Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 00 (2018) 000–000

Transportation Research Procedia

www.elsevier.com/locate/procedia

International Symposium of Transport Simulation (ISTS'18) and the International Workshop on Traffic Data Collection and its Standardization (IWTDCS'18)

Short-term prediction for bike-sharing service using machine learning

Bo Wang[a], Inhi Kim[a]*

[a] *Institute of Transport Studies, Department of Civil Engineering, Monash University, Clayton, VIC 3800, Australia*

Abstract

The bike-sharing service has brought many conveniences to citizens and served as an effective supplement to the mass transit system. For docked bike-sharing service, each docking station has the designated location to store bikes and the station could be empty or saturated in different times. Bike-sharing operators generally redistribute bikes between stations by driving trucks according to their experiences which might lead unnecessary human resources It is ineffective for the operators and inconvenient for users to access this service. Therefore, predicting an accurate number of available bikes in the stations is important for both the operators and users. This paper mainly focuses on the short-term forecasting for docking station usage in a case of Suzhou, China. Two latest and highly efficient models here, LSTM and GRU, are adopted to predict the short-term available number of bikes in docking stations with one-month historical data. Random Forest is used to compare as a benchmark. The results show that both RNNs (LSTM and GRU) and Random Forest able to achieve good performance with acceptable error and comparative accuracies. Random forest is more advantageous in terms of training time while LSTM with complex structures can predict better for the long term. The maximum difference between the real data and the predicted value is only 1 or 2 bikes, which supports the developed models are practically ready to use.

Keywords: bike-sharing; short-term forecasting; RNN (recurrent neural networks); random forest;

* Corresponding author. Tel.: +61 3 9905 4240.
E-mail address: inhi.kim@monash.edu

2352-1465 © 2018 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)
"Peer-review under responsibility of the scientific committee of the International Symposium of Transport Simulation (ISTS'18) and the International Workshop on Traffic Data Collection and its Standardization (IWTDCS'18)"

1. Introduction

The rapid development of mobile and internet technology creates many opportunities for the shared economy. Bike-sharing has recently become one of the most popular forms of the shared economy. The bike-sharing service is widely accepted and enjoyed in many major cities all around the world. These bikes have brought much convenience for the short trips, reduced greenhouse gas emission, and encouraged bike users to embrace a healthy way to exercise. The sharing service offers in two different ways; one is the docked bike-sharing and the other is dockless bike-sharing. While the dockless bike-sharing program has brought many issues such as abandoning bikes, blocking the pedestrian space, and poor management, the docked bike-sharing program has become more popular.

However, it is not easy to manage these bikes in an efficient way. Since the capacity of docking stations limit the number of bikes for borrowing and returning, bike stations can be empty or saturated at a certain period. In order to address this issue, bikes are manually rebalanced by trucks with an arbitrary timetable (for example, twice a day at about 8:00 am and 23:00 pm in the city like Suzhou, China). Due to the unbalanced distribution in the number of bikes at docking stations, these resources cannot be fully utilized. It needs a better way of managing the bikes, therefore, this paper aims to utilize advanced prediction models for better bike-sharing management.

Short-term traffic forecasting is an important research topic in ITS, which can be applied to predict traffic indicators such as traffic flow, delay, speed, travel time, etc. The methodologies in this area can be categorized into three types: statistics, non-linear theories and machine learning (Gang et al., 2016, Vlahogianni et al., 2004). Wang (2016) compared the regional bike rental demand prediction results by using several models in machine learning and found that NN-based and tree-based models can reach most high prediction accuracy.

The increasing number of research outcomes in recent years show that machine learning leads the state-of-the-art results for short-term forecasting problems, and they also keen to more adaptiveness with data fusion problem (Zhang et al., 2017, Almannaa et al., 2018). In the past decades, many studies have tried conducting bike related forecasting using different types of prediction models. The study of Singhvi et al. (2015) presents a log-log regression model to predict the bike usage pattern of morning peak hours in New York City. Taxi usage, weather, and spatial factors were also considered to improve the accuracy.

