Predicting Travel Demand of a Docked Bikesharing System Based on LSGC-LSTM Networks

JING LUO1, DAI ZHOU1,2,3, ZHAOLONG HAN1,2,3, GUANGNIAN XIAO5, AND YUNLONG TAN6

1School of Naval Architecture, Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
2Shanghai Key Laboratory for Digital Maintenance of Buildings and Infrastructure, Shanghai Jiao Tong University, Shanghai 200240, China
3Key Laboratory of Hydrodynamics of Ministry of Education, Shanghai Jiao Tong University, Shanghai 200240, China
4State Key Laboratory of Ocean Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
5School of Economics and Management, Shanghai Maritime University, Shanghai 201306, China
6Guangzhou Transport Research Institute, Guangzhou 510627, China

Corresponding authors: Dai Zhou (zhoudai@sjtu.edu.cn) and Zhaolong Han (han.arkey@sjtu.edu.cn)

This work was supported in part by the Shanghai Natural Science Foundation under Grant 17ZR1415100 and Grant 18ZR1418000, in part by the National Natural Science Foundation of China under Grant 71803110, and in part by the Program for Intergovernmental International S&T Cooperation Projects of Shanghai Municipality under Grant 18290710600.

ABSTRACT The sustainable development of docked bikesharing systems has gained focus again owing to several problems in dockless bikesharing systems, including wanton destruction, theft, illegal parking, loss, and bankruptcy. The prediction of pickup/return demands is a critical issue for the sustainable operation of docked bikesharing systems. We propose a novel local spectral graph convolution (LSGC)- long short-term memory (LSTM) to predict pickup/return demands based on multi-source data. We apply LSGC to indicate the spatial dependency according to the geographic information system data that provide the location of stations, and we apply LSTM to demonstrate the temporal dependency based on the time-series data that represent pickup/return demands for public bikes. The LSGC-LSTM and six baseline models are trained with multi-source data of a month from a docked bikesharing system. The baseline models consist of a recurrent neural network, a LSTM, a gated recurrent unit, a graph attention LSTM network, an adaptive graph convolutional recurrent network, and a dynamic graph convolutional neural network. Results indicate that the LSGC-LSTM obtains a higher prediction accuracy and a higher efficiency than the baseline models.

INDEX TERMS Intelligent transportation systems, road transportation, transportation.

I. INTRODUCTION

Most cities are struggling with “urban disease,” including traffic congestion, and air and noise pollution [1], [2]. The “2019 Third Quarter Urban Transport Analysis Report of Major Cities” issued by Gaode Map pointed out that the daily traffic indexes of the 10 most congested cities exceed 1,700 on average. Harbin city has the highest index (1.948), demonstrating that the travel duration during rush hour is 94.8% more than that in the free-flow condition on average. Thus, transportation sharing including bike, e-bike and car sharing has gradually become an alternative to private transportation [3]. As an active, environmentally friendly and convenient mode [4], [5], bike-sharing comprises the docked and dockless bike-sharing programs [6]. Docked bikesharing users may swipe membership/citizen cards or scan QR codes with smartphones to pick up/return bikes, while dockless bikesharing users may scan QR codes with smartphones or enter the password (mainly for the early version of Ofo) to pick up bikes and return them by turning off the lock [7]. Recently, a growing number of travelers have been making short-distance trips with shared bikes, rather than commuting, for fear of contracting Corona Virus Disease 2019. To avoid the cross infection of travelers, the China Urban Public Transport Association established the group standard of the “Hygiene Guarantee Operation Specification of Internet Rental Bicycles” on February 17, 2020.

China was referred to as “the bike kingdom” in the 1980s [8]. Nevertheless, the bike ridership had been seeing a dramatic decline since the 1990s because cars quickly
entered thousands of Chinese households. This downward trend turned when docked bikesharing programs were introduced. The first docked bikesharing program in China was launched by a private enterprise in Beijing in 2005 [9]. The program peaked in 2008 with more than 8,000 bikes distributed at 100 sites, but went bankrupt quickly because of successive losses in 2009. The pilot docked bikesharing program operated by governments in China was implemented in 2008 in Hangzhou, a city with an outstanding cycling facility. Ten years later, this docked bikesharing program had become a large-scale network with 101,700 bikes distributed to 4,198 sites. As of 8:40 on October 23, 2019, the rental of the Hangzhou docked bikesharing program has reached 1 billion times. One of the bikes that was put into use on September 15, 2009 became the “king of rent” for having been rented a total of 26,141 times. Given the low price and ease of use, docked bikesharing programs are popular with residents and have ushered in the spring of development. By July 2018, 256 cities in China had initiated their docked bikesharing programs, attracting over 50 million users. As an alternative to docked bikesharing programs, dockless bikesharing programs were introduced in 2015. Ofo, which was created by undergraduate students of Peking University, was the pilot dockless bikesharing program. Subsequently, many dockless bikesharing programs were implemented, attracting many users. At the peak of dockless bikesharing programs, more than 50 programs and 125 million users were involved. According to the data from iimedia (https://www.iimedia.cn/c1061/73647.html), the numbers of users of dockless bikes in China in 2016, 2017, 2018, 2019, and 2020 were 28, 205, 235, 256, and 253 million, respectively. The users began to decrease in 2020, because of problems in dockless bikesharing systems, including wanton destruction, theft, illegal parking, loss and bankruptcy. As the only listed company in the field of bikesharing, Youon Technology Co., Ltd. (http://www.ibike668.com) launched increasing docked bikesharing programs. In 2015, 2016, 2017, 2018, and 2019, the numbers of Youon docked bike-sharing programs were 165, 210, 252, 275, and 290, respectively. Although the number of docked bikesharing users is less than that of dockless bikesharing, the number of docked bikesharing users has been growing steadily. Predicting the pickup/return demands for sites/areas may be important for both docked and dockless bikesharing programs [10]. Accurate demand prediction underlies a low rebalance cost and a high utilization rate [11].

Most existing studies on demand prediction use statistical models at the station level [12]. Buck and Buehler developed a multiple regression model for determining the contributory factor of the demand of the Capital Bikesharing system in Washington, DC [13]. Their results indicate that the number of nearby bike lanes, the population density, and the mixed land uses have a significant impact on the demand. Wang et al. built log-linear and negative binomial regression models for analyzing the determinants of average daily station trips, and found that neighborhood demographics, the distance to the central business district, and the proximity to trails and other bikesharing stations are the main determinants [14]. Gebhart and Noland predicted the hourly usage of the Washington, DC, bikesharing system based on hourly weather data, including rainfall, wind, fog, temperature, and humidity levels [15]. Their results demonstrate that the hourly usage of the bikesharing system is negatively affected by high humidity levels, rain, and cold temperatures.

