HIGHWAY TOLLGATES TRAFFIC FLOW PREDICTION USING MACHINE LEARNING ALGORITHMS

Dai Yirui, Dong Meirong, Lim Chong Seng Hermann

Institute of Systems Science, National University of Singapore, Singapore 119615

ABSTRACT

Along with the fast development of interconnected economy, modern-day traffic networks have become larger and more complex. Ineffective management of these traffic networks have multiple negative impacts on the individual as well as the national economy.

The study aims to use machine learning algorithms such as XGBoost and LightGBM to predict the conditions at traffic bottlenecks such as highway tollgates to reduce congestion, enhance road users' experience and improve infrastructure management.

There are two main issues focused in this study, the first is to forecast the average traveling time through the tollgates and the second is to estimate the traffic volume at the tollgates. The specific problem set and related information are provided in Knowledge discovery in databases (KDD) competition 2017[1]. Mean Absolute Percentage Error (MAPE) is used to evaluate accuracy of submission results during the competition. MAPE is also adopted and servers as the benchmark for performance measurement in this study. The provided data include overall road network topology, how each road interconnected with others, trajectories of travelling vehicles, time taken and volume of vehicle counts at tollgates, and parameters describing weather conditions at corresponding time window. Features like potential bottlenecks of each road, volume patterns and irregularities, travelling patterns of top frequent commuters are examined in this study. Contemporary research and work are also explored and some are referred to as baseline and guidance for solution implementation. The team managed to reach a MAPE = 0.242141 for speed prediction, and volume prediction MAPE = 0.137198 using XGBoost and MAPE = 0.137694 using LightGBM respectively.

***Index Terms*—** tollgates, XGBoost, LightGBM, KDD, MAPE

1. INTRODUCTION

Traffic management is an increasingly difficult task for almost every government and traffic authorities due to rapid globalisation and greater population mobility. Poor traffic management can adversely impact quality of life and economic productivity due to rising fuel consumption, cost of movement and number of accidents. Although the consequences are alarming, traffic congestion remains a prevalent issue in many countries because they have inadequate resources to address the problem. Therefore, minimising traffic problems with optimized costs while satisfying the growing demand for more efficient traffic networks should be the core objective of today's traffic management studies.

To achieve this goal, many have applied advanced technologies including machines learning[2] and deep learning[3] in their works. In this study, the team aims to use machine learning algorithms such as XGBoost and LightGBM to predict traffic flow volume pattern and travel time patterns would occur in the near future(e.g.next 2 hours) on routes to help traffic authorities and commuters to make informed travel decisions, to divert traffic and to address the core objective.

The study uses Knowledge discovery in databases (KDD) 2017 dataset for analysis. The dataset is provided by "Hanzhou Jiaotong Amap", the task is to predict highway tollgates traffic flow and the results are to be measured and ranked based on the "Mean Absolute Percentage Error (MAPE)" score.

The dataset is chosen because manual tollgates are usually the bottlenecks of traffic network and can easily cause traffic congestion and overwhelm traffic authorities and commuters. One small improvement made to the bottlenecks could easily be one giant leap for traffic conditions across the route, by Pareto principle.

2. PROBLEM OVERVIEW

The road network topology is a directed graph formed by a sequence of interconnected road links. There are 6 routes in the network. The bird view of the road network and routes between intersections and tollgates are shown in below figures.