Gradient Boosting Regression Tree (GBRT) was adopted to predict the total number of the bike usage and a multi-similarity-based inference model was applied to predict the usage of each station cluster, which pre-produced by a bipartite clustering algorithm Li et al. (2015). DeepST (a multi-layer CNN model) was developed to predict the short-term NYC bike usage in grids of the city Zhang et al. (2016). The grids like pixels in a photo as input were trained from different time periods with CNN model and external factors. Additionally, other studies have proposed mathematic models for demand and rental analysis or planning of bike systems (Xu et al., 2012, Wang, 2016). Since the previous studies in general make macroscopical predictions of the bike-sharing systems, the forecasting for the station-level needs to be investigated. Therefore, in this study, and neural networks and tree-based models are tested, and their results are compared for forecasting station-level availability of bike-sharing.

2. Methodology

2.1. Data collection

The docking bike-sharing system in Suzhou, China is chosen as a study case. This system is also known as Suzhou Youon Public Bicycle Systems offering over 46,000 bikes with 2,000 docking stations across the city. Different novel short-term traffic forecasting models were developed on top of the Suzhou bike-sharing dataset and the features of these models were discussed.

2.2. Data Description

The number of available bikes for all stations are collected and stored in the server in every minute. Each record contains station ID, the number of available bikes, and the timestamp. We randomly select 3 stations; station 628, 635 and 648 for this forecasting research. Each station has a maximum of 40 docking spots. Borrowing or returning bikes changes the available bikes from these stations. If the number is close to zero, it means nearly no bikes in the station for borrowing. On the contrary, the larger the number is, the more available bikes are in the station, but the less space for returning bikes.

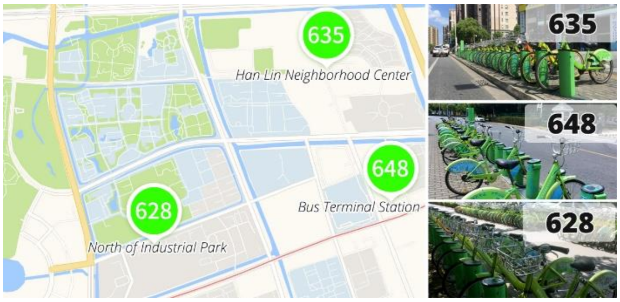

Fig. 2. Locations of the stations.

The locations are shown in Fig. 2. There are many commercial and residential areas, offices, and university campuses around the bike stations. The activities generated from these areas lead to a high potential bike usage. The main users of the bikes could be students, residents and company staff. Also, there is a metro line (the red line in Fig. 2) below the station 628 and 648), which deduce metro commuters to take up a large part of the total bike usages.

The dataset starting from 9th June and ending on 11th July, 2017. As a result, there are 45,950 observation data for each station. The raw data is split into a preset time interval (5 or 10 minutes) with different predicted requirements. For instance, each time-step will take the average value of 5 continuous 1-minute raw data if the time interval is 5 minutes. Under this condition, the whole data would contain 9,190 records with 5-minute interval and 4595 records with 10-minute interval. For each subset, the paper will always use 95% of all the records as training set and the remaining 5% for the testing set. In both two sets, the inputs length changes with different experiments and the output would always be one time interval.

2.3. Models

A series of fixed-length time sequences is selected as basic training and testing data. As shown in Fig. 3, t stands for current time-step and the numbers in the boxes means the observed values of available bikes. The values of continuous several time-steps in time sequence are arranged as input dataset. The next time-step $t+1$ will be predicted as output through the models.

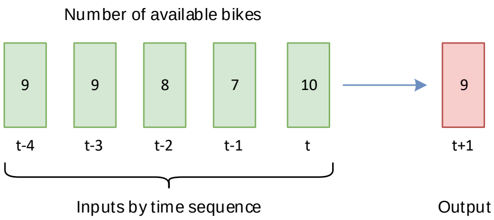

Fig. 3. Graphical Concepts of Inputs and Output.

(1) *Random Forest.*

Random forest (Breiman, 2001) was first introduced by Breiman in 2001 and it is a combination of multiple classifications and regression tree. The main idea of this algorithm is using bagging method to gather number of multiple weak learning units and then calculate the ensemble result. Compared with the traditional bagging method (Breiman, 1996), the random forest adds some random progress during the process of training. Random forest is easy to apply and the requirement of computational resources is relatively low.