To improve the prediction accuracy of the pickup/return demands for the bikesharing system, spatial econometrics were used to consider spatial dependencies. Rudloff and Lackner predicted pickup/return demands for the bikesharing system Citybike Wien in Vienna, Austria based on the weather and historical demand data from full or empty neighboring stations [16]. Zhang et al. built a multiple linear regression model for analyzing the effect of built environment and the usage of nearby stations on pickup/return demands for the Zhongshan public bike system [17]. Their results indicate that the length of bike lanes and branch roads, population density, mixed land uses, and the usage of nearby stations are the key variables. Bao et al. used the geographically weighted regression method to explore the effect of multiple influencing factors on the bikesharing ridership of the Citi Bike system of New York City [18]. Shen et al. developed spatial autoregressive models for analyzing the impact of weather conditions, the bicycle infrastructure, the access to public transportation, the surrounding built environment, and the impact of bike fleet size on the usage of Singapore dockless bikes [19]. They concluded that the usage of a station is significantly affected by the usage of its nearby stations.

The concept of intelligent transportation systems and smart cities has prompted scholars to integrate big data and machine learning into the transportation system analysis system [20], [21]. Meanwhile, as a type of neural network for non-linear machine learning models, graph neural networks (GNNs), which fill the gaps in graph data learning, have received extensive attention in recent years [22], [23]. Various variants of GNNs are also adopted in the field of transportation research. For example, the use of graph data, which reflect the temporal and spatial relationship, with the temporal and spatial convolutional network can accurately predict short-term traffic flow in real time. Therefore, the natural and inevitable trend is to effectively adapt the GNN to the traffic problem, and the GNN has strong application capabilities and an application value.

The learning process of the GNN can be directly built on graph data [22], [24], [25]. In 2009, scholars further elaborated the GNN, defined the theoretical basis of the GNN, and proposed a supervised learning method to train the network model [26]–[29]. Subsequently, researchers further introduced convolution into the GNN and applied learnable convolution operations to graph data for the first time [30]. Then, researchers focused on improving and expanding this frequency-domain graph-based convolutional neural network (CNN) model, in attempt to simplify the complexity of the model, so that the learning system can be extended to
large-scale graph data learning tasks. Therefore, a graph CNN based on spatial domain was proposed. This model has greatly improved the computational efficiency of the graph convolution model [31]. The graph convolution model and its variants based on the spatial domain strengthen the adaptability of the model to various graph data and lay a solid foundation for the wide application of GNNs [32].

GNN has also achieved good application effects in the transportation field. For example, in terms of short-term traffic flow prediction, traffic flow data have highly nonlinear and complex relationship patterns. Traditional prediction methods are lacking in modeling the dynamic temporal and spatial relationships of traffic flow. Thus, a number of models based on graph GNNs are proposed. Yu et al. developed a deep learning framework named spatio-temporal graph convolutional networks to solve the time-series prediction problem [33]. The framework is characterized by its faster training speed and fewer parameters compared with its regular recurrent and convolutional units. Lin et al. used six types of adjacency matrices (including spatial distance matrix, demand matrix, average trip duration matrix, and demand correlation matrix) to describe temporal dependencies in predicting the bikesharing demand [12]. Similarly, Bai et al. proposed two adaptive modules for enhancing graph convolutional network, a node adaptive parameter learning module and a data adaptive graph generation module, to avoid the predefined graph [28]. To effectively predict the traffic demand when spatial dependencies change over time, Diao et al. and Chen et al. built dynamic spatio-temporal graph convolutional neural networks to estimate the dynamic Laplacian matrices and used the models to the short-time speed and the taxi demand [20], [25]. Furthermore, the graph convolutional neural network models are extended from a variety of perspectives. Yu et al. applied an original three-dimensional (3D) graph convolutional model to fit the spatio-temporal data with a higher accuracy compared with those with 2D graph convolutional models [23]. Wu et al. extended the LSTM model by introducing the graph attention structure to improve the interpretability and capture complex spatio-temporal correlations according to the dataset from the “Guiyang traffic data platform” (http://www.guiyangdata.gov.cn) [22]. The proposed model is used to predict the multi-link traffic flow demand. Chen et al. applied the multitask learning technology based on the graph convolutional network to predict the taxi departure and arrival flows with real-world taxi trajectories from the city of Xi’an [29]. Zhang et al. combined a graph convolutional network and a 3D CNN network to integrate the inflow and outflow information for the urban rail transit [34]. For more details, please refer to a comprehensive review by Wang et al. [35].

Existing studies of spatio-temporal deep learning models promote the research in traffic prediction tremendously. However, these models become more and more complicated, resulting in a potentially low interpretability and spatio-temporal transferability. Therefore, we combine GNN and long short-term memory (LSTM) to represent the temporal and spatial dependencies of a docked bikesharing program, and predict pickup/return demands. On the other hand, the sample size (including the number of shared bikes in a bikesharing program and the number of links in a transportation network) increases rapidly in the era of big data, which makes it more important to use an efficient demand prediction model. Therefore, we use local spectral graph convolution (LSGC) to decrease the computation burden and improve the availability of the real-time demand prediction. Therefore, the contributions of this paper are twofold. First, we employ GNN and the commonly used LSTM to describe the temporal and spatial dependencies to improve the spatio-temporal transferability. Second, we apply LSGC to improve the computation efficiency and facilitate the real-time prediction.

II. PROBLEM DESCRIPTION
A. DATA DESCRIPTION
The multi-source data employed in the current study comprise the network structure and 452,964 usage records in September 2016 of the docked bikesharing system of a city in Zhejiang Province. The city is in the southeast coast of Zhejiang Province and the southern wing of the Yangtze River Delta. The city has a land area of 926 km$^2$ and a total registered population of over 1,221 thousand as of the end of 2018. The city ranked 15th in “Top 100 Counties and Cities in National Scientific and Technological Innovation in 2019,” indicating that it possesses a strong financial foundation to support a large-scale docked bikesharing system. The results of the 2016 household travel survey indicate that the mode rate (the proportion of trips with a certain travel mode in terms of all trips for a specific region or a city) of bikes was 5.09%, which is near that of normal bus transits (5.46%).