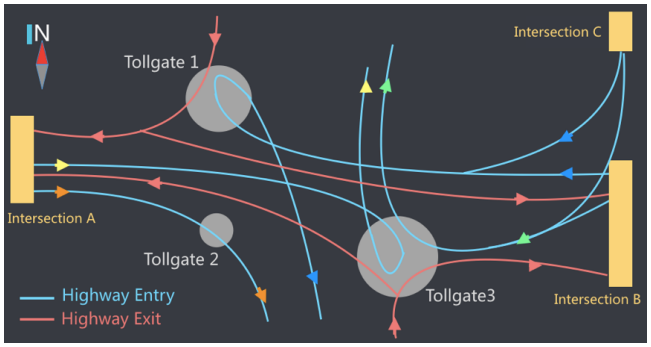


Fig. 1. Visualization of Road Network Topology

	intersection_id	tollgate_id	link_seq	link_count
0	A	2	110,123,107,108,120,117	6
1	A	3	110,123,107,108,119,114,118,122	8
2	B	1	105,100,111,103,116,101,121,106,113	9
3	B	3	105,100,111,103,122	5
4	C	1	115,102,109,104,112,111,103,116,101,121,106,113	12
5	C	3	115,102,109,104,112,111,103,122	8

Fig. 2. Details of roads interconnections

Problem 1 is to estimate for every 20-minute time window the average travel time of vehicles for a specific route:

- Routes from Intersection_id A to Tollgate_ids 2 3;
- Routes from Intersection_id B to Tollgate_ids 1 3;
- Routes from Intersection_id C to Tollgate_ids 1 3.

Problem 2 is to predict average tollgate traffic volume for every 20-minute time interval window where tollgates 1 and 3 has entry and exit traffic and 2 has only entry traffic.

There are 5 classes of data provided in the dataset which are links, routes, trajectories, volume and weather. In this study, data (.csv) files under /dataSets/training folder (Date:) are used as training data, and /dataSets_phase2 (Date:) are used for evaluation purposes. The reason is that in KDD competition 2017, the final data used to evaluate contestant performance was not made known.

Each link is asigned with a link_id. Important features like length, width and lanes are provided for calculation of speed and identification of potential bottlenecks.

Field	*Type*	*Description*
link_id	string	link id
length	float	length (meter)
width	float	length (meter)
lanes	int	number of lanes
in_top	string	incoming road link(s), separated by comma (as shown in Figure 3)
out_top	string	outgoing road link(s), separated by comma (as shown in Figure 3)
lane_width	float	lane width (meter)

Fig. 3. Links

Routes are set of definitions that connect starting point: intersection to ending point: tollgate with list of links in between.

Field	*Type*	*Description*
intersection_id	string	intersection ID
tollgate_id	string	tollgate ID
link_seq	string	a sequence of link IDs from the intersection to the tollgate separated by commas (shown in Figure 4)

Fig. 4. Routes

Trajectories are used to describe how each individual vehicle travel at a certain time period. The data is used for speed calculation.

Field	*Type*	*Description*
intersection_id	string	intersection ID
tollgate_id	string	tollgate ID
vehicle_id	string	vehicle ID
starting_time	datetime	time point when the vehicle enters the route
travel_seq	string	trajectory in the form of a sequence of link traces separated by ";", each trace consists of link id, enter time, and travel time in seconds, separated by "#"
travel_time	float	the total time (in seconds) that the vehicle takes to travel from the intersection to the tollgate

Fig. 5. Trajectories of Vehicles

Volumes data is count of individual vehicle at timestamp. Vehicle type, vehicle model, whether this vehicle uses "ETC" (Electronic Toll Collection) are captured in this dataset. Volumes data is aggregated by 20 minutes time window, and is the main source for volume predictions. It may also provide some insights for bottleneck detection.

Field	*Type*	*Description*
time	datetime	the time when a vehicle passes the tollgate
tollgate_id	string	ID of the tollgate
direction	string	0: entry, 1: exit
vehicle_model	int	this number ranges from 0 to 7, which indicates the capacity of the vehicle (bigger the higher)
has_etc	string	does the vehicle use ETC (Electronic Toll Collection) device? 0: No, 1: Yes
vehicle_type	string	vehicle type: 0-passenger vehicle, 1-cargo vehicle

Fig. 6. Vehicle Counts at Tollgates

Weather data is captured by various types of sensors. It is used to examine the relationship between weather conditions and traffic operation.

