

A Comparative Study of Using Spatial-Temporal Graph Convolutional Networks for Predicting Availability in Bike Sharing Schemes

Zhengyong Chen, Hongde Wu, Noel E. O'Connor and Mingming Liu *

Abstract—Accurately forecasting transportation demand is crucial for efficient urban traffic guidance, control and management. One solution to enhance the level of prediction accuracy is to leverage graph convolutional networks (GCN), a neural network based modelling approach with the ability to process data contained in graph based structures. As a powerful extension of GCN, a spatial-temporal graph convolutional network (ST-GCN) aims to capture the relationship of data contained in the graphical nodes across both spatial and temporal dimensions, which presents a novel deep learning paradigm for the analysis of complex time-series data that also involves spatial information as present in transportation use cases. In this paper, we present an Attention-based ST-GCN (AST-GCN) for predicting the number of available bikes in bike-sharing systems in cities, where the attention-based mechanism is introduced to further improve the performance of an ST-GCN. Furthermore, we also discuss the impacts of different modelling methods of adjacency matrices on the proposed architecture. Our experimental results are presented using two real-world datasets, Dublinbikes and NYC-Citi Bike, to illustrate the efficacy of our proposed model which outperforms the majority of existing approaches.

I. INTRODUCTION

Nowadays, there is increasing interest and demand for adopting bike-sharing systems globally. In fact, an efficient bike-sharing system can not only reduce cost and commute time for urban commuters, but can also effectively mitigate the level of air pollution emissions generated in cities [1]. An important consideration to make bike-sharing system efficient is to balance supply and demand in the bike-sharing network [2]. To do this, traditional management methods such as manual monitoring systems, have been deployed to enable relocation of bikes across different stations using other means of transportation, e.g. trucks. However, this approach can easily lead to supply-demand imbalance due to estimation errors of system operators and unexpected traffic delays during the bike transition. Thus, due to the uncertainty of departure and arrival of bikes at any bike station, it is important to take a more proactive approach by accurately predicting the number of bikes that will be available for users to access at any given time and location.

Recently, convolutional neural networks (CNN) have been applied to extract the relationship between adjacent traffic networks whilst the recurrent neural networks (RNN) were used to arrest the temporal information. For short-term traffic prediction, fully connected long short-term memory (LSTM) [3] and CLTFP [4], two architectures mixed the long short-term memory networks with convolutional operation, were

proposed in order to catch both temporal and spatial cues. However, LSTM or other networks with recurrent architecture are computationally intensive and hard to converge the network parameters with global optimization, since the recursive training accumulates the error for the prediction. On the other hand, CNN-based methods also have their limitation since the convolution process the data in 2-D form restrictively, which may not be the natural structure of traffic data.

These above issues of CNN and RNN-based methods were investigated and addressed by the spatial-temporal graph convolutional networks (ST-GCN) [5], a variant of a graph neural network (GNN) for utilizing spatial information. Spatial-temporal convolutional blocks were introduced and applied repeatedly in this architecture, combining several graph convolutional layers [6] with sequential convolution in order to represent the spatial-temporal relations. Subsequent to this approach, STG2Seq [7], a sequence-to-sequence variant of STGCN, is proposed with more reference on historical data and an attention module, for multi-step passenger demand forecasting. However, there are still some important issues to be solved in the ST-GCN architecture. For instance, how effective a specific adjacency matrix scheme can contribute to traffic demand prediction. Also, to what extent an attention-based mechanism can be applied to further improve the accuracy for a given demand prediction model.

To answer these questions, our key objective in this paper is to investigate how ST-GCN, supplemented with an attention-based mechanism, can further enhance the performance of bike availability prediction across different bike stations in cities. From an application/service perspective, we believe the proposed method can help cyclists make their personalized travel plan more appropriately by finding the best bike station nearby with high confidence in availability. Thus, the contribution of our work can be summarized as follows:

1. We combine an attention mechanism with the ST-GCN, namely AST-GCN, to improve the ability of extracting spatial-temporal features for the prediction task. In comparison with the existing methods, our model shows a promising performance.
2. We review related works in the recent literature and summarize four categories for modelling adjacency matrices, namely spatial based, temporal based, spatial-temporal based and adaptive based adjacency matrix.
3. Given our findings in 1 and 2, we evaluate our proposed AST-GCN model with the adjacency matrices of interest using a real-world dataset, Dublinbike, for bike sharing

Z. Chen, H. Wu, N. O'Connor, and M. Liu are with the School of Electronic Engineering, Dublin City University, Dublin, Ireland. Z. Chen and H. Wu are joint first authors.