(2) LSTM (Long short-term memory neural networks)

LSTM is a variation of recurrent neural networks for the time series prediction (Hochreiter and Schmidhuber, 1997). As shown in the Fig. 4. (a), the LSTM cell can hold and update a state during the training process. Thus, the model makes a prediction with the previous learning experience. The mathematical expressions can be denoted as

$$f_t = \left(W_f[h_{t-1}, x_t] + b_t\right) \tag{1}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{2}$$

$$\widetilde{C_t} = \tanh(W_C \times [h_{t-1}, x_t] + b_C) \tag{3}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{C_t} \tag{4}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

where t is the current step, x is the input, o is the output, W is the weight matrix, b is the bias. f_t, i_t, C_t and h_t are intermediate variables, which decide to remember or forget the input data (Olah, 2015).

Two LSTM layers are used in this paper and the output layer would make final regression results. Fig. 4. (b) shows the basic structure of the model as well as the running steps of the LSTM cells. When the continuous input values flow in the LSTM cell, it would unfold and handle these values by a sequence as Fig. 4. (c). After the last one is finished, the cell would make an output result for the next layer.

*(3) GRU (*Gated Recurrent Unit*)*

GRU was introduced in 2014 (Cho et al., 2014). It is an improved recurrent neural network based on LSTM. It merges the input part and forgetting part together so the number of the gates from 4 becomes 3. As a result, GRU saves more computational resources than LSTM with similar performance. To compare the difference between LSTM and GRU, we use same network structure as LSTM in figure 3 (b). The main expressions can be indicated by following formulas (Olah, 2015).

$$z_t = \sigma(W_z[h_{t-1}, x_t]) \tag{7}$$

$$r_t = \sigma(W_r[h_{t-1}, x_t]) \tag{8}$$

$$\widetilde{h_t} = \tanh(W[r_t \odot h_{t-1}, x_t]) \tag{9}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h_t} \tag{10}$$

(4) Evaluation

This article selects mean square error (MSE), mean absolute error (MAE) and Mean absolute percentage error (MAPE) to measure the performance of the different models. They are denoted by

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2 \tag{11}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y_i}| \tag{12}$$

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \widehat{y_i}|}{y_i} \tag{13}$$

where N is the number of testing sample, y is the real data and $\hat{y}$ is the corresponding prediction. In this article, the mean absolute error means the average difference between real available bikes and the predictive one in a station.

3. Results

3.1. Training Time Comparison

The LSTM and GRU run on top of Keras and Tensorflow with GPUs. We use Scikit-learn and CPU to build and test random forest. Neural networks need a number of training epochs to adjust the weights in the model. We select

25 as the number of epochs because the performance becomes stable after 25 times. Each estimator means one sub-tree in the random forest, and 1000 estimators would reach a stable outcome for this forecasting problem.

Table 2. Average training times of the models.

Time Interval	**Sequence Length**	**Training Time (s) - 25 epochs**		**Training Time (s) - 1000 estimators**
		LSTM	**GRU**	**RF**
1	5	45.83	37.59	5.31
	10	78.15	61.58	13.62
	20	143.42	109.05	36.78
	30	210.45	160.81	64.77
5	5	14.50	12.08	2.56
	10	20.81	16.90	4.35
	20	34.47	26.22	8.75
	30	47.81	37.37	13.71
10	5	10.82	6.46	1.87
	10	13.84	7.45	2.93
	20	20.97	9.68	5.32
	30	28.19	11.73	7.69

Table 2 shows the training time comparison with different scenarios. The longer time interval or sequence length, the longer training time for all models. GRU requires less training time than LSTM because it holds only 3 gates in its structure, which means that fewer computation effort is required. For the random forest, the number of estimators means the number of decision trees. It takes more training time when increasing the sample size. Although it runs with CPU, the real training time is much shorter than RNNs, which means it can be practical to implement in the real project.