The docked bikesharing system comprises 186 stations in the downtown area. Cyclists may find a docked bikesharing bike by walking 300 m at most wherever in the old downtown area. The bike rent is free for an hour, and one yuan for each hour, which is paid by swiping exclusive docked bikesharing membership cards or versatile citizen cards. A cyclist may return the public bike in an hour and take the same or another docked bike immediately to continue riding and avoid charging. The sample data of usage records are provided in Table 1. To guarantee the accuracy of statistical analysis and demand prediction, the usage records are cleaned to eliminate records that do not represent the actual demands. The steps and rules for cleaning the records are shown as follows.

1. Fake pickup/return records are cleaned. A pickup/return record is regarded as fake if a cyclist takes a broken public bike, which may be caused by a broken chain or a flat tire, and returns it within a short time. Thus, we eliminate the usage records if the pickup station is the same as the return station and the duration is equal to or less than 2 min. The critical value of 2 min is employed to prevent the deletion of actual usage records that have the same pickup and return station. According to the aforementioned rule, up to 28,689 usage records are eliminated.
TABLE 1. Sample data of usage records in the docked bikesharing system.

| Name of origin station | ID of origin station | ID of origin pile | Pickup time | Name of destination station | ID of destination station | ID of destination pile | Return time | Cycling time (min) | Rent (yuan) |
|------------------------|---------------------|-------------------|-------------|----------------------------|--------------------------|-----------------------|-------------|-------------------|------------|
| Experimental primary school | 57 | 28 | 2016/9/1 08:40:57 | Maiyu bridge | 94 | 5 | 2016/9/1 10:23:30 | 102 | 1 |
| Finance bureau | 101 | 10 | 2016/9/1 08:57:06 | Fire department | 64 | 1 | 2016/9/1 09:03:23 | 19:49:47 | 50 | 0 |
| Daye apartment | 24 | 1 | 2016/9/10 18:59:41 | Fire department | 64 | 11 | 2016/9/19 21:10:17 | 3 | 0 |
| Wenzhong experimental school | 151 | 2 | 2016/9/19 21:07:07 | Dayuan bridge | 55 | 29 | 2016/9/25 16:18:34 | 22 | 0 |
| Wulongsan park | 191 | 1 | 2016/9/25 15:56:24 | Police station | 8 | 12 | 2016/9/25 | |

2. Incomplete usage records are cleaned. A pickup/return record is recognized as incomplete if at least one field is missing. Incomplete usage records are created due to data coding errors or false writing operation. Incomplete usage records may result in a biased result for the statistical description and decrease the accuracy of demand prediction. According to this rule, a total of 1,317 incomplete usage records are eliminated.

3. Repair and maintenance records are cleaned. Public bikes are generally repaired and maintained by the staff members of the docked bikesharing system. The staff members move public bikes that are broken or must be updated in repair/maintenance centers and carry the repaired/updated public bikes to proper stations. Repair/maintenance records may be identified through null usernames or origin/destination station names specified by “the Docked Bikesharing Administration Center.” On the basis of this rule, up to 39 usage records are deleted.

4. Reposition records are cleaned. Repositioning is employed to shift public bikes from stations with insufficient demand to stations with insufficient supply to balance the docked bikesharing distribution at all stations. Reposition records must be deleted because they are used to provide guarantee for regular demands. Reposition records may be flagged if the usernames contain the word “test.” On the basis of this rule, a total of 665 usage records are removed.

Overall, up to 30,201 usage records are eliminated after data cleanup according to the aforementioned cleaning conditions, and a total of 422,763 usage records are reserved for the subsequent statistical analysis and demand prediction. Like the docked bikesharing systems implemented in other cities, the docked bikesharing system in this subject city is mainly used to complete shuttle trips and trips with short duration/distances. As depicted in Figures 1 and 2, the travel distance and duration of more than 80% of the trips are less than 2,050 m and 20 min, respectively.

B. THE TEMPORAL DEPENDENCY
The temporal dependency is commonly present for time-series or panel data. Figure 3 denotes the aggregate pickup/return demands for all 186 stations in morning/evening rush hours, with the labels in the horizontal axis representing the day of the month. The morning rush hours are between 7:00 and 9:00, and the evening rush hours are between 16:00 and 18:00. The definition of rush hours is suitable for most cities. The pickup demand is usually greater than the return demand during morning/evening rush hours, and that in the morning rush hours is obviously greater than that in the evening rush hours. The lowest return demand in the morning rush hours was 459, which was recorded on September 17. This low demand was generated due to the heavy rain throughout the day. The characteristic of the pickup time in an entire week (9/5/2016-9/11/2016) is shown in Figure 4 to present the periodic property in the days of the week. The characteristic of the pickup time at weekdays is more similar than that at weekends. Nevertheless, notable differences in demands at different weekdays exist. For example, the pickup demand at Wednesday afternoons is much less than that at other weekdays.

To further demonstrate the strength of the temporal dependency between the time of day and the day of the week, we use the two-sample Kolmogorov-Smirnov test to determine whether the distributions of the pickup demand for two days/weeks differ significantly with a significance level of 0.05. In terms of the time of the day, we analyze the pickup demand in an entire week (9/5/2016-9/11/2016) and compute the p-values for each pair of days as shown in Table 2. The results indicate that the pickup demand of the workdays and Saturday has a similar distribution, which is obviously different from those of Sunday. In terms of the day of the week, we analyze the pickup demand for three weeks (9/5/2016-9/11/2016, 9/12/2016-9/18/2016, and 9/19/2016-9/25/2016) and compute the p-values for each pair of weeks. All three p-values are $7.606 \times 10^{-7}$, indicating that the distributions of the pickup demand for the three weeks have a significant difference.

C. THE SPATIAL DEPENDENCY
Spatial dependency is generally seen in transportation related analysis, including the demand prediction of freight transport and passengers. Tobler [36] indicated that “Everything is
related to everything else, but near things are more related than distant things.” This statement is known as “The First Law of Geography.” A spatial weight matrix is erected with the reciprocal of the shortest distance between any two docked bikesharing stations according to the transportation network to evaluate the degree of spatial dependency between
a station and its neighboring stations. The normalized spatial weight matrix $W$ is obtained by calculating the reciprocal of each entry and standardizing each row of the spatial weight matrix sequentially. Thus, the standardized spatial weight matrix comprises $w_{ij}$ ($i, j = 1, 2, \ldots, N$). The global Moran’s $I$ index is defined according to Equation 1 and
determines the significance of the spatial dependency. The global Moran’s indexes for the pickup and return demands are 0.233 and 0.230, respectively. The corresponding p values of the global Moran’s I indexes for pickup and return demands are both 0.000. They are significant at the significance level of 0.05, demonstrating that the spatial dependency exists from a global perspective.

\[
I = \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}(x_i - \bar{x})(x_j - \bar{x})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}, \tag{1}
\]

where \( N \) denotes the number of docked bikesharing stations, \( x_i \) represents the accumulated pickup/return demands throughout September 2016 for the \( i \)-th station, \( \bar{x} \) indicates the average demand for all stations throughout September 2016, and \( w_{ij} \) signifies the normalized spatial weight for the \( i \)-th and \( j \)-th stations.