Field	*Type*	*Description*
date	date	date
hour	int	hour
pressure	float	air pressure (hPa: Hundred Pa)
sea_pressure	float	sea level pressure (hPa: Hundred Pa)
wind_direction	float	wind direction (°)
wind_speed	float	wind speed (m/s)
temperature	float	temperature (℃)
rel_humidity	float	relative humidity
precipitation	float	precipitation (mm)

Fig. 7. Weather Conditions

Mean Absolute Percentage Error (MAPE) is used to evaluate the accuracy of both task 1 and task 2.

Task 1: Prediction of Speed

$$MAPE = \frac{1}{R}\sum_{r=1}^{R}(\frac{1}{T}\sum_{t=1}^{T}|\frac{d_{rt} - p_{rt}}{d_{rt}}|) \quad (1)$$

Task 2: Prediction of Volume

$$MAPE = \frac{1}{C}\sum_{c=1}^{C}(\frac{1}{T}\sum_{t=1}^{T}|\frac{f_{ct} - p_{ct}}{f_{ct}}|) \quad (2)$$

- d: actual average travel time for route r at time t
- p: predicted average travel time for route r at time t
- f: actual traffic volume for a specific tollgate-direction pair c at time t
- p: predicted traffic volume for a specific tollgate-direction pair c at time t
- R: Number of Routes
- C: Number of tollgate-direction pairs
- T: Number of To-Predicted Time Windows

3. RELATED WORK

The existing works on KDD 2017 have proposed many solutions for highway tollgates traffic prediction. Some of which are XGboost, Radial Basis Function (RBF) nets and deep neural networks. The most favourable model is XGboost. Therefore, the team uses tuned XGboost models (Chen, 2019)[4] as the main tool to generalize the regression model in this study.

Remarkably, XGboost has received tremendous attention in recent years (Morde Setty, 2019)[5] because of its promising speed and performance (Brownlee, 2016)[6]. The top few winning teams of KDD cup 2017 competition have also recommended this learning algorithm for their predictions.

An alternative approach is to use a set of continuous distributions to approximate the regression via RBF. By using RBF, the idea is to combine (or sum) a set (kernels) of curves (Gaussian distributions) with various control parameters (std mean) to generalize the regression model (Deshpande, 2017)[7].

More complex approaches that can stack and vote from different model (LASSO, GBDT, ADABoost and RandomForest etc) layers and deep neural networks such as(CNN-LSTM-Attn) and Temporal-Spatial-LSTM (TSLSTM) are also examined. However, details including how the models are implemented and their parameters are not provided by the author. This complex approach managed to reach a MAPE value around 0.104 for volume prediction.

In the end, due to time constraints, the team decided to follow the proposal submitted by the 1st place team[8] in KDD cup 2017 competition to implement XGboost and LightGBM.

4. PROPOSED APPROACH

With reference from KDD cup 2017 winning team's proposal, the team has proposed following step by step approach:

Average Travel Time Prediction

Phase	**Action taken**	**Action Category**
Study-1	Aggregate raw trajectories data based on 20-mins time window	Coding
Study-2	Plot volume of all time window at a particular day (1st day)	Exploring
Feature-1	Process and select feature from link and route tables for data training and prediction (link_counts, route_length, double_in_link_count, double_out_link_count, min_width)	Exploring
Feature-2	Process the features from trajectories table and clean outliers (add time lag for 2 hours, because based on other research works, the average traveling time from previous 2 hours have more significant impacts on the data)	Exploring
Feature-3	Process and understand the data from weather table	Exploring
Feature-4	Combine all the tables	Exploring
Base-1	Implement baseline approach	Coding
Improve-1	Test the baseline with different features and parameters to improve accuracy	Experimenting