*Corresponding author. Email: mingming.liu@dcu.ie

availability prediction. Our results show that: (a) adaptive spatial-temporal adjacency matrix can achieve the best performance; (b) spatial-temporal based adjacency matrix can achieve better results than that only using spatial-based or temporal-based adjacency matrix; (c) spatial-based adjacency matrix achieves similar performance as the temporal-based one.

The rest of the paper is organized as follows. We introduce some previous researches related to traffic demand prediction in Section II and formulate our problem in Section III. Experimental setups are demonstrated in Section IV and the results are discussed in Section V. Finally, we summarize our work in Section VI.

II. RELATED WORK

A. Existing Methods

In general, forecasting traffic demand is difficult, when a traffic demand depends not only on the historical demand pattern of the target area (e.g., suburb) but also on the pattern of other areas (e.g., urban). To meet this challenge, many studies using deep learning such as CNN, RNN, and GNN have been proposed.

As the traditional convolutional operation in CNN process the data with a 2D approach, the layout of a city is geographically divided into square blocks in order to extract spatial relationships from all regions [8], nearest regions [9] or in other 2D forms [10]. RNN based methods and their variants [11] are applied to catch temporal correlation, for instance, structuring the historical traffic demand sequence for each region [3] and presented as a 1D feature-level fused architecture [4]. GNN based methods, with natural advantages in utilizing spatial information, model the traffic network by a general graph instead of treating the traffic data arbitrarily (e.g., grids and segments) in CNN and RNN methods. GCN, as a variant of GNN, which is able to combine spatial and temporal information, is widely used in the scenario of traffic demand prediction as seen in many recent works [6] [7] [5].

Attention is a popular technique in deep learning that mimics physiological cognitive attention. The effect enhances the importance of small parts of the input data and de-emphasising the rest. This technique has been used to enhance the prediction performance for many sequence-based tasks of GNNs, i.e. Graph attention networks [12]. In traffic demand prediction, the importance of each previous step to target demand is different, and this influence changes with time. For instance, a temporal attention mechanism [7] is able to add an importance score for each historical time step to measure the influence and this strategy can effectively improve the accuracy on prediction accuracy.

B. Adjacency Matrices

An adjacency matrix is used to indicate whether a pair of vertices is connected by edge or not in graph data. For a traffic network, it is important to understand how an adjacency matrix can be used to best capture the inter-connectivity between different nodes in the graph. To the

best of our knowledge, four types of adjacency matrices have been investigated in previous research works, namely spatial (S), temporal (T), spatial-temporal (ST) and adaptive (A). A spatial adjacency matrix is usually distance-based. Euclidean distances between different stations (i.e., nodes in graph) [5] [13] or the natural geographical distance [14] are usually used as weights for its entries. A temporal adjacency matrix can be defined based on the similarity score [7] (i.e., Pearson correlation coefficient) between the temporal information (i.e., historical traffic demand sequence) of each pair of nodes/stations. To combine the benefits of both spatial and temporal features, an spatial-temporal embedding (ST embedding) can be generated for each node in a graph [15]. However, in such a scenario, it can be hard to describe the adjacency matrix intuitively with the high dimension embedding features and thus the adjacency matrix needs to be adaptively defined along with the training process of GCN [16] [17].

III. METHODOLOGY

A. Notations and Problem Statement

We consider a scenario where N bikes stations are included as part of a bike-sharing system. Let $\mathbf{N} := \{1, 2, \dots, N\}$ be the set for indexing the bike stations in the system. For a given bike station $i \in \mathbf{N}$, let $A_t^i \in \mathbb{R}$ be the number of available bikes at the station i at time t . We denote $\mathbf{A}_t \in \mathbb{R}^N$ the vector consisting of the number of available bikes across all stations N at time t . In addition, each bike station i is associated with a set of features for model training, e.g. weather condition, weekday, etc, and let $F_t^i \in \mathbb{R}^d$ represent the values of its features at time t , where d is the number of features used. Similarly, we let $\mathbf{F}_t \in \mathbb{R}^{N \times d}$ be the feature set values of all bike stations at time t . Given the notation above, our learning objective is to find a function $\mathbf{H}(\cdot)$ which is able to address the following problem:

$$\mathbf{A}_{t+1:t+n} = \mathbf{H}(\mathbf{A}_{t-m+1:t}; \mathbf{F}_{t-m+1:t})$$

where m, n denotes the input and output length for the model respectively. Also, the notation $t+1:t+n$ presents the output as a sequence of vectors from steps $t+1$ to $t+n$.