The local Moran’s I test is performed to identify the local spatial dependency of each station. The local Moran’s I indexes are given in Equation 2 for pickup/return demands. As shown in Figure 5, the high local Moran’s I indexes of the pickup demand are aggregated on the lower-left part, which is the urban area of the city. The distribution of the local Moran’s I indexes of the return demand is like that of the pickup demand. The local and global Moran’s I indexes indicate that spatial dependencies must be accommodated when the docked bikesharing demands are predicted.

\[
I_i = \frac{(x_i - \bar{x})}{S^2} \sum_{j=1}^{n} w_{ij}(x_j - \bar{x}), \tag{2}
\]

where \( S^2 = \sum_{i=1}^{n}(x_i - \bar{x})^2/n \) is the sample variance of the docked bikesharing demands.

**III. METHODOLOGY**

**A. DOCKED BIKESHARING SYSTEM BASED GRAPH**

An undirected graph is believed to comprise multiple nodes (vertices) and edges, while a directed graph consists of multiple nodes and arcs. Social network, broadband network, and molecule graphs are common graphs. To represent the docked bikesharing system as a graph, we employ nodes to indicate the locations of docked bikesharing system stations, and the edges to denote the links connecting any two stations. The lengths of edges are calculated according to the shortest distances, instead of the straight-line distances, between any two stations based on non-motorized vehicle lanes, which is the same as that when evaluating spatial dependency. In addition, a docked-bikesharing-system-based graph belongs to undirected graphs because any two stations are connected based on non-motorized vehicle lanes. The docked-bikesharing-system-based graph is characterized by several aspects: 1) no isolated nodes/edges exist, and the docked bikesharing system network generally remains unchanged; 2) the status (pickup/return demands) of each station changes with time; and 3) nodes have meaningful characteristics, including the number of piles, the number of nearby bus/railway stations, and pickup/return demands.

The docked-bikesharing-system-based graph and the relationship among docked bikesharing system stations may be represented as \( G = (V, E) \), where \( V \) denotes \( N \) vertices (stations) with node \( v_i \in V (i = 1, 2, \ldots, N) \); and \( E \) indicates the edges between any two stations, with edge \( e_{ij} \in E (i, j = 1, 2, \ldots, N, i \neq j) \). The docked-bikesharing-system-based graph is distinct from the conventional transportation network graph in terms of the graph structure. Specifically, a node is only connected with its neighborhood nodes in the conventional transportation network, while a station is assumed to connect with all other stations in the docked-bikesharing-system-based graph. To construct an adjacency matrix, we therefore, suppose that a node is adjacent to another node only when their distance is less than a predefined threshold. In this way, the connectivity of nodes in the docked-bikesharing-system-based graph may be denoted by adjacent matrix \( A \in \mathbb{R}^{N \times N} \), where each entry \( A_{ij} = 1 \) \((i,j) = 1, 2, \ldots, N) \) if node \( i \) is adjacent to node \( j \) and \( A_{ij} = 0 \) \((i,j) = 1, 2, \ldots, N) \) otherwise \((A_{ij} = 0, i = 1, 2, \ldots, N) \). According to the adjacent matrix, the degree matrix of the graph based on the docked bikesharing system may be constructed to count the number of adjacent stations connected to a station. The degree matrix is defined as \( D \in \mathbb{R}^{N \times N} \), where \( D_{ii} = \sum_j A_{ij} (i,j = 1, 2, \ldots, N) \). \( D \) is a diagonal matrix, with all non-diagonal entries equal to zero.

**B. DEMAND PREDICTION WITH MULTI-SOURCE DATA**

The prediction of pickup/return demands belongs to a representative time-series problem. In this study, demand prediction is made by accommodating the spatial dependency in a graph based on a docked bikesharing system and the spatial dependency with the historical demand data. To address the problem of the docked bikesharing system demand prediction, the main notations associated with the problem are listed in Table 3.

| 9/5 | 9/6 | 9/7 | 9/8 | 9/9 | 9/10 | 9/11 |
|-----|-----|-----|-----|-----|------|------|
| 0.994 | 0.994 | 0.994 | 0.994 | 0.994 | 0.004 |     |
| 1.000 | 1.000 | 0.994 | 0.902 | 0.012 |     |      |
| 1.000 | 0.994 | 0.902 | 0.068 |     |      |      |
| 1.000 | 0.902 | 0.012 |     |      |      |      |
| 0.902 | 0.004 |     |      |      |      |      |
| 0.902 |     | 0.030 |      |      |      |      |

The prediction of pickup/return demands for the next period according to the demands for the historical \( H \) periods, that is,

\[
\hat{x}_{T+1}^* = \arg \max_{\hat{x}_{T+1}} \log P(\hat{x}_{T+1} | x_T, H, \ldots, x_T, G), \tag{3}
\]

where \( x_T \in \mathbb{R}^n \) denotes the vector of the previous docked bikesharing demands of all stations at the \( T \)-th period, while \( \hat{x}_{T+1}^* \in \mathbb{R}^n \) represents the vector of the predicted docked bikesharing demands of all stations at the \((T+1)\)-th period.
FIGURE 5. The distribution of local Moran’s I indexes for picking up demand.
C. THE LSGC MODEL FOR INDICATING THE SPATIAL DEPENDENCY

CNNs are widely used in various fields, including facial recognition, speech recognition and transportation demand prediction, because they maintain a proper balance between promising modelling performance and acceptable computational burden. However, CNNs are difficult to employ directly in this study because the stations are arranged irregularly, which is distinct from pixels in an image. A feasible approach that uses CNNs in predicting docked bikesharing demands is to introduce spectral graph convolution (SGC), which may consider the historical demand of nearby stations when predicting the demand of a station.