3.2. Predictions

After each training step, the corresponding testing sets are estimated by the model. As an example, we randomly select only 50 testing samples from each testing group for visualization. Fig. 5. shows the part of predicted results with 10-mins time interval by using three models. The main findings are listed as follows. As a whole, all models show good predictions with some variations. The detailed observations from the results are following;

- LSTM, GRU and Random Forest follows the trends very well. The difference between the real data and predicted data is less than 1 or 2 bikes. It is good enough to help develop the bike-sharing management based on these predictions;
- The accuracies between the three predictions are very comparative;
- LSTM and GRU get similar trends in most of the cases because they have similar model structures;
- When the time interval is short, random forest gets a better performance;
- By increasing the sequence length, the fluctuation of three predictions is decreased.

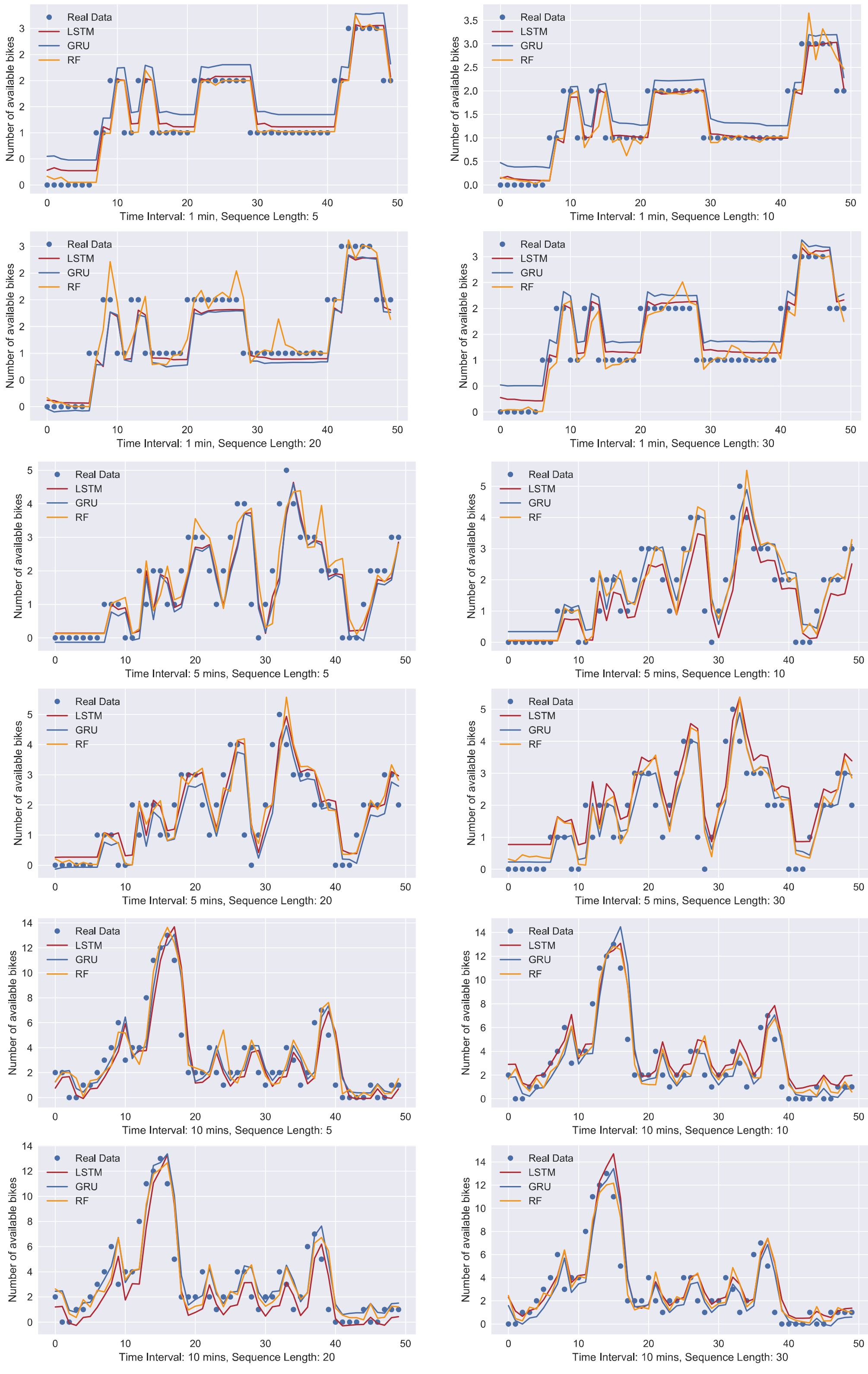

Fig. 5. Predicted available bikes of testing stations.