B. Attention-based ST-GCN

In this section, we introduce the attention-based ST-GCN architecture that used for solving our bike sharing availability prediction problem. We note that the ST-GCN architecture has been presented in [5], and the architecture consists of two identical ST-Conv-Blocks and a fully connected output layer. Specifically, an ST-Conv-Block consists of two temporal gated convolutional (TGC) layers and one spatial graph convolutional (SGC) layer, which are the essential modules of ST-GCN. In general, TGC is in charge of extracting temporal features and SGC is able to extract spatial features from the data. However, since there is no attention on the temporal channel of ST-GCN, this significantly degrades the performance for sequence to sequence based learning tasks. As such, the model's

learning capability may be significantly reduced due to “lost of focus”. To deal with this issue, we introduce a temporal-attention module (TAM) in each ST-Conv-Block, as shown in Fig. 1 where the temporal-attention module is depicted in green.

Remark: An attention mechanism was introduced in [18] and [19] to extract both spatial and temporal information from ST-GCN networks. The architectures proposed in both works applied attention operation to extract spatial and temporal information separately. In particular, the model in [18] consisted of 15 ST-Conv blocks in total with two attentions matrices calculated from them, while the model in [19] was stacked by 10 ST-Conv blocks with two attention matrices computed from each ST-Conv block. With increased model complexity and computation cost, stacking multiple ST-Conv blocks with attention matrices calculated separately may be of less interest since the spatial and temporal information may not be combined towards an effective spatial-temporal embedding in such a case. Instead, our model only consists of 2 ST-Conv blocks and the proposed AST-GCN architecture lightly merges spatial-temporal information with attention by calculating the attention matrix only once in each ST-Conv Block, which reduces the computation costs during the model training process. Specifically, the first TGC module generates original temporal information and the last TGC module generates spatial-temporal information (as it takes account of the output of the preceding SGC layer as its input). Passing through two average 3D pooling layers, both information are combined before a Relu activation function is applied. A sigmoid function is connected here to generate probabilistic weights (attention matrix) with values between 0 and 1. With this matrix in place, the attention-based temporal information is generated by using a dot product with the output of the first TGC layer and then concatenated as input to the subsequent ST-Conv Block. Both spatial and temporal information in the data flow are fully captured before passing to the dense layer for sequential output prediction.

IV. ALGORITHMS AND EXPERIMENTS

In this section, we discuss the different configurations investigated for comparative studies.

A. Experimental Datasets

- Dublinbike: DublinBikes is a bike-sharing scheme in operation in Dublin City, Ireland. The system is illustrated in Fig. 2, where each node is a bike station and each blue number in the circle indicates the number of available bikes in real-time. Real-time data is accessible using an API and we also have access to historic data, recorded every five minutes, which includes timestamps, station states, number of available bikes and station locations, etc. We choose the data ¹ from 01/07/2020 to 01/10/2020 for our studies.

¹<https://data.smartdublin.ie/dataset/analyze/33ec9fe2-4957-4e9a-ab55-c5e917c7a9ab>

- NYC-Bikes [20]: This dataset includes the NYC Citi daily bike orders of people using the bike sharing scheme. We choose the transaction records from April 1st, 2016 to June 30th, 2016 (91 days). This contains the following information: bike pickup station, bike drop-off station, bike pick-up time, bike drop-off time and trip duration.
- Visualcrossing Weather Data ²: This dataset provides weather conditions at different locations at different historical time points, including temperature, humidity and wind speed, etc. This weather dataset has been integrated for experiments that use the Dublinbikes dataset.

B. Experimental Setup

- Dublinbike: For this scenario, we use the number of available bikes at each bike station in the first 3 hours to predict the number of available bikes at each bicycle station 45 minutes later, where each data point is the averaged number of available bikes in 15 minutes. This implies that we take the past 12 consecutive observation points to predict the following 3 points of our interest. The dataset consists of 110 bike stations in total. The data is then separated into a training set (60%), a validation set (20%) and a testing set (20%) in a sequential manner.
- NYC-Bikes: NYC Citi Bike is dock-based and every depot of bikes is considered as a station. Following the same experiment setup as in CCRCN [15], we filter out the stations with fewer orders and keep the 250 stations with the most orders. The time step is set to half an hour. Among the last four weeks considered, the first two are used for validation, and the last two are for testing.