The SGC is constructed according to the Laplacian matrix, which may be derived based on the adjacency matrix \( A \) and degree matrix \( D \). The Laplacian matrix is defined as:

\[
L = D - A.
\]  

(4)

Given that Laplacian matrix \( L \) is a symmetric positive semi-definite matrix, it may be diagonalized by eigen-decomposition as follows:

\[
L = U \Lambda U^T,
\]

(5)

where \( \Lambda \) denotes a diagonal matrix that contains the eigenvalues as entries on the main diagonal, \( U \) comprises all the eigenvectors, and \( U^T \) represents the transpose of \( U \).

SGC is operated by multiplying input \( x_t \in \mathbb{R}^n \) by filter \( f_0 = \text{diag}(\theta) \) with the a vector of parameters \( \theta \in \mathbb{R}^n \). Filter \( \text{diag}(\theta) \) is a diagonalized matrix with \( \theta \) on the main diagonal. The SGC may be defined as:

\[
f_0 * Gx_t = Uf_0U^Tx_t = U\text{diag}(\theta)U^Tx_t,
\]

(6)

where * denotes the operator of the SGC, and filter \( f_0 \) is a vector of parameters that represents the trainable convolutional kernel.

To reduce the computation burden, LSGC is suggested by using polynomial filter \( f_{0^r} = \sum_{i=0}^{K-1} \theta_i^r A^i \) with trainable parameter vector \( \theta^r \in \mathbb{R}^K \), where \( K \) is a predefined hyperparameter. The \( K \)-hop LSGC may be defined as:

\[
f_{0^r} * Gx_t = U \sum_{i=0}^{K-1} \theta_i^r A^i U^T x_t = \sum_{i=0}^{K-1} \theta_i^r L^i x_t.
\]

(7)

Compared with the SGC, the \( K \)-hop LSGC gains advantages from its low computation burden. On one hand, it has only \( K \) parameters to train, while SGC has up to \( N \) parameters to train. On the other hand, the \( K \)-hop LSGC does not require eigen-decomposition, which is a necessary step in SGC. In addition, the \( K \)-hop LSGC makes considering demands of \( K \)-hop adjacent nodes convenient when predicting the demands of a node. \( K \)-hop adjacent nodes refer to nodes that can be reached from a node by taking \( K \) steps based on the adjacency matrix. Therefore, the LSGC is applicable to transportation problems due to the fact that the transportation network is generally large and irregular. Furthermore, if the LSGC is combined with the LSTM, it can be used to deal with the time-series data of transportation fields.

D. THE LSTM MODEL FOR INDICATING THE TEMPORAL DEPENDENCY

Unlike the LSGC structure, the LSTM structure in the LSGC-LSTM model is used to model the temporal dependency for the docked bikesharing demands. In the LSGC-LSTM model, the gate structures are the same as those in the stand-alone LSTM, except that the inputs are replaced by node-level features, which are derived by the operation of the LSGC. Therefore, forget gate \( f_t \), input gate \( i_t \), input cell state \( \tilde{C} \), and output gate \( o_t \) are denoted as:

\[
f_t = \sigma(W_f \cdot GC_t + U_f \cdot h_{t-1} + b_f),
\]

(8)

\[
i_t = \sigma(W_i \cdot GC_t + U_i \cdot h_{t-1} + b_i),
\]

(9)

\[
\tilde{C} = \sigma(W_C \cdot GC_t + U_C \cdot h_{t-1} + b_C),
\]

(10)

\[
o_t = \sigma(W_o \cdot GC_t + U_o \cdot h_{t-1} + b_o),
\]

(11)

where \( \cdot \) denotes the operator of the matrix multiplication. \( W_f, W_i, W_C, \) and \( W_o \in \mathbb{R}^{N \times K} \) are parameter matrices that correspond to the inputs, while \( U_f, U_i, U_C, \) and \( U_o \in \mathbb{R}^{N \times N} \) are parameter matrices that correspond to the hidden states of the previous period for the three gates and the input cell state. Accordingly, \( b_f, b_i, b_C, \) and \( b_o \in \mathbb{R}^N \) are the four bias vectors, while \( \sigma \) is the sigmoid activation function.

Furthermore, the last cell state and the last hidden state are defined as follows:

\[
C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C},
\]

(12)

\[
h_t = o_t \odot \text{tanh}(C_t),
\]

(13)

where \( \odot \) denotes the hyperbolic tangent activation function. At the final period \( T \), hidden state \( h_T \) is taken as the output of the LSGC-LSTM, with predicted demand \( \hat{y}_{T+1} = y_T \). Let \( y_{T+1} \in \mathbb{R}^N \) represent the actual demand at period \( T + 1 \), namely \( y_{T+1} = x_{T+1} \) in the current study. According to predicted and actual demands, the loss value for predicting the demand at the period \( T + 1 \) can be derived as follows:

\[
\text{Loss} = L(y_{T+1}, \hat{y}_{T+1}) = L(x_{T+1}, h_T),
\]

(14)

where \( L(\cdot) \) denotes a function that evaluates the residual between actual demand \( y_{T+1} \) and predicted demand \( \hat{y}_{T+1} \). The \( L(\cdot) \) function may be denoted as various types.

E. THE ARCHITECTURE OF LSGC-LSTM

LSGC-LSTM comprises a LSGC model and a LSTM model (as shown in Figure 6). The LSGC model is used to consider
spatial dependency, while the LSTM model is used to consider the temporal dependency based on the node-level features derived from the LSGC model. Generally, the spatial dependency of the docked bikesharing stations must be considered because pickup/return demands of these stations are correlated. Specifically, nearby stations have similar distributions of pickup/return demands because travelers may walk to a neighboring station to pick up/return a shared bike when they cannot pick up/return a shared bike at a certain station. Notably, LSGC-LSTM integrates LSGC with LSTM in a natural way, rather than combines LSGC and LSTM sequentially. This structure enables LSGC-LSTM to describe the spatial and temporal dependencies simultaneously, contributing to a potential improvement of the prediction accuracy for pickup/return demands.

\[ g(\cdot) \text{ represents the application of the LSGC-LSTM model.} \]

**TABLE 4. The pseudocode of the LSGC-LSTM.**

Inputs: \( x_{T-H+1}, x_{T-H+2}, \ldots, x_T, A \)

Parameters: \( W_f, W_i, W_c, W_o, U_f, U_i, U_c, U_o, b_f, b_i, b_c, b_o, \theta' \)

Initialize: \( h_0 \in \mathbb{R}^N, C_0 \in \mathbb{R}^N \)

Main code:

\begin{align*}
\text{for } t = 1 \text{ to } T \text{ do} \\
& G_C = \text{LSGC}(x_t) \\
& h_t, C_t = \text{LSTM}(G_C, h_{t-1}, C_{t-1}) \\
\text{end for} \\
\text{Return: } h_T
\end{align*}

The pseudocode of the LSGC-LSTM model is shown in Table 4 to represent the proposed model in a more intuitive way. The pseudocode denotes the process of mapping the historical docked bikesharing demands and the adjacency matrix to the predicted demand of the next period. In Table 4, the LSGC function refers to the mapping process of Equations 4-7, while the LSTM function signifies the mapping process of Equations 8-14.