Average Volume Prediction

Phase	Action taken	Action Category
Study-1	Aggregate raw volume data based on time window (20 mins)	Coding
Study-2	Plot volume of all time window at a particular day (1st day)	Exploring
Study-3	Look for irregularities and try to explain the reason. Exclude irregularities if not helpful for prediction	Exploring
Study-4	Plot volume by day for the whole training period	Exploring
Study-5	Look for irregularities and try to explain the reason. Exclude irregularities if not helpful for prediction	Exploring
Study-5	Look for irregularities and try to explain the reason. Exclude irregularities if not helpful for prediction	Exploring
Study-5	Look for irregularities and try to explain the reason. Exclude irregularities if not helpful for prediction	Exploring
Study-6	Plot top (4) frequent 'travellers' and see if there are similar travelling patterns (Explain for peak and trough)	Exploring
Base-1	Implement baseline approach	Coding
Base-2	Experiment with denoised data (excluded holiday periods), calculate baseline MAPE as benchmark	Experimenting
Base-3	Parse and join weather data with volume data	Coding
Base-4	Further experiment with weather data, check if MAPE improve	Experimenting
Improve-1	Aggregate raw volume data based on time window (20 mins) but include 'has$_etc'as feature$	Coding
Improve-2	Implement function to calculate mean volume values of each time window for past n (5) days	Coding
Improve-3	Exclude irregular high volumes at midnights	Experimenting
Improve-4	Experiment with volume vs. log(volume)	Experimenting
Improve-5	Experiment with different evaluation metrics: 'mae'	Experimenting

5. EXPERIMENTAL RESULTS

For task 1, the first step is to extract useful information from the given dataset. The team has used these features from the dataset for task 1.

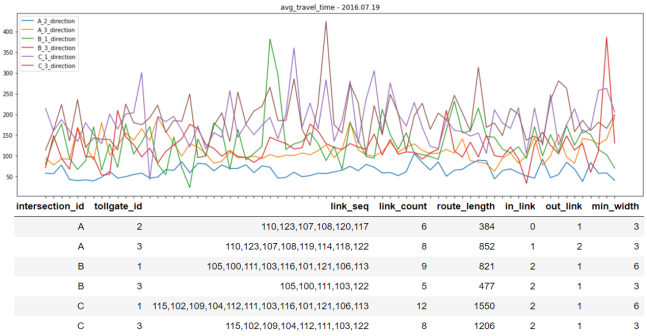
link	link_id, length, width, lanes, in_top, out_top, lane_width
route	intersection_id , tollgate_id, link_seq
trajectories	intersection_id, tollgate_id, vehicle_id, starting_time, travel_seq, travel_time
weather	Date, hour, pressure, sea_pressure, wind_direction, wind_speed, temperature, rel_humidity,precipitation lane_width

Then, the team has processed and extracted these features.

route and link	link_count, route_length, in_link, out_link,min_width
trajectories	lag1, lag2, lag3, lag4, lag5, lag6, lag7, date, month, day, weekday, hour, minute, holiday, rush_hour

The more important features extracted are "route_length", "lags" and "hour" (in trajectories table).

A longer route will usually result in longer travel time, as shown in the diagram below, the travel time in longer routes C-1 and C-2 is generally is higher.



intersection_id	tollgate_id	link_seq	link_count	route_length	in_link	out_link	min_width
A	2	110,123,107,108,120,117	6	384	0	1	3
A	3	110,123,107,108,119,114,118,122	8	852	1	2	3
B	1	105,100,111,103,116,101,121,106,113	9	821	2	1	6
B	3	105,100,111,103,122	5	477	2	1	3
C	1	115,102,109,104,112,111,103,116,101,121,106,113	12	1550	2	1	6
C	3	115,102,109,104,112,111,103,122	8	1206	2	1	3

Fig. 8. Average Travel Time by Routes

The lags carry travel time information from the previous 6 time windows (2 hours) which can significantly impact model performance as proved by other studies.The travel_hour (e.g. 1am,2am) is also able to improve the performance (improve MAPE by almost 0.1 as shown by the experiment results in this study).