To evaluate the performance across different models, Mean Absolute Error (MAE) has been selected as the performance metric, indicating an intuitive margin between the predicted and the true amount of available bikes at each station.

C. Baseline Algorithms

- Dublinbike: To the best of our knowledge, there has been no GNN based methods implemented for the Dublinbike dataset. In particular, there has also been no ST-GCN based methods applied for solving the prediction for this dataset. For comparative studies, we conduct the experiments and use ST-GCN [5] as our baseline.
- NYC-Bikes: A lot of methods have been reported using this dataset to predict traffic demand. The state of the art work is presented in CCRCN [15]. Based on this, we compare the performance of different methods, including our proposed model, in a similar experimental setting. Specifically, the following methods are compared: (a) HA ³; (b) XGBoost [21]; (c) FC-LSTM [22];

²<https://www.visualcrossing.com/weather-data>

³The average of historical values at previous time steps of a fixed length.

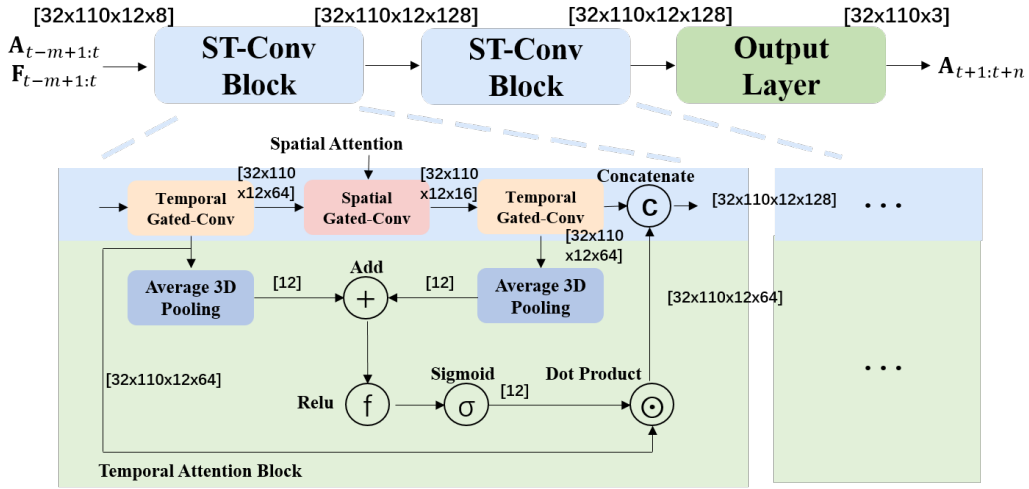


Fig. 1. The proposed ST-GCN with a temporal-attention module (TAM).

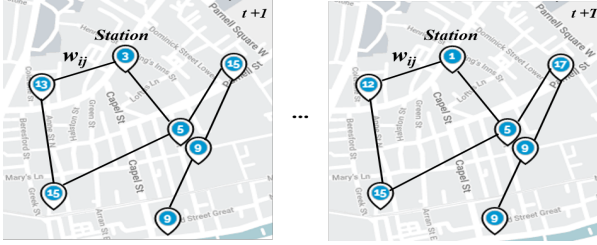


Fig. 2. A subset of bike stations of the Dublin bike sharing system operates in real-time. The edges are added for the illustration of the inherent graph signals.

- (d) DCRNN [23]; (e) ST-GCN [5]; (f) STG2Seq [7];
 (g) GraphWaveNet [16] and (h) CCRNN [15].

D. Network Setup

The historical data length used for both the Dublinbikes dataset and the NYC-citi dataset is set to 12, the prediction length is set to 3 in Dublinbikes and 12 in NYC-citi respectively. The feature dimension used in NYC-citi is 2 representing the pick-up and drop-off demand. The feature dimension used for the Dublinbikes dataset is 8, details of the feature selection will be discussed in the results section. All models are optimized by Adam algorithm [24]. Other setting of parameters are presented in Table I. The dimensions of the data flow during the training process of the proposed model are overlapped in Fig. 1 for illustration purposes. It is worth noting that the input of the first temporal gated-Conv is strictly the same as the input of the corresponding ST-Conv block while the input of the second temporal gated-Conv is the output of previous spatial gated-Conv block. The concatenate operation concatenates the output of the first and the second temporal gated-Conv block.