To predict multi-step-ahead time-series demands, we adopt the most intuitive recursive prediction strategy to predict the demands of the next \( N \) periods based on the demands of the historical \( H \) periods. In the recursive strategy, we first predict the first period by using the LSGC-LSTM model. Subsequently, the demand just predicted is taken as a part of the input for predicting the next period with the same one-step model [37]. We continue in this operation until we have predicted all the \( N \) steps. The prediction is described by

\[ \hat{x}_{T+h} = \begin{cases} 
& g(x_T, \ldots, x_{T-H+1}) \quad \text{if } h = 1 \\
& g(\hat{x}_{T+h-1}, \ldots, \hat{x}_{T+1}, x_T, \ldots, x_{T-H+1}) \quad \text{if } h \in \{2, \ldots, H\}, \\
& g(\hat{x}_{T+h-H}, \ldots, \hat{x}_{T+h-H}) \quad \text{if } h \in \{H+1, \ldots, N\}
\end{cases} \]

(15)

where \( g(\cdot) \) represents the application of the LSGC-LSTM model.

**IV. EXPERIMENTAL RESULTS AND ANALYSIS**

**A. EXPERIMENTAL SETTINGS**

All experiments are conducted in a Windows desktop that has an NVIDIA GeForce GTX 1080 GPU with an 8 GB video memory. We take the pickup/return records of the first 20 days, the following five days, and the last five days as the training, validation, and test sets, respectively. All pickup/return records are aggregated within a time step of 5 mins. The models are used to predict the pickup/return demands of each station in the next 5/10/15/20 mins with the multi-step recursive strategy mentioned above. Three commonly used measures are applied to evaluate the models. The measures are the symmetrical mean absolute percent error (SMAPE), the root mean square error (RMSE), and the mean absolute error (MAE). The SMAPE is used because zero demand exists for some stations in a specific time step. The SMAPE is defined as follows:

\[ \text{SMAPE} = \frac{100\%}{qn} \sum_{i=1}^{q} \sum_{j=1}^{n} \frac{|\hat{x}_{ij} - x_{ij}|}{(|x_{ij}| + |\hat{x}_{ij}|)/2}, \]

(16)

where \( q \) denotes the number of predicted samples, \( n \) is the number of stations in the docked bikesharing system, \( x_{ij} \) and \( \hat{x}_{ij} \) indicate the actual and predicted demands for the \( i \)-th sample of the \( j \)-th station, respectively. The RMSE is given
as follows:

$$\text{RMSE} = \sqrt{\frac{1}{qn} \sum_{i=1}^{q} \sum_{j=1}^{n} (x_{ij}^* - x_{ij})^2}.$$  \hfill (17)

The MAE is given as follows:

$$\text{MAE} = \frac{1}{qn} \sum_{i=1}^{q} \sum_{j=1}^{n} |x_{ij}^* - x_{ij}|.$$  \hfill (18)

### B. EXPERIMENTAL RESULTS

We use the commonly used recurrent neural network (RNN), LSTM, gated recurrent unit (GRU), graph attention LSTM network (GAT-LSTM) [22], adaptive graph convolutional recurrent network (AGCRN) [28], and dynamic graph convolutional neural networks (DGCNN) [25] as baseline models. These baselines models are summarized as follows:

- **RNN** The RNN here refers in particular to the “basic RNN,” where each neuron in a given layer is connected with each node in the next layer. Each neuron is valued in a time-varying manner.
- **LSTM** The LSTM is a special type of RNN and is characterized by its capacity of learning long-term dependencies. In accordance with the “basic RNN,” the LSTM has the chain like structure. However, the repeating module has a distinct structure that comprises four neural network layers.
- **GRU** The GRU is also a special type of RNN and similar to the LSTM in the structure. However, the GRU has fewer parameters than the LSTM because it does not have an output gate.
- **GAT-LSTM** The GAT-LSTM extends the LSTM to incorporate the graph attention structure in the input-to-state and state-to-state transitions. Similar to LSTM, GAT-LSTM can also be used as a block to construct a more complex structure.
- **AGCRN** The AGCRN uses two adaptive modules to avoid the pre-defined graph. The two modules consist of a node adaptive parameter learning module and a data adaptive graph generation module.
- **DGCNN** The DGCNN includes tensor decomposition into the deep learning framework by introducing two components. The two components consist of a global component that describes long-term spatio-temporal relationship and a local component that represents the traffic fluctuations.

The initial learning rate is set as $10^{-3}$ and a decay rate is predefined as 0.7 for every five steps. Convergence is achieved when a gap is less than $10^{-3}$ between two successive epochs for up to 10 epochs. The hyperparameter optimization for the LSGC-LSTM is shown in Table 5. The best number of previous time steps is 12 and the best batch size is 60.

On the basis of all the measures of 5/10/15/20 mins in Tables 6 and 7, the performance of LSGC-LSTM is better than those of the baselines models in predicting pickup/return demands. The results indicate that the spatial dependency must be considered in the model. The global/local Moran’s I indexes indicate that pickup/return demands of each station are associated with those of the nearby stations. High pickup/return demands arise in the old urban area, further improving the spatial dependencies. The three measures are inconsistent for distinct models. This discrepancy may be because the SMAPE denotes the relative difference between the predicted and actual demands, but RMSE and MAE denote the absolute difference based on the L1 and L2 norms. As expected, the prediction accuracy is highest for 5 mins, followed by those for 10, 15, and 20 mins.

The prediction performance of the return demand outperforms that of the pickup demand based on most measures. This phenomenon may be attributed to the larger variance of
C. EXPERIMENTAL ANALYSIS

The LSGC-LSTM achieves better prediction performance and has greater efficiency compared with the baselines possibly due to the following reasons. Although the RNN maintains a promising accuracy for the one-step prediction, it fails to work well for the multi-step prediction. The LSTM could solve the problem of vanishing gradients, but it requires a lot of time to get trained. The GRU is less complex than the LSTM, but it may be not as good as the LSTM for a large-scale dataset. The GAT-LSTM has a great learning capacity, but it may need more labels besides the actual data decreases to approximately 0.8, which is much smaller than those of the baseline models.