The second step is to process outliers and normalize the data. Because the data given is limited (about a month of data), the team decided to use average values to replace the outliers (e.g. Holidays) instead of eliminating them. After

normalisation, the maximum travel time has reduced from 6711.11 to 639.24.

	tollgate_id	vehicle_id	travel_time
count	109244.000000	1.092440e+05	109244.000000
mean	2.274496	1.040367e+06	106.447486
std	0.700266	2.717108e+04	71.761686
min	1.000000	1.000004e+06	9.260000
25%	2.000000	1.014494e+06	60.230000
50%	2.000000	1.039902e+06	93.575000
75%	3.000000	1.063742e+06	134.010000
max	3.000000	1.088979e+06	6711.110000

	tollgate_id	vehicle_id	travel_time
count	109244.000000	1.092440e+05	109244.000000
mean	2.274496	1.040367e+06	105.506049
std	0.700266	2.717108e+04	61.917086
min	1.000000	1.000004e+06	9.260000
25%	2.000000	1.014494e+06	60.200000
50%	2.000000	1.039902e+06	93.535000
75%	3.000000	1.063742e+06	133.830000
max	3.000000	1.088979e+06	639.240000

Fig. 9. Travel Time Before & After

The data pattern after normalisation is as below.

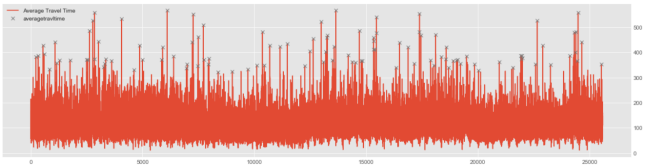


Fig. 10. Overall Average Travel Time

There are still some peaks due to holiday traffic. However,these peaks do not have significant influence on the performance because accuracy did not improve after the peaks are removed.

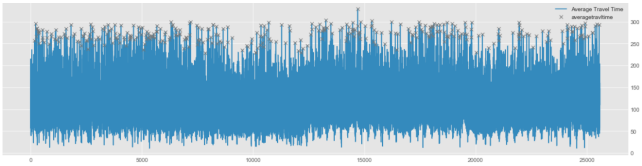


Fig. 11. Overall Average Travel Time

The team has also tried log transformation and normalisation on the dataset. There is not any improvement in accuracy. It could be because there is not any significant outliers in the dataset after data processing earlier.

The last step is to train the dataset using the chosen model XGBoost. The model was chosen because it has produced good accuracy in the experiments done by other researchers. The best MAPE result achieved by the team is 0.23756.

For task 2, the raw data is presented in a way that timestamp is taken when a vehicle passed through a particular toll gate at a certain time:

	time	tollgate_id	direction	vehicle_model	has_etc	vehicle_type
0	2016-09-19 23:09:25	2	0	1	0	NaN
1	2016-09-19 23:11:53	2	0	1	0	NaN
2	2016-09-19 23:13:54	2	0	1	0	NaN
3	2016-09-19 23:17:48	1	0	1	1	NaN
4	2016-09-19 23:16:07	2	0	1	0	NaN

Fig. 12. Volume Raw Data

To estimate the volume of target time window, firstly, individual vehicle count should be summed up to every 20 minutes time interval. The team has used our reference code to perform the aggregation.

	tollgate_id	time_window	direction	volume
0	3	[2016-09-19 00:00:00,2016-09-19 00:20:00)	0	17
1	3	[2016-09-19 00:00:00,2016-09-19 00:20:00)	1	181
2	1	[2016-09-19 00:00:00,2016-09-19 00:20:00)	0	13
3	1	[2016-09-19 00:00:00,2016-09-19 00:20:00)	1	140
4	2	[2016-09-19 00:00:00,2016-09-19 00:20:00)	0	2

Fig. 13. Volume at 20 mins Time Window

5.1. *A dive into the data:*

The volume pattern of each tollgate direction pair on the first day is shown in below chart. Notice that tollgate 2 has only 1 direction thus leave us total 5 tollgate direction pairs. From the chart, it is clear to see that there are some peaks at rush hour periods.