To describe the relationship between model performance and time interval and sequence length, we plot nine 3D graphs to show the change of evaluation metrics among three stations.

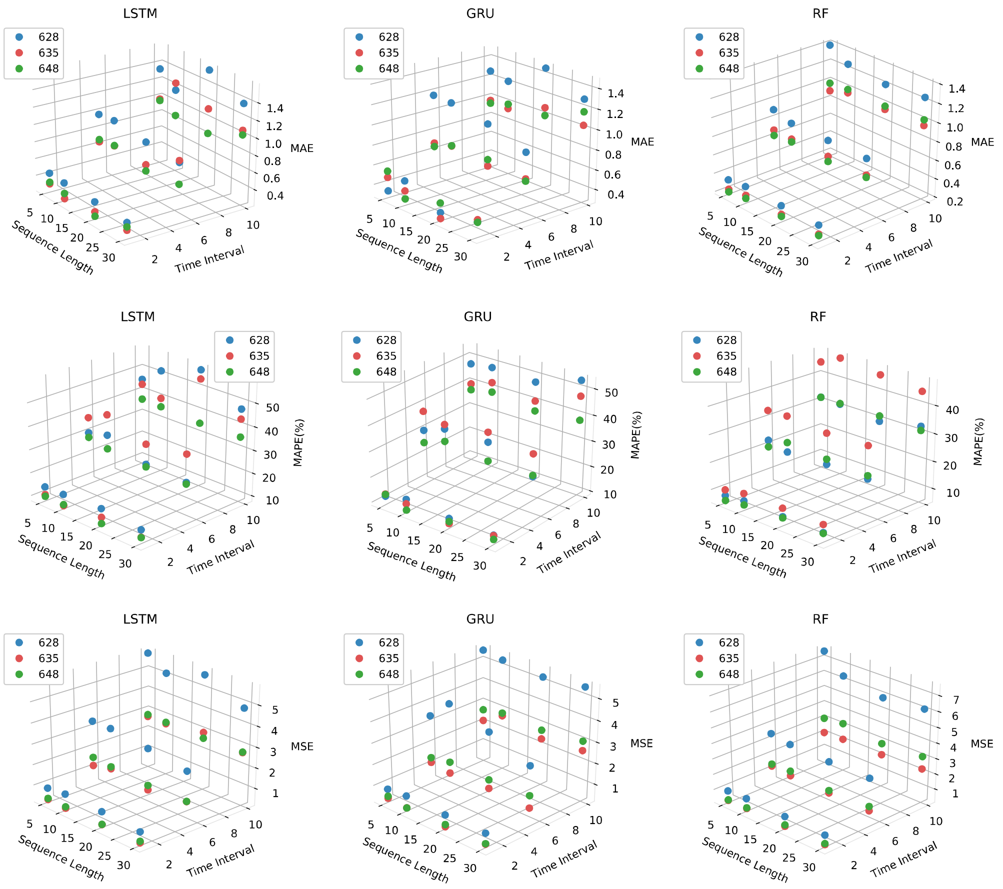

Fig. 6. Performance of models by time interval and sequence length.

The time interval only has three different levels: 1min, 5mins, and 10mins. The reason we choose these 3-time intervals is that in most case, people can access a station within 10 mins in Suzhou. Additionally, metro passengers can check the bike availability on mobile devices in the future when they are about to get off the metro. The sequence length has four different levels: 5, 10, 20, and 30. As a result, the lines in the graphs are not smooth but broken lines. The main features of these graphs are listed as follows.

- The MSE of RF is very good when the time interval is short but with the increase of the time interval, RF gets worse performance than other two. The memory unit in the RNN may help with long interval sequence;
- Generally, these three models get quite similar results. However, the blue line always higher than others in the graph of GRU and Random Forest, which means station 628 gets worse results than those two models. Since 628 station has the larger usage amount, the variation may be higher than others;
- All lines get higher with the increase of the time interval. There are two potential reasons may cause this problem. One is that forecasting long time interval is harder. The other one is that the sample size is dramatically decreased when expanding the interval so that the accuracy is affected;
- Different sequence lengths do not change much performance but sequence lengths affect the results gently and the longer one leads to a better performance in most cases.