The RMSE of the test data with respect to the training burden is shown in Figure 7 to compare the computation burden between the LSGC-LSTM and baseline models. LSGC-LSTM achieves the fastest convergence. After the LSGC-LSTM is trained for 10 seconds, the RMSE of the test data decreases to approximately 0.8, which is much smaller than those of the baseline models.

![FIGURE 8. The variation of SMAPE with respect to time for the test dataset from September 26th to September 30th.](image)
Generally speaking, the prediction accuracies decline rapidly for all models except the LSGC-LSTM and the DGCNN as the number of time steps predicted increases. The RNN has a promising prediction accuracy for the one-step prediction, but it cannot predict pickup/return demands for the multi-step situation. The DGCNN makes an effective multi-step prediction possibly because it uses a global component to capture the long-term spatio-temporal demand relationship among nearby nodes. The accurate prediction of the LSGC-LSTM for the multi-step demand may be attributed to its relatively simple structure, which contributes to the avoidance of the over-fitting.

Although we use the multi-source data to predict pickup/return demands, it is necessary to apply more kinds of data in the big-data era. For example, the weather and the traffic congestion data may be useful for predicting pickup/return demands. When it suddenly begins to rain, the pickup demand may drop precipitously. When the traffic congestion occurs, more travelers may choose to pick up shared bikes to make commuting trips for avoiding arriving late. If these data are included in the data source, the accuracy for the one-step or multi-step prediction is expected to improve.

Another interesting finding is that all the measures (SMAPE, RMSE, and MAE) have large values during peak hours and small values otherwise (as shown in Figures 8, 9, and 10), respectively. In addition, the distributions of all the measures in terms of time are similar in different days. This finding further indicates that great attention should be paid to the peak hours to supervise the training process to avoid the over-fitting. Although the AGCRN can avoid the pre-defined graph to improve the prediction accuracy, it is more suitable for making prediction when nodes have diversified patterns in pickup/return demands. The DGCNN can accommodate time-varying spatial dependencies, but it may achieve better prediction performance for the prediction with a longer time span.
should be paid to the accurate prediction of pickup/return demands in the peak hours. In other words, the prediction accuracy may be improved significantly by predicting pickup/return demands during peak hours more accurately, providing an opportunity to improve the profit of bikesharing companies and the satisfaction of cyclists.

The peak values of the three measures have different characteristics. Specifically, the peak values of the SMAPE are similar across different days, while those of the RMSE and MAE are similar. This phenomenon indicates that the SMAPE becomes insensitive to the change in predicted pickup/return demands when its value is large, while the RMSE and the MAE remain sensitive to the change in predicted pickup/return demands even if their values are large.

V. CONCLUSION AND FUTURE DIRECTIONS

On the basis of multi-source data, we develop the LSGC-LSTM model for predicting pickup/return demands of 186 stations for the docked bikesharing program. A promising result is obtained in terms of accuracy and computation burden compared to those of the baseline models. The result indicates that spatial dependency must be considered in predicting pickup/return demands for docked bikesharing systems. LSGC-LSTM may be used to predict demands in other transportation scenarios, including the ridership of each bus stop/subway station, pickup/return demands of each carsharing station, and the number of travelers for each scenic spot. The main limitation of the LSGC-LSTM lies in that it may not be suitable for simultaneously predicting multiple types of demands (e.g., pickup and return demands) and predicting demands of a long period (e.g., demands in a day).

Further research may be conducted from the following aspects. Firstly, more temporal and spatial influencing factors need to be included in the model. The temporal influencing factors include the weather, the temperature (or the apparent temperature), the relative humidity, the air quality index, the wind power, the precipitation, the ultraviolet intensity, and the day of the week (i.e., working day or not). The spatial influencing factors include the population density, the population constitution, the car ownership, the e-bike ownership, the private bike ownership, the percentage of the working population, the land use style, the public transit infrastructure, and the distribution of points of interest in a region around a docked bikesharing station. The scope of each region is generally determined with Thiessen polygons in GIS softwares including ArcGIS.

Secondly, more datasets that are generated from other cities and at other times need to be used for to test the model. In this study, we only use a dataset from a city in a specific time period to train and test the proposed model and baselines. Although a promising result is achieved in terms of the prediction accuracy, the temporal and spatial transferability of the model need to be further validated because the parameters of the model are characterized by the temporal and spatial distinctive features of the specific dataset. Therefore, it contributes to the improvement of the model in terms of temporal and spatial transferability.

Thirdly, more up-to-date models need to be used to predict pickup/return demands. For example, a reinforcement learning approach proves to be a promising approach in analyzing time-series data [38]. It is also possible to use the reinforce learning approach to improve the prediction accuracy if the reward is designed effectively.

ACKNOWLEDGMENT

The authors thank Anming Zhang and Zhengyi Zhou for their helpful suggestions.