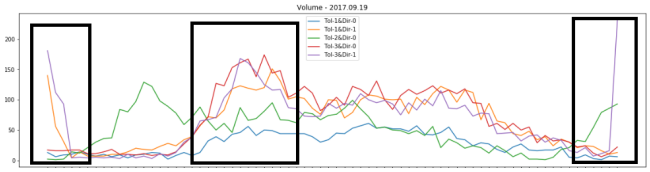


Fig. 14. Volume in 20 Mins Window, Sep 19th

The other observation is that there is high volume for certain tollgates at midnight. However, by comparing with the overall data at all times, it is difficult to conclude meaningful features for those passing at midnight. In this work, the team decided to exclude the data from midnight while training since they are not relevant to the rush hour data but add much noise.

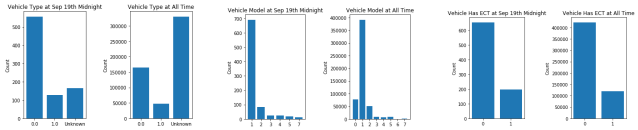


Fig. 15. Features of Vehicle at Midnight vs Overall

To study the commuting behavior of individual vehicle, the team has selected the top 4 vehicles by counting the number of appearances made at any tollgate. It is noticeable that each of these vehicles has its own unique travelling patterns. It is not feasible to rationalize peak periods by only looking at these patterns.

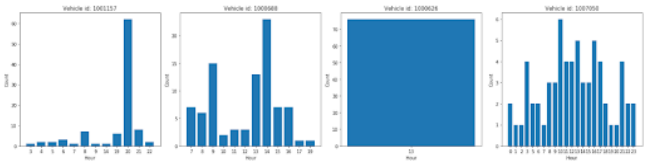


Fig. 16. Appearance of Top 4 Vehicles in by hour

The volume of all tollgate direction pairs on a daily basis is shown in below chart. It is clear to see that the volume pattern between Sep 30th and Oct 7th is very different from the rest. This behavior is aligned with the fact that this time period is a long holiday in China, and thus many people will not travel in the same way as in working days. In this case, the target time period that requires our prediction is not in holiday period. Therefore, in order to give a more precise prediction, it is important to only look at those non-holiday data.

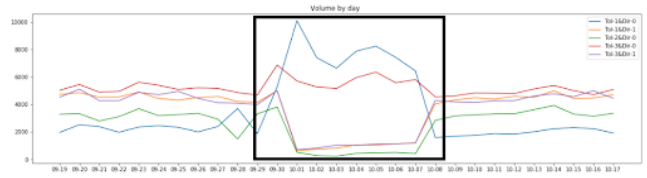


Fig. 17. Overall Volume On Daily Basis

5.2. *Incorporating with weather data as features:*

Weather conditions can significantly influence traffic operations (Hani etc, 2009)[9]. To better predict traffic flow, the team has done some experiments incorporating with weather data. With reference from github code, the team has done some data preprocessing to join weather data with volume data:

- Align time window: weather - 3 hours vs. volume - 20 mins
- Convert 'wind_direction' to numeric (360 degrees)
- Fill in 'NaN' fields with overall mean of corresponding feature

Below table shows the comparison between MAPE with and without weather data:

	Overall MAPE	**Rush Hours MAPE**
Weather (N)	0.268922	0.191247
Weather (Y)	0.429293	**0.187615**

5.3. *Further Denoising Data Adding 'has_etc' feature:*

One of the findings from previous studies is that irregular high volumes happen often at midnight. These irregularities are hard to explain and not helpful for prediction at target time period. In addition, as proposed by the winning team, features including 'has_etc' may help to improve prediction accuracy. This makes sense because vehicles that have ETC need not wait for fee collection at the tollgate. Below table shows the comparison between MAPE before and after:

	Overall MAPE	**Rush Hours MAPE**
MidN(Y) & ETC(N)	0.429293	0.187615
MidN(N) & ETC(Y)	0.304405	**0.156712**