4. Conclusion and discussion

Short-term traffic forecast plays a significant role in operating traffic management in ITS. An accurate forecast can act as guidance for commuters to better arrange their departure time, commuting modes, and commuting routes. Also, it is beneficial to the shared-bike service provider in terms of arranging bike delivery schedule and bike distribution.

This paper focuses on short-term traffic forecast on the available number of bikes of the shared-bike stations using machine learning techniques. Three short-term traffic forecast methods are proposed to make the prediction of available bikes, including LSTM, GRU, and RF. MSE, MAE, and MAPE are chosen as criteria to evaluate the performance of the three models. According to the predicted results, the three models all work well in the short-term forecast of the number of available bikes. Firstly, random forest works marginally better than others when the time interval is short. Secondly, the results from LSTM and GRU are quite similar on predicted behaviors but GRU has more accurate results and faster training time than LSTM. Finally, due to the short training time and no complicated hyper-parameter setting, the random forest needs fewer computation resources and it is easier for training.

Since the output data (predicted results) in this paper only has one time-step, the multiple time steps need to be conducted. This paper does not include other factors that can influence the bike usages such as weather (temperature, air quality and dry/wet), day of the week, time of the day, and special events (school holiday, sports/entertainment events, and road construction). Since research on RNN has developed exponentially in last few years, it is worthwhile implementing several variations on the basic RNN structure. Furthermore, the optimal model structures, sequence length, the time interval for better prediction are required for further research.

Acknowledgements

This work was supported by Jiangsu Industrial Technology Research Institute.

References

ALMANNAA, M. H., ELHENAWY, M. & RAKHA, H. A. 2018. Predicting Bike Availability in Bikesharing Systems Using Dynamic Linear Models.

BREIMAN, L. 1996. Bagging predictors. *Machine learning,* 24**,** 123-140.

BREIMAN, L. 2001. Random forests. *Machine learning,* 45**,** 5-32.

CHO, K., VAN MERRIëNBOER, B., GULCEHRE, C., BAHDANAU, D., BOUGARES, F., SCHWENK, H. & BENGIO, Y. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

GANG, C., SHOUHUI, W. & XIAOBO, X. Review of spatio-temporal models for short-term traffic forecasting. Intelligent Transportation Engineering (ICITE), IEEE International Conference on, 2016. IEEE, 8-12.

HOCHREITER, S. & SCHMIDHUBER, J. 1997. Long short-term memory. *Neural computation,* 9**,** 1735-1780.

LI, Y., ZHENG, Y., ZHANG, H. & CHEN, L. Traffic prediction in a bike-sharing system. Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2015. ACM, 33.

OLAH, C. 2015. *Understanding LSTM Networks* [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/ [Accessed June 20 2017].

SINGHVI, D., SINGHVI, S., FRAZIER, P. I., HENDERSON, S. G., O'MAHONY, E., SHMOYS, D. B. & WOODARD, D. B. Predicting Bike Usage for New York City's Bike Sharing System. AAAI Workshop: Computational Sustainability, 2015.

VLAHOGIANNI, E. I., GOLIAS, J. C. & KARLAFTIS, M. G. 2004. Short‐term traffic forecasting: Overview of objectives and methods. *Transport reviews,* 24**,** 533-557.

WANG, W. 2016. Forecasting Bike Rental Demand Using New York Citi Bike Data.

XU, J., ZHANG, Z. & RONG, J. 2012. The forecasting model of bicycle parking demand on campus teaching and office district. *Procedia-Social and Behavioral Sciences,* 43**,** 550-557.

ZHANG, J., ZHENG, Y., QI, D., LI, R. & YI, X. DNN-based prediction model for spatio-temporal data. Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2016. ACM, 92.

ZHANG, J., ZHENG, Y., QI, D., LI, R., YI, X. & LI, T. 2017. Predicting Citywide Crowd Flows Using Deep Spatio-Temporal Residual Networks. *arXiv preprint arXiv:1701.02543*.