure, such as length and width. Intuitively, the longer and narrower the road, the longer it may take for vehicles to pass through. However, there are slight discrepancies between intuition and reality. We visualized the travel time based on the length and width of the road, and found that there is a clear positive correlation between travel time and the length of the road, but for the width, wider roads do not necessarily result in a significant decrease in vehicle travel time, except for a noticeable improvement at 12 meters.

5.6 Feature Extraction and Model Training

After feature extraction, we can begin model training. First, we need to extract basic structural features such as length and width. Intuitively, longer and narrower roads may take more time for cars to pass through. However, there are some discrepancies between this assumption and reality. We visualized the relationship between travel time and road length/width, and found a clear positive correlation between travel time and road length, but no significant decrease in travel time with increasing road width, except for a noticeable improvement at 12 meters. We also extracted upstream/downstream and ID attributes of the network structure. Finally, we focused on constructing time features. In reality, traffic has a strong continuity, meaning that traffic tides gradually accumulate or dissipate, and the increase in travel time is a continuous and smooth process, making it difficult to have sudden drops in travel time. Therefore, in addition to specific date, holiday, weekend, and other attributes, we constructed five lag values to predict the current travel time.

5.7 Discussion

As the competition has already closed, we were unable to submit our results to obtain scores, nor could we compare our predictions with the complete dataset.

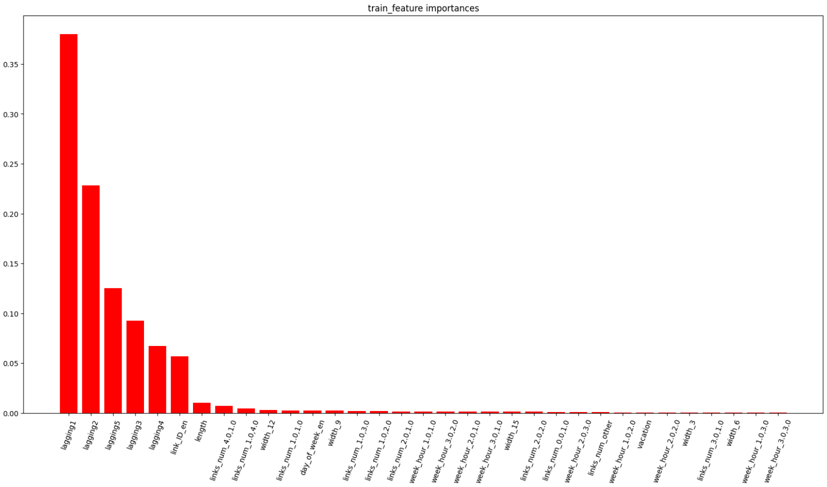

Fig. 3. the importances of all features.

However, the results achieved locally are still satisfactory, and we only need to consider the problem of overfitting. From this practice, we also discovered that the network structure is a vast topic, and many factors need to be considered when selecting a suitable approach. In the future, we hope to further explore the similarities and differences between traditional machine learning models and graph-based machine learning models for network structures.

6 Traffic Prediction in PeMS04

6.1 Traffic Prediction Introduction

The previous data set is too large, and need a more data processing to suit for different gnn models, but it is too time-consuming. So, we use a smaller and have been labeled data set that related to graph project. PeMS04 provides a unified database of traffic data collected by Caltrans on California's highways. The traffic data obtained since January 1, 2018, collected every 5 minutes, so the shape of the original traffic data after reading is (307, 16992, 3), in which the three-dimensional characteristics are flow, occupy, speed. The original adjacency matrix data is a distance.csv file, which contains the formats of from,to and distance. For convenience, the distance (corresponding edge weight on the figure) in this paper is set as 1 whenever nodes are connected.

6.2 Traffic Prediction Algorithms

We mainly used GAT, GCN, Chevnet, ASTGCN and ARIMA algorithms to get the predicted results. For GAT, GCN and Chevnet, MAE, MAPE and RMSE are three different loss functions and use the default learning rate in Adam. Finally, We got the result. For ASTGCN, We use L1 loss and also Adam to got the results. For ARIMA, it is a non-nn model for linear prediction, And We got the results with a very long time.

6.3 Traffic Prediction Evaluations & Summary

The GAT, GCN, Chevnet, algorithms in criterion MAE, MAPE and RMSE. The results and predictions see in below:

Table 1. The GAT, GCN, Chevne algorithms results.

Algorithms	MAE	MAPE	RMSE	Loss	LR
GCN	69.45	0.55%	96.75	MSE	0.001
GAT	59.68	0.59%	82.97	MSE	0.001
Chevnet	93.10	0.81%	128.47	MSE	0.001

Table 2. ASTGCN algorithms results.

Algorithms	validation batch	loss	Optimizer	Criterion
ASTGCN	1/107	275.67	Adam	L1 loss
ASTGCN	101/107	331.83	Adam	L1 loss

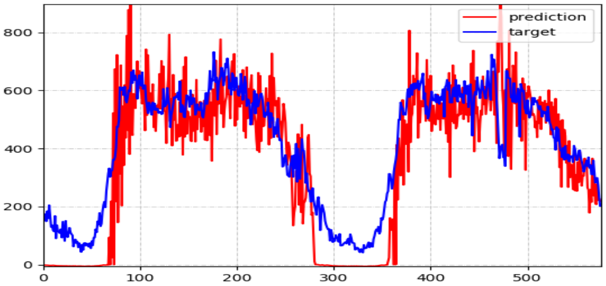

Fig. 4. Prediction in GCN

References

1. Fusco, Colombaroni, and Isaenko 2016 Fusco, G.; Colombaroni, C.; and Isaenko, N. 2016. Short-term speed predictions exploiting big data on large urban road networks.
2. Nguyen, Liu, and Chen 2017 Nguyen, H.; Liu, W.; and Chen, F. 2017. Discovering congestion propagation patterns in spatio-temporal traffic data. IEEE Transactions on Big Data 3(2):169–180.
3. Min and Wynter 2011 Min, W., and Wynter, L. 2011. Realtime road traffic prediction with spatio-temporal correlations. Transportation Research Part C 19(4):606–616.
4. Lippi, Bertini, and Frasconi 2013 Lippi, M.; Bertini, M.; and Frasconi, P . 2013. Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning. IEEE Transactions on Intelligent Transportation Systems 14(2):871–882.
5. Zheng et al 2014 Zheng, Y .; Capra, L.; Wolfson, O.; and Y ang, H. 2014. Urban computing: Concepts, methodologies, and applications. ACM Transactions on Intelligent Systems and Technology 5(3):38.