5.4. *Adding in Statistical Features:*

As proposed by KDD cup 2017 winning team, statistical features (mean, min, max etc) of corresponding time window of last n days, overall statistics of last n days, and rush hours statistics may help improve prediction accuracy. To experiment the impacts of statistical feature, the team has calculated the mean of corresponding time window of last 5 days for all training/testing instances. The calculation process can consume significant time and resources, so caution should be taken when input large data. Below table shows the comparison between MAPE before and after adding the statistical feature:

	Overall MAPE	**Rush Hours MAPE**
Stats (N)	0.304405	0.156712
Stats (Y)	0.202389	**0.137198**

Visualization of final predicted volume vs actual volume in target time periods:

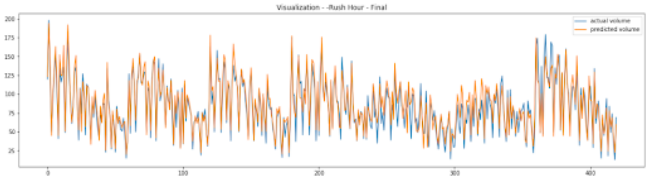


Fig. 18. Visualization of Actual vs. Predicted at Target Time Window

6. DISCUSSIONS

Due to time constraints, there are many other proposed approaches that could potentially improve the accuracy of the predictions are not included in this study. The team proposed

the use of methods such as data augmentation, moving average for time series, step by step modelling (use of latest available data to update model), ensembled and weighted systems (e.g. XGBoost, deep neural networks, and LightGBM) for performance testing in future work.

Also, some approaches like the use of logarithm for labelling, and M-estimator as objective function are explored but not applied due to reduction of accuracy and lack of implementation details. The team recommend any derivative works based this study to try the approaches for performance tuning.

7. CONCLUSIONS

The study discussed the problem of travel time and volume predictions at highway tollgates to help commuters to manage their journey and traffic authorities to plan their resources.

By integrating above findings and techniques, MAPE has improved from 0.191247 to 0.137198 for the target time windows for volume prediction and achieved 0.23756 for travel time prediction . The accuracy is quite remarkable as it can be ranked in the top 10 for volume prediction in KDD cup 2017.

8. REFERENCES

[1] Alibaba Cloud Tianchi, "Tianchi Competition - KDD CUP 2017," https://tianchi.aliyun.com/competition/entrance/231597/information, [Online; accessed October 30, 2019].

[2] Jaros law Rzeszotko and Sinh Hoa Nguyen, *Machine Learning for Traffic Prediction*, 2011.

[3] Kaishun Wu Zhidan Liu, Zhenjiang Li and Mo Li, *Urban Traffic Prediction from Mobility Data Using Deep Learning*, 2018.

[4] Zeyu Chen, "KDD CUP 2017 code," https://github.com/chenzeyuczy/KDD2017, [Online; accessed October 30, 2019].

[5] Venkat Anurag Setty Vishal Morde, "XGBoost Algorithm: Long May She Reign!," https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d, [Online; accessed October 30, 2019].

[6] Jason Brownlee, "A Gentle Introduction to XGBoost for Applied Machine Learning," https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/, [Online; accessed October 30, 2019].

[7] Mohit Deshpande, "Using Neural Networks for Regression: Radial Basis Function Networks," https://pythonmachinelearning.pro/using-neural-networks-for-regression-radial-basis-function-networks/, [Online; accessed October 30, 2019].

[8] Huan Chen Ke Hu, Pan Huang and Peng Yan, *KDD CUP 2017 Travel Time Prediction Predicting Travel Time – The Winning Solution of KDD CUP 2017*, 2017.

[9] Jiwon Kim Hani S. Mahmassani, Jing Dong, Roger B. Chen, and Byungkyu (Brian) Park, *Incorporating Weather Impacts in Traffic Estimation and Prediction Systems*, 2009.