REFERENCES

[1] C. Zhang, Y. Liu, W. Lu, and G. Xiao, “Evaluating passenger satisfaction index based on PLS-SEM model: Evidence from chinese public transport service,” Transp. Res. A, Policy Pract., vol. 120, pp. 149–164, Feb. 2019.
[2] F. Zong, Y. Tian, Y. He, J. Tang, and J. Lv, “Trip destination prediction based on multi-day GPS data,” Phys. A, Stat. Mech. Appl., vol. 515, pp. 258–269, Feb. 2019.
[3] T. F. Welch, S. R. Gehrike, and A. Widita, “Shared-use mobility competition: A trip-level analysis of taxi, bikeshare, and transit mode choice in Washington, DC,” Transportmetrica A, Transp. Sci., vol. 16, no. 1, pp. 43–55, Dec. 2020.
[4] L. Xu, C. Wang, Z. Miao, and J. Chen, “Governmental subsidy policies and supply chain decisions with carbon emission limit and consumer’s environmental awareness,” RAIRO Oper. Res., vol. 53, no. 5, pp. 1675–1689, Nov. 2019.
[5] G. Xiao and Z. Wang, “Empirical study on bikesharing brand selection in China in the post-sharing era,” Sustainability, vol. 12, no. 8, p. 3125, Apr. 2020.
[6] M. Chen, D. Wang, Y. Sun, E. O. D. Waygood, and W. Yang, “A comparison of users’ characteristics between station-based bikesharing system and free-floating bikesharing system: Case study in Hangzhou, China,” Transportation, vol. 47, no. 2, pp. 689–704, Apr. 2020.
[7] K. B. Campbell and C. Brakewood, “Sharing riders: How bikesharing impacts bus ridership in New York City,” Transp. Res. A, Policy Pract., vol. 100, pp. 264–282, Jun. 2017.
[8] G. Xiao, R. Wang, C. Zhang, and A. Ni, “Demand prediction for a public bike sharing program based on spatio-temporal graph convolutional networks,” Multimedia Tools Appl., pp. 1–19, Mar. 2020.
[9] P. Lin, J. Weng, Q. Liang, D. Alivanistos, and S. Ma, “Impact of weather conditions and built environment on public bikesharing trips in Beijing,” Netw. Spatial Econ., vol. 20, no. 1, pp. 1–17, Mar. 2020.
[10] X. Ma, X. Zhang, X. Li, X. Wang, and X. Zhao, “Impacts of free-floating bikesharing system on public transit ridership,” Transp. Res. D, Transp. Environ., vol. 76, pp. 100–110, Nov. 2019.
[11] J. Tang, F. Liu, W. Zhang, R. Ke, and Y. Zou, “Lane-changes prediction based on adaptive fuzzy neural network,” Expert Syst. Appl., vol. 91, pp. 452–463, Jan. 2018.
[12] L. Lin, Z. He, and S. Peeta, “Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach,” Transp. Res. C, Emerg. Technol., vol. 97, pp. 258–276, Dec. 2018.
[13] B. Darren and B. Ralph, “Bike lanes and other determinants of capital bikeshare trips,” in Proc. 91st Transp. Res. Board Annua Meeting, 2012, pp. 1–11.
[14] X. Wang, G. Lindsey, J. E. Schoner, and A. Harrison, “Modeling bike share station activity: Effects of nearby businesses and jobs on trips to and from stations,” J. Urban Planning Develop., vol. 142, no. 1, Mar. 2016, Art. no. 04015001.
[15] K. Gebhart and R. B. Noland, “The impact of weather conditions on bikeshare trips in Washington, DC,” Transportation, vol. 41, no. 6, pp. 1205–1225, Nov. 2014.
[16] C. Rudloff and B. Lackner, “Modeling demand for bikesharing systems: Neighboring stations as source for demand and reason for structural breaks,” Transp. Res. Rec., J. Transp. Res. Board, vol. 2430, no. 1, pp. 1–11, Jan. 2014.
[17] Y. Zhang, T. Thomas, M. Brussel, and M. van Maarseveen, “Exploring the impact of built environment factors on the use of public bikes at bike stations: Case study in Zhongshan, China,” J. Transp. Geography, vol. 58, pp. 59–70, Jan. 2017.

[18] J. Bao, X. Shi, and H. Zhang, “Spatial analysis of bikeshare ridership with smart card and POI data using geographically weighted regression method,” IEEE Access, vol. 6, pp. 76049–76059, 2018.

[19] Y. Shen, X. Zhang, and J. Zhao, “Understanding the usage of dockless bike sharing in Singapore,” Int. J. Sustain. Transp., vol. 12, no. 9, pp. 686–700, Oct. 2018.

[20] K. Chen, F. Chen, B. Lai, Z. Jin, Y. Liu, K. Li, L. Wei, P. Wang, Y. Tang, J. Huang, and X.-S. Hua, “Dynamic spatio-temporal graph-based CNNs for traffic prediction,” 2018, arXiv:1812.02019. [Online]. Available: http://arxiv.org/abs/1812.02019

[21] B. Yu, H. Yin, and Z. Zhu, “Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting,” J. Comput. Sci. Technol., vol. 32, no. 4, pp. 625–640, Jul. 2017.

[22] Z. Chen, X. Wang, D. Zhang, Y. Liu, K. Xie, and S. He, “Dynamic spatial-temporal graph convolutional neural networks for traffic forecasting,” in Proc. AAAI Conf. Artif. Intell., vol. 33, 2019, pp. 890–897.

[23] A. Micheli, “Neural network for graphs: A contextual constructive approach,” IEEE Trans. Neural Netw., vol. 20, no. 3, pp. 498–511, Mar. 2009.

[24] F. Scarselli, M. Gori, A. Chung Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” IEEE Trans. Neural Netw., vol. 20, no. 1, pp. 61–80, Jan. 2009.

[25] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, “Adaptive graph convolutional recurrent network for traffic forecasting,” 2020, arXiv:2007.02842. [Online]. Available: http://arxiv.org/abs/2007.02842

[26] 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, Jul. 2020.

[27] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs,” 2013, arXiv:1312.6203. [Online]. Available: https://arxiv.org/abs/1312.6203

[28] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” 2016, arXiv:1609.02907. [Online]. Available: http://arxiv.org/abs/1609.02907

[29] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “Graph neural networks: A review of methods and applications,” 2018, arXiv:1812.08434. [Online]. Available: http://arxiv.org/abs/1812.08434

[30] B. Yu, H. Yin, and Z. Zhu, “Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting,” in Proc. Int. Joint Conf. Artif. Intell., 2018, pp. 3634–3640.

[31] J. Zhang, F. Chen, Y. Guo, and X. Li, “Multi-graph convolutional network for short-term passenger flow forecasting in urban rail transit,” IET Intell. Transp. Syst., vol. 14, no. 10, pp. 1210–1217, Oct. 2020.

[32] S. Wang, J. Cao, and P. Yu, “Deep learning for spatio-temporal data mining: A survey,” IEEE Trans. Knowl. Data Eng., early access, Sep. 22, 2020, doi: 10.1109/TKDE.2020.3025580.

[33] W. R. Tobler, “A computer movie simulating urban growth in the detroit region,” Econ. Geogr., vol. 46, pp. 234–240, Jun. 1970.

[34] S. Ben Taieb, G. Bontempi, A. F. Atiya, and A. Sorjomaa, “A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition,” Expert Syst. Appl., vol. 39, no. 8, pp. 7067–7083, Jun. 2012.

[35] T. Kuremont, T. Hirata, M. Obayashi, S. Mabu, and K. Kobayashi, “Training deep neural networks with reinforcement learning for time series forecasting,” in Time Series Analysis-Data, Methods, and Applications, Rijeka, Croatia: InTech, 2019.