E. Adjacency Matrix Setup

The adjacency matrix in the original ST-GCN architecture is not adjustable/trainable. As a result, this fixed adjacency matrix may not fully capture the spatial relationship between

TABLE I
 EXPERIMENT SETTING FOR TWO DATASETS

Setup	Dublinbikes	NYC-citi
Station amount	110	250
Historical data length	12	12
Prediction length	3	12
Feature dimension	8	2
Batch size	32	32
Initial learning rates	0.001	0.0001
Optimizer	Adam algorithm	Adam algorithm
Weight decay	0.001	N/A
LR adjustment strategy	cosine annealing	adjust at equal intervals

nodes in the graph. To improve it, we adapt the fixed adjacency matrix to a trainable adjacency matrix and then initialize the matrix using meaningful contextual information, e.g. distance between nodes, similarity between stations' historical time-series data. Further, an adaptive adjacency matrix (AAM) is able to extract spatial attention information from the graph adaptively, and thus it makes the AST-GCN effective in capturing both spatial and temporal attention information. For our comparative studies, different setups of adjacency matrices are investigated as follows:

- For the implementation of the adjacency matrix proposed in ST-GCN [5], the sigma is set to 0.2 and the epsilon is set to 0.368;
- For the implementation of the adjacency matrix proposed in STG2Seq [7], the sigma is set to 0.05;
- For the implementation of the adjacency matrix proposed in CCRNN [15], the dimension of station feature is set to 20 and the sigma is set to 1.
- Other adjacency matrices do not need parameters to be set. In other words, these adjacency matrices are calculated directly without parameters or are purely adaptive.

V. RESULTS AND DISCUSSION

A. Feature Selection for Dublinbikes Dataset

In order to select the best features for our experiments, an ablation study has been carried out for a set of features which model temporal, spatial as well as weather characteristics. Specifically, we adopt ST-GCN as our basic setting for evaluation of different feature combinations. Our full feature sets are as follows: (1) number of available bikes (AB); (2) time of day (TD); (3) weekday (WD); (4) weather condition description (WCD); (5) temperature (T); (6) wind speed (WS); (7) cloud coverage (CC) and (8) Humidity (H).

Results of the ablation study are reported in the Table II, from which we easily conclude that the following feature combination gives the best performance: number of available bikes (AB), time of day (TD), weekday (WD) and weather conditions description (WCD).

TABLE II
RESULTS OF THE ABLATION STUDY OF FEATURE COMBINATIONS

Feature combination	MAE
AB	3.24
AB+TD	3.19
AB+TD+WD	3.21
AB+TD+WD+WCD	3.16
AB+TD+WD+WCD+T	3.36
AB+TD+WD+WCD+WS	3.40
AB+TD+WD+WCD+CC	3.30
AB+TD+WD+WCD+H	3.40
All Together	3.56

B. Results Discussion

- NYC-Bikes: The results on the NYC dataset are compared between the proposed AST-GCN and existing algorithms reported in [15] as shown in Table III. It is shown that the AST-GCN algorithm outperforms the existing graph based architectures (i.e., ST-GCN and STG2Seq) with 24.67% improvement in MAE, from 2.4976 to 1.8815. Also, although CCRNN beats all of its competitors, the AST-GCN shows minor difference in performance, and it still demonstrates comparable metrics compared to other sequence based models including Graph WaveNet, DCRNN.
- Dublinbikes: As shown in Table IV, after applying distance initialized AAM (DIAAM) on ST-GCN, the prediction results achieve better results with MAE equals 1.27. By replacing ST-GCN to AST-GCN, the MAE result has significantly improved from 1.27 to 1.04 for MAE. Among others, the embedding AAM (EAAM) makes the best performance which leads to the MAE equals 1. Results in Fig. 3 further highlight this key finding. Specifically, the biases between the ground truth and the first timestamp (i.e. NAB prediction for the first 15 minutes) as well as the third timestamp (i.e. NAB prediction for the 45 minutes) are both negligible showing that our proposed model can achieve impressive prediction performance for both short-term

(15 mins) and long-term (45 mins) for the best case scenario.

TABLE III
EXPERIMENT RESULT OF AST-GCN ON NYC-CITI [15]

Model	MAE
HA	3.4617
ST-GCN	2.7605
STG2Seq	2.4976
XGBoost	2.4690
FC-LSTM	2.3026
Graph WaveNet	1.9911
DCRNN	1.8954
AST-GCN + EAAM	1.8815
CCRNN	1.7404

TABLE IV
EXPERIMENT RESULT OF AST-GCN ON DUBLINBIKES

Model	Categories ¹	MAE (%)
ST-GCN + Euclidean distance	S	1.36 (0%)
ST-GCN + DIAAM	S + A	1.27 (-6.67%)
AST-GCN + DIAAM	S + A	1.04 (-23.5%)
AST-GCN + EAAM [16]	ST + A	1.00 (-26.5%)
AST-GCN + Euclidean distance [5]	S	1.06 (-22.0%)
AST-GCN + Geographical distance [14]	S	1.09 (-19.8%)
AST-GCN + Temporal correlation [7]	T	1.07 (-21.3%)
AST-GCN + ST embedding [15]	ST	1.01 (-25.7%)

C. Performance Evaluation w.r.t. Adjacency Matrices

In this section, we discuss how different adjacency matrices can affect the learning performance for our proposed AST-GCN architecture. Our results are illustrated in Table IV where the percentage in parenthesis shows the difference of the achieved MAE in comparison to the basic setting: ST-GCN + Euclidean distance. Unsurprisingly, our results show that those fixed adjacency matrices, including both spatial based and temporal based, achieve the worst results among all other settings. In contrast, the adaptive-based settings can generally achieve better results compared to the fixed types, but with one exception for the spatial-temporal based setting, i.e. AST-GCN + ST embedding, which also shows a competitive result. For the adaptive-based settings, the embedding AAM, i.e. AST-GCN + EAAM, achieves the best result compared to the other AAM setting initialized by distance, i.e. AST-GCN + DIAAM.

D. Performance Evaluation w.r.t. Different Bike Stations

In this section, we present the prediction results for each bike station in the Dublinbike dataset using the best trained model (AST-GCN + EAAM). Our objective here is to illustrate the confidence with which a user can rely on our proposed prediction model to make a decision when he/she decides to get access to a bike from his/her nearby area. Our station-wise results are illustrated in Fig. 4 and Fig. 5. Specifically, Fig. 4 shows the heat-map of station-wise MAE over the geographical map of Dublin city where the bike stations are facilitated. The red marks indicate a higher MAE and blue-green marks indicate a lower MAE

¹The abbreviations in this column have been presented in Section II-B.

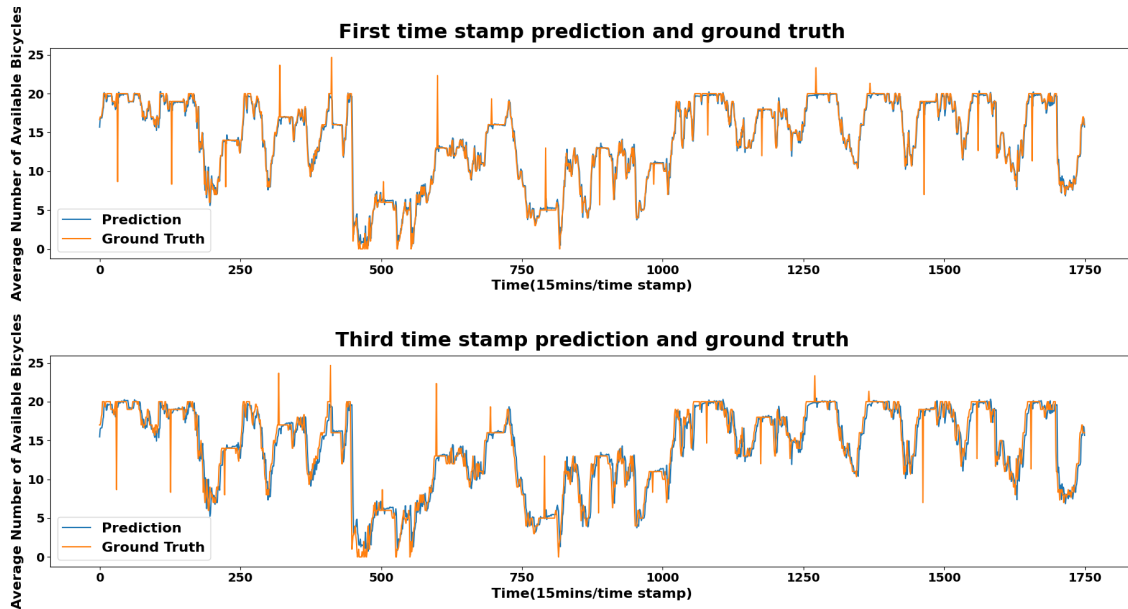


Fig. 3. Comparison between ground truth and prediction.

in the corresponding area. Generally speaking, the results demonstrate that the prediction is more accurate (low-MAE values) outside of the city center showing that users can collect bikes with high confidence in the availability of bikes. The highest prediction error occurs in the heart of city centre, i.e. the bike station located at the “Princes Street/O’Connell Street”, with the MAE equalling to 2.4. This may be caused by a frequent access and return of bikes by users in this central commuting area, leading to a relatively higher uncertainty in bike availability. The second highest prediction error appears in the western part of the city, i.e. the green-blue region indicated in the rectangular box in Fig. 4. However, this is mainly due to the aggregated effect where a few bike stations are very close to each other in the “Benburb Street” area. An in-depth view of the region, i.e., the upper left corner of the rectangular box in Fig. 4, further validates that the prediction error of each bike station therein is low. Finally, the statistical histogram of the station-wise MAE is illustrated in Fig. 5 showing that most bike stations have an MAE-based prediction error less than 1.5 bikes, which indicates that our proposed forecasting system is very robust and accurate for a number of bike stations in the Dublin city.

VI. CONCLUSION

In this paper, we propose a spatial-temporal graph convolutional network architecture embedded with a temporal-attention module (AST-GCN) to predict the number of available bikes in bike-sharing systems using realistic datasets. The temporal attention module is able to extract temporal attention information which aims to enhance the prediction accuracy compared to that of the original ST-GCN architecture reported in [5]. Our experimental results show that the proposed AST-GCN can perform better than most of existing

methods in the NYC-Citi dataset. As for the Dublinbikes dataset, our proposed model has demonstrated a very promising result of 1.00 MAE as the selected performance metric. In addition, we have thoroughly investigated how different modelling of the adjacency matrices can affect the overall model performance through a comprehensive comparative study on the DublinBikes dataset. Current results have shown that embedding AAM can achieve the best results compared to many other settings.

To conclude, we believe that the work presented in this paper is an important step towards making bike sharing systems more efficient thanks to the ST-GCN enabled techniques. Most importantly, we wish to note that it is our intention in this paper to deeply explore different variants of ST-GCN by using open access datasets to form a performance benchmark for the benefit of the research community. Further improvements to the network structure, adjacency matrices and advanced feature selection will be investigated as part of our future work.

ACKNOWLEDGMENT

This work is supported in part by Science Foundaition Ireland SFI Grant SFI/12/RC/2289 P2. Hongde Wu is supported by the research master scholarship at Dublin City University.

REFERENCES

- [1] I. Otero, M. Nieuwenhuijsen, and D. Rojas-Rueda, “Health impacts of bike sharing systems in europe,” *Environment international*, vol. 115, pp. 387–394, 2018.
- [2] T. Raviv and O. Kolka, “Optimal inventory management of a bike-sharing station,” *Iie Transactions*, vol. 45, no. 10, pp. 1077–1093, 2013.
- [3] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, and W. C. Woo, “Convolutional lstm network: A machine learning approach for precipitation nowcasting,” *Advances in neural information processing systems*, vol. 2015, pp. 802–810, 2015.

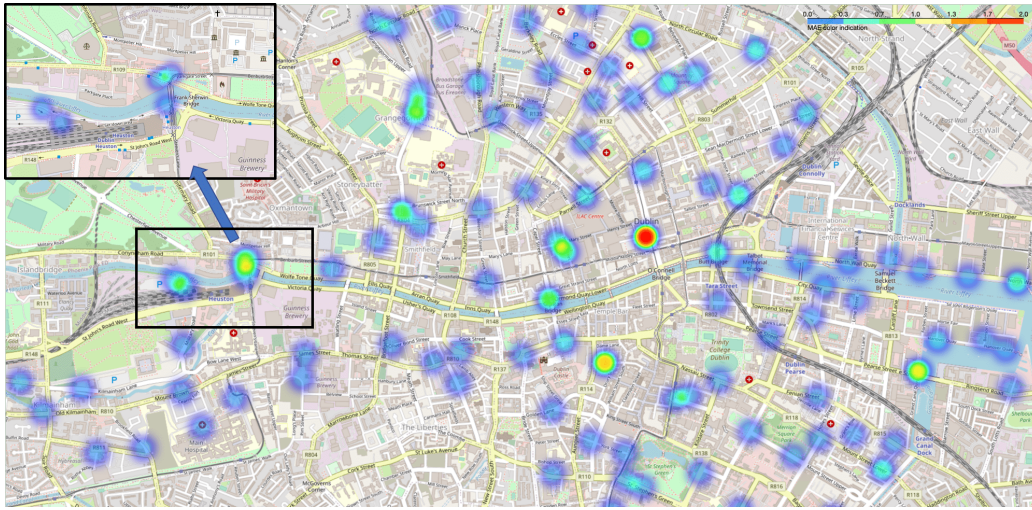


Fig. 4. Heat-map of the station-wise MAE-based prediction error in Dublin city. A warm-toned color (red) indicates a higher MAE and a cool tone color (green-blue) indicates a lower MAE.

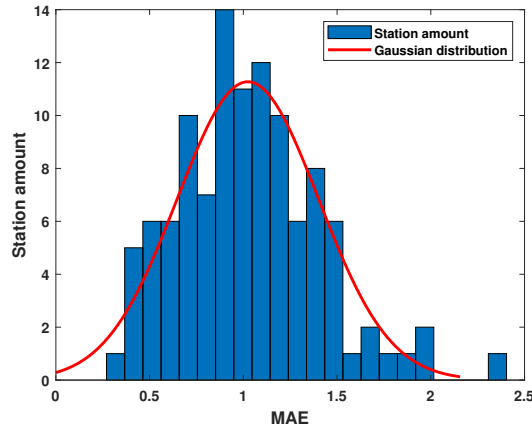


Fig. 5. Histogram of station-wise MAE.

[4] Y. Wu and H. Tan, "Short-term traffic flow forecasting with spatial-temporal correlation in a hybrid deep learning framework," *arXiv preprint arXiv:1612.01022*, 2016.

[5] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pp. 3634–3640, ijcai.org, 2018.

[6] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3837–3845, 2016.

[7] L. Bai, L. Yao, S. Kanhere, X. Wang, Q. Sheng, et al., "Stg2seq: Spatial-temporal graph to sequence model for multi-step passenger demand forecasting," *arXiv preprint arXiv:1905.10069*, 2019.

[8] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, 2017.

[9] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial-temporal network for taxi demand prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[10] J. Chu, X. Wang, K. Qian, L. Yao, F. Xiao, J. Li, and Z. Yang, "Passenger demand prediction with cellular footprints," *IEEE Transactions*

on Mobile Computing, 2020.

[11] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, pp. 5668–5675, 2019.

[12] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[13] Z. Chen, B. Zhao, Y. Wang, Z. Duan, and X. Zhao, "Multitask learning and gcn-based taxi demand prediction for a traffic road network," *Sensors*, vol. 20, no. 13, p. 3776, 2020.

[14] T. S. Kim, W. K. Lee, and S. Y. Sohn, "Graph convolutional network approach applied to predict hourly bike-sharing demands considering spatial, temporal, and global effects," *PloS one*, vol. 14, no. 9, p. e0220782, 2019.

[15] J. Ye, L. Sun, B. Du, Y. Fu, and H. Xiong, "Coupled layer-wise graph convolution for transportation demand prediction," *arXiv preprint arXiv:2012.08080*, 2020.

[16] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," in *International Joint Conference on Artificial Intelligence 2019*, pp. 1907–1913, Association for the Advancement of Artificial Intelligence (AAAI), 2019.

[17] W.-L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, and C.-J. Hsieh, "Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 257–266, 2019.

[18] K. Shiraki, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Spatial temporal attention graph convolutional networks with mechanics-stream for skeleton-based action recognition," in *Proceedings of the Asian Conference on Computer Vision*, 2020.

[19] W. Zhang, Z. Lin, J. Cheng, C. Ma, X. Deng, and H. Wang, "Stagcn: two-stream graph convolutional network with spatial-temporal attention for hand gesture recognition," *The Visual Computer*, vol. 36, no. 10, pp. 2433–2444, 2020.

[20] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "Dnn-based prediction model for spatio-temporal data," in *the 24th ACM SIGSPATIAL International Conference*, 2016.

[21] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, (New York, NY, USA), p. 785–794, Association for Computing Machinery, 2016.

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," vol. 9, p. 1735–1780, Nov. 1997.

[23] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting," *arXiv e-prints*, p. arXiv:1707.01926, July 2017.

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.