Station Correlation Attention Learning for Data-driven Bike Sharing System Usage Prediction

Xi Yang
The University of Connecticut
xi.yang@uconn.edu

Suining He
The University of Connecticut
suining.he@uconn.edu

Huiqun Huang
The University of Connecticut
huiqun.huang@uconn.edu

***Abstract*—After years of development, bike sharing has been one of the major choices of transportation for urban residents worldwide. However, efficient use of bike sharing resources is challenging due to the unbalanced station-level demands and supplies, which causes the maintenance of the bike sharing systems painstaking. To achieve system efficiency, efforts have been made on accurate prediction of bike traffic (demands/pick-ups and returns/drop-offs). Nonetheless, bike station traffic prediction is difficult due to the spatio-temporal complexity of bike sharing systems. Moreover, such level of prediction over the entire bike sharing systems is also challenging due to the large number of bike stations.**

To fill this gap, we propose `BikeGAAN`, a graph adjacency attention neural network to predict station-level bike traffic for entire bike sharing systems. The proposed prediction system consists of a graph convolutional network with an attention mechanism differentiating the spatial correlations between features of bike stations in the system and a long short-term memory network capturing temporal correlations. We have conducted extensive data analysis upon bike usage, weather, points of interest and event data, and derived the graph representation of the bike sharing networks. Through experimental study on over 27 millions trips of bike sharing systems of four metropolitan cities in the U.S., New York City, Chicago, Washington D.C. and Los Angeles, our network design has shown high accuracy in predicting the bike station traffic in the cities, outperforming other baselines and state-of-art models.

***Index Terms*—bike sharing, pick-up and drop-off, spatio-temporal, data-driven, station-based traffic prediction, graph convolutional network, adjacency attention, data analysis**

I. Introduction

Bike sharing has become one of the major mobility options for urban residents worldwide due to its advantages in convenience and economy over other means of urban transportation. As a representative product of the sharing economy, it is often hailed as a good helper to solve the "last-mile" problem in citizen transportation, enhancing connectivities across various city zones. Due to the social and business importance, the global bike sharing market is estimated to hit $5 billion by 2025[1].

A docked/station-based bike sharing system is usually operated in the way that a user picks up a bike from one station (demand) and drops it off at another station (return), naturally forming a *bike sharing network*, where each node or vertex of the network represents a bike station. The graph edges can be weighted based on the *correlations* between the stations, which can be further represented by the *weighted adjancy matrix* in the graph theory. Such correlations are usually characterized by the trips resultant from the stations' neighboring spatial and temporal features. A trip from one station to another is generated mainly because of the riders' diverse interests or preferences in the origin and the destination at a certain time period as shown in Fig. 1. For example, during the rush hours of a workday morning, people tend to leave their homes for work places, leading to high bike usage volume at stations near residential (Station A) and office areas (Stations B, C and D) of the city. Therefore, the spatio-temporal correlations between stations are an important aspect to describe the bike usage behavior. However, the correlations between stations can be highly complex due to the neighborhood spatio-temporal characteristics. It is extremely challenging in identifying and differentiating the importance of the complex correlations between stations within the mobility modeling. Furthermore, accurate prediction with respect to each individual station, is essential but challenging for rebalancing the demands (pick-ups) and supplies (drop-offs) at the bike sharing systems, which is the pressing issue for many city planners.

Fig. 1: Illustration of bike sharing station networks.

To address the above concerns and enable the data-driven rebalancing, in this work we propose a Bike sharing Graph Adjacency Attention Network called `BikeGAAN` for station-based pick-up and drop-off prediction with historical spatial and temporal traffic features and some external related factors including weather conditions, weekends/holidays, points

[1]https://www.globenewswire.com/news-release/2019/12/03/1955257/0/en/Bike-Sharing-Market-is-Predicted-to-Hit-5-Billion-by-2025-P-S-Intelligence.html

2155-6814/20/$31.00 ©2020 IEEE
DOI 10.1109/MASS50613.2020.00083

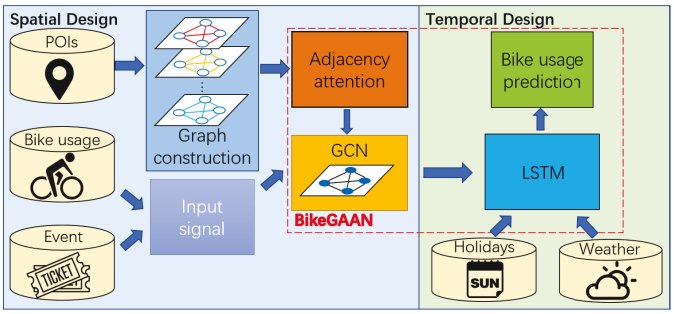


Fig. 2: System overview and information flow of BikeGAAN.

of interest (POIs) and events. We formulate the differences between stations' surrounding POI distributions into station correlations, represented by the adjacency matrix of the graph representation of the bike sharing networks, and develop an adjacency attention mechanism differentiating the importance of each category of POIs. We incorporate the spatio-temporal characteristics of the bike sharing networks via fusion of graph convolutional network (GCN) and long short-term memory (LSTM), leading to high prediction accuracy.

Specifically, our main contributions are as follows:

1) *Comprehensive Bike Data Analysis*: To identify the complex correlations between stations, we have conducted a comprehensive and detailed real-world data analysis on how the weather conditions, events, weekends/holidays, station locations and surrounding POIs impact the bike usage in four metropolitan cities of New York City (NYC) NY, Chicago IL, Washington D.C. (DC) and Los Angeles (LA) CA, and visualized them to validate our model design insights.
2) *Station Correlation Attention Model*: To better differentiate the correlations across stations, we propose a novel *adjacency matrix attention* mechanism for the graph convolutional network (GCN) which flexbily determines the contributions of different spatial characteristics upon the spatial correlations between bike stations. We study and design the station graph adjacency based on the stations' neighborhood features in terms of POI distributions. The proposed model accurately predicts the station-level bike usage (pick-ups and drop-offs) of a large number of stations.
3) *Extensive Experimental Studies*: We have conducted extensive experimental studies upon over 27 millions trips in total from totally 1,122 stations of the four metropolitan bike sharing systems in the U.S.: Citi Bike in NYC, Divvy in Chicago, Capital Bikeshare in DC, and Metro Bike in LA. The experimental studies have shown that our model outperforms the baselines and state-of-art models in multi-station prediction.

Our proposed system provides the accurate prediction results for bike sharing system operators to efficiently rebalance their bike distributions, and can be further integrated with mobile apps or interactive bike websites (say, like Citi Bike Station Map[2]) to inform the citizens regarding predicted bike availability in the next few hours/minutes to better plan their rides.

We further overview the system framework of BikeGAAN in Fig. 2. The graph representation of the bike stations is first constructed based on the spatial features (say, POIs distributions) around stations. An adjacency attention mechanism further captures the correlations between stations and differentiates the importance of each category of POIs. The graph convolutional network takes in the historical bike usage data and nearby events around stations as inputs, and the long short-term memory network further learns the temporal characteristics of the bike sharing networks with the aid of external features including weather and weekends/holidays. While our work focuses upon the bike sharing system studies, our model designs including the adjacency attention mechanism and POI differences can be easily extended to other domains in the smart cities and transportation problems, including ride sharing [1] and crowd flow prediction [2].

The rest of the paper is organized as follows. We first review the related work in Section II. After that, we present the data analysis in Section III. Given the derived features, we then present the core model of our approach in Section IV. We show the performance evaluation in Section V. We discuss the deployment in Section VI and conclude in Section VII.

II. Related Work

To achieve efficient operation of the bike sharing systems, efforts have been made on accurate prediction of bike usage. Based on prediction granularity, there are three categories of prediction models in current works: *city-level*, *cluster-level*, and *station-level* prediction [3]. In *city-level* prediction, the aim is to predict bike usage for an entire city [4]. Towards finer prediction granularity, researchers have studied the *cluster-level* prediction to predict bike usage for clusters of bike stations [5]–[7].

While city-level and cluster-level predictions save the computational cost by simplifying the problems, station-level prediction still benefits the bike sharing system management the most due to the granularity for each individual station. However, the station-level prediction is challenging due to the spatio-temporal characteristics of the bike sharing usage patterns of a large number of stations in a city. Previous studies on station-level bike usage prediction have taken into account the impact of POIs on bike mobility patterns [3], [8]–[11], but few of them have considered the difference between the impact of different POI categories. Therefore, in this study we focus on station-level bike usage prediction, where we formulate the differences between POIs distributions around stations into their correlations, represented by the adjacency matrix of the graph representation of the bike sharing networks, and develop an attention mechanism on the station correlations. Attention mechanisms have been widely used to improve the performance of deep learning models [12]–[20]. Different from the prior studies of graph attention networks [21]–[24], based on extensive data analysis, we design a novel graph adjacency attention mechanism differentiating the complex

[2]https://member.citibikenyc.com/map/

TABLE I: List of symbols and definitions.

Symbols	Definitions
N	Total number of stations
i, k	Indices of the stations
s	Number of features for each station
t	Index of timestamps
$\mathbf{z}$	Features of a station
z	Index of station features
j	Index of the features
p, d	Pick-ups and drop-offs
$\mathcal{T}, \tau$	Historical and future timestamps
$y, \hat{y}$	Ground truth and predictions of bike usage
G, V, E	Graph, graph nodes and graph edges
e	The edge weight of graph G
$\mathcal{G}_{lat}, \mathcal{G}_{lon}$	Grid dimensions around each station
C	The number of POI categories
c	Index of POI categories
$\mathbf{p}_i$	POIs around station i
$\mathbf{A}, \tilde{\mathbf{A}}$	Adjacency matrix and normalized adjacency matrix
$\mathbf{D}$	Degree matrix of adjacency matrix
L	The number of layers of GCN
l	Index of GCN layers
$\mathbf{ext}, w$	External features and the number of them
$\mathbf{v}, \mathbf{W}, \mathbf{U}, \mathbf{b}$	Trainable parameters
$\mathbf{h}$	LSTM hidden state
$\mathbf{c}$	LSTM cell state
h	Number of hidden units of LSTM
$\hat{\mathbf{A}}, \tilde{\mathbf{A}}$	Intermediate and weighted adjacency matrices
ϵ, α	Attention score and attention weight

TABLE II: Summary of datasets used for the four cities.

Data	Citi	Divvy	Capital	Metro
User trips	20,551,697	3,113,950	3,398,417	290,342
Weather	8,760	8,760	8,760	8,760
POIs	14,615	3,976	13,227	3,323
Events	3,662	10,442	1,588	4,149

station correlations, specifically the POI categories, within bike sharing networks.

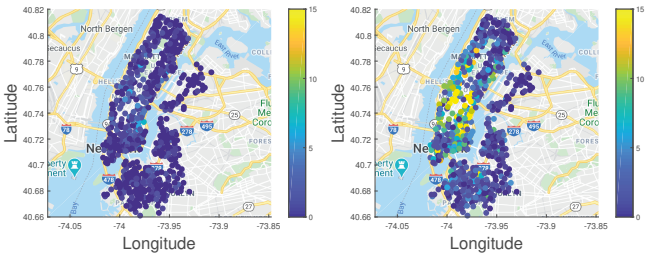
III. Bike Sharing System Data Analysis

We first overview the datasets in Sec. III-A and then present the data analysis of the datasets in Sec. III-B. Table I summarizes the symbols and their definitions presented in this work.

A. Data Overview

We use four types of datasets in this work: the user trip data, the weather condition data, POI data and event data. The number of data points collected for the four cities are listed in Table II. Details of each dataset are discussed in the following.

1) *User Trip Data*: The collected trip data describe every single trip including the trip's duration, start/end time, start/end stations and their longitude/latitude, and user information. We have collected totally 27,354,406 trips from the four cities in 2019: 20,551,697 trips from Citi Bike in NYC, 3,113,950 trips from Divvy Bike in Chicago, 3,398,417 trips from Capital Bikeshare in DC, and 290,342 trips from Metro Bike in LA.
2) *Weather Condition Data*: To characterize the hourly weather conditions, we select the hourly temperature, precipitation and wind speed from open source weather data API[3] for the four cities.
3) *POI Data*: We have collected totally 35,141 POIs for all four cities through the OpenStreetMap amenity keys[4]. The amenity keys fall into 7 major categories, including sustenance, education, transportation, financial, healthcare, entertainment arts & culture and others. The major types contain a total of 107 minor types (the amenity keys).
4) *Event Data*: We have collected totally 18,253 events of all 27 categories such as music, festivals, etc. in 2019 for the four major cities from open datasets[5], including 3,662 from NYC, 10,442 from Chicago, 4,149 from LA and 1,588 from DC. The event data contain the GPS coordinates of event locations and the starting time of events.



(a) Sunday morning **(b)** Monday morning

Fig. 3: Heatmaps of Citi Bike system of NYC on (a) Sunday 01-20-2019 from 8:00 a.m. to 9:00 a.m.; (b) Monday 01-21-2019 from 8:00 a.m. to 9:00 a.m. The colors of the nodes represent the bike pick-up volumes of the stations (the warmer colors indicate more demands).

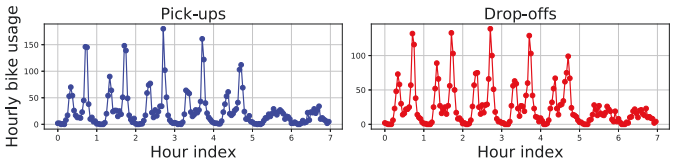


Fig. 4: Hourly bike pick-ups and drop-offs from Monday June 3rd to Sunday June 9th of a station in NYC (40.751873°N, 73.977706°W).

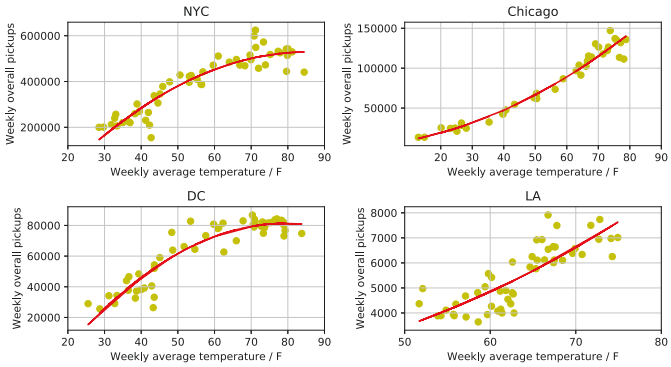


Fig. 5: Weekly overall bike usage as a function of weekly average temperature.

B. Spatio-Temporal Analysis

Given the above datasets, we present the following insights regarding effects of weather, POIs, weekends/holidays and

[3]https://api.weather.com
[4]https://wiki.openstreetmap.org/wiki/Key:amenity
[5]https://api.eventful.com/

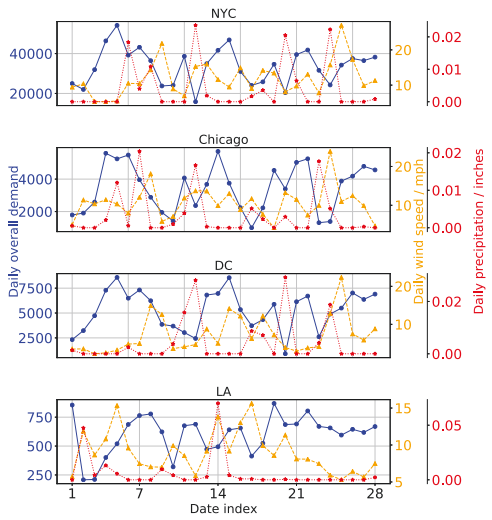


Fig. 6: Daily overall bike usage compared with daily average precipitation and wind speed in February for the four cities.

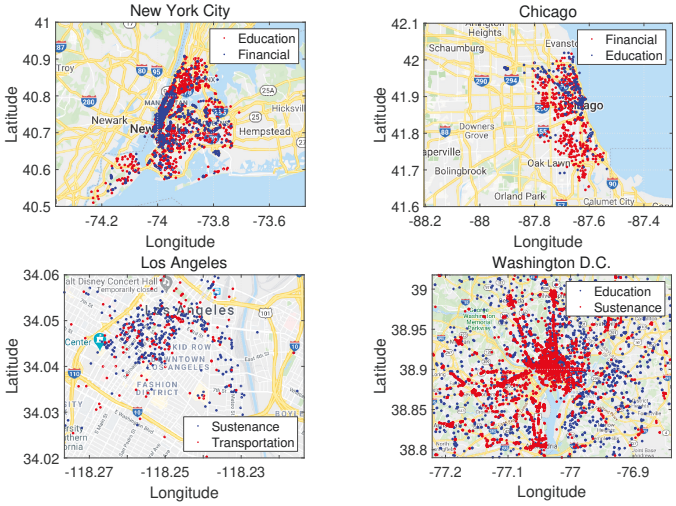


Fig. 7: Distributions of different categories of POIs in the four major cities. Only two types of POIs are shown here for a clearer visualization.

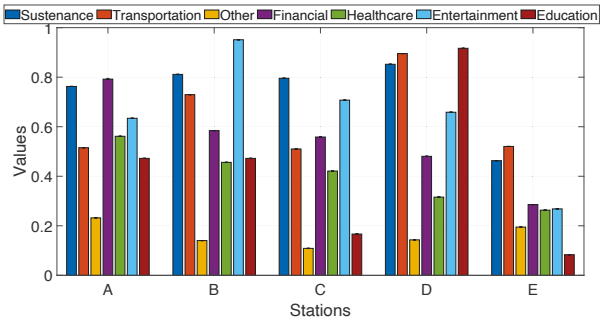


Fig. 8: POI vectors (normalized) at five stations in NYC.

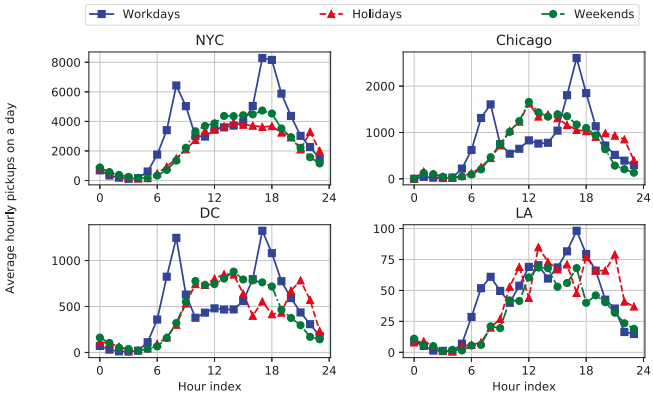


Fig. 9: Averaged hourly pickups of the whole sharing systems on workdays (the week before July 4th, 2019), holidays (July 4th, 2019), and weekends (the weekends after July 4th, 2019) for the four cities.

events upon the station traffic based on the data and visualization.

1) Overview of Spatial and Temporal Traffic: The bike usage pattern of a sharing system shows a strong spatio-temporal characteristic as illustrated in Figs. 3 and 4. As shown in Fig. 3 the stations at the center of Manhattan, NYC have higher bike demands than the rest of the city, while the bike demands of the whole system on the Monday morning is much higher than that on Sunday morning. The temporal characteristic is illustrated in Fig. 4, from which we can see that the temporal pattern of bike usage is different between workdays and weekends. Therefore, in this work we consider such spatial and temporal characteristics in our model design.

2) Station Traffic and Weather: The bike usage is highly correlated with weather conditions. The three dominating effects are *temperature*, *precipitation* and *wind speed*. Fig. 5 shows the relationship between the hourly user trip volume of the entire city and temperature for the four cities. While the temperature's impact on hourly bike usage is not significant for a short-term period, its long-term influence cannot be ignored. Fig. 6 further presents the correlations between pickups and the precipitation and wind speed. In general, precipitation and wind speed negatively impact the number of bike usage. A notable decrease in bike usage will happen when precipitation or high wind speed occurs. Given above, we take these three factors into account and feed them to the deep learning model as the external features.

3) Station Traffic and POIs: As discussed in Sec. III-B1, the bike usage of stations is highly location dependent. One of the major reasons behind this is that different locations have different distributions of POIs. Fig. 7 illustrates this clearly where we compare the distributions of two categories of POIs for the four cities for visualization. For example, the financial POIs are mainly located at the center of Manhattan, surrounded by educational facilities. The different distributions of each type of POIs result in the dependency of bike usage on locations. Such distribution differences represent the spatial correlations between stations. It should be noted that those correlations are not static and may change with time. In addition, different categories of POIs have different contributions to the station correlations. Therefore, we take into account the distributions of the POIs within a $\mathcal{G}_{lat} \times \mathcal{G}_{lon}$ meters grid of each station as input features, and the dynamic of the POIs correlations and the differences in the significance of each POIs type are included in our model design.

To further characterize the neighborhood environment, we form a POI vector for each bike station. Specifically, we denote

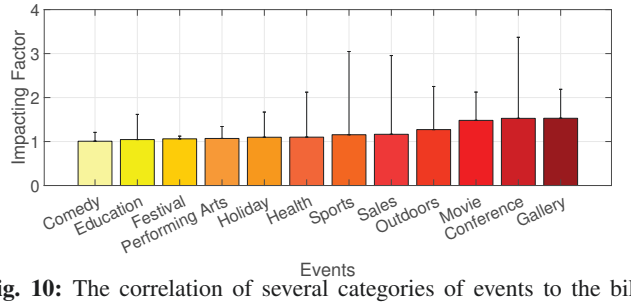


Fig. 10: The correlation of several categories of events to the bike usage of NYC.

the numbers of C different POI categories in a $\mathcal{G}_{lat} \times \mathcal{G}_{lon}$ grid centered at station i as a vector $\mathbf{p}_i$. The c-th element of the vector, $\mathbf{p}_i[c]$, is the number of POIs for category c in the grid area for station i. We further show in Fig. 8 the various POI vectors (normalized POI number with respect to each category) of five bike stations with the highest bike usage in NYC. We can observe diverse POI numbers for these stations, and our formulation in `BikeGAAN` will incorporate such features.

4) Station Traffic and Weekends/Holidays: As shown in Fig. 9, the bike usage for a station on a workday has a different mobility pattern from that on weekends/federal holidays. Therefore, we consider weekends/federal holidays as an external factor. Specifically, we set an indicator as 1 if the usage time is on weekends/holidays and 0 otherwise.

5) Station Traffic and Events: Event is another important contribution to the bike usage in the systems [25]. To analyze the influence of the occurrence of different types of events on the station bike usage, we calculate the hourly frequency of each type of event around each station within 2019. Considering each station as the center, we find a square grid with a side length of 500 meters for each station. Then we find the number of events in different categories within the grid to evaluate their influences upon the bike usage.

Specifically, we find the average hourly bike usage within a range from one hour before to one hour after a certain category of event (say, gallery), and those bike usage in the same hours of a day without that event. We respectively aggregate these two kinds of usage, and find their ratios afterwards as their impact factors. Fig. 10 shows the mean impact factors as well as their standard deviations of 12 different categories of events on the bike usage of NYC. Clearly, the larger impact factors indicate more bike usage given the events nearby, and we can observe that *gallery*, *conference* and *movie* are the three most influential events. In this work we take into account the events as additional features for the stations.

In summary, based on the analysis above, besides the bike usage we use nearby POIs and events as the features of each bike station. Weather conditions including temperature, wind speed and precipitation, as well as weekends/holidays or not are used as the external features. As an example, for the time interval of [0:00 a.m., 1:00 a.m.], 2019-01-01, the external vector is given by [47 °F, 1.5 mph, 0.08 in, 1] (1 for holiday).

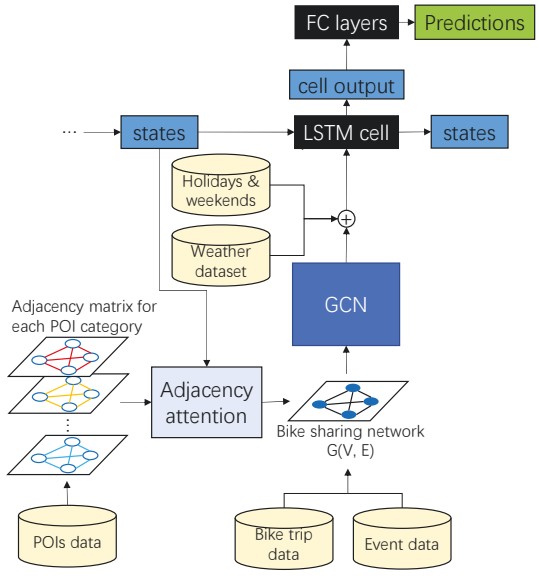


Fig. 11: Illustration of model designs for `BikeGAAN` at the last timestamp where the cell output is connected to a fully-connected (FC) layer to generate prediction. There are no such FC layers for the previous timestamps.

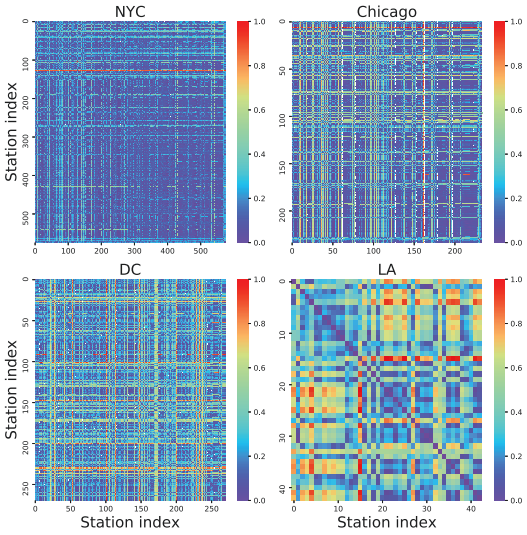


Fig. 12: Illustration of adjacency matrices (after min-max normalization) at initial timestamp for the four bike sharing networks.

IV. `BikeGAAN`: Graph Adjacency Attention For Bike Station Traffic Prediction

Given the above data analysis, we first describe definitions of the problem we study in Sec. IV-A, and then give detailed descriptions about `BikeGAAN` design in Sec. IV-B.

A. Problem Definition

We consider that for N stations in a city, each station i has s features at timestamp t:

$$\mathbf{z}_{i,t} = (z_{i1,t}, ..., z_{ij,t}, ..., z_{is,t}), \tag{1}$$

where $i \in [1, ..., N]$ and $j \in [1, ..., s]$. Let the first two features, $z^{(p)}_{i1,t}$ and $z^{(d)}_{i2,t}$, be the bike usage (pick-ups and drop-offs) of station i at time t, while the rest of the features be the number

of events around stations. The problem of this work is defined as: given the features of each station of previous $\mathcal{T}$ timestamps,

$$\mathbf{Z}_{\mathcal{T}} = (\mathbf{z}_{1,\mathcal{T}}, \ldots, \mathbf{z}_{i,\mathcal{T}}, \ldots, \mathbf{z}_{N,\mathcal{T}}), \quad i \in [1, ..., N], \tag{2}$$

we aim at predicting the bike usage, $\hat{y}^{(p)}_{i,\tau}$ and $\hat{y}^{(d)}_{i,\tau}$ for $i \in [1, ..., N]$, at the next τ timestamps in the bike sharing system.

To approach the above problem, we have `BikeGAAN`, a data-driven bike usage prediction system, and the core structure is shown in Fig. 11. The details of each component will be discussed in the following sections.

B. Detailed Formulations in `BikeGAAN`

We further present the detailed designs in `BikeGAAN` including graph generation, graph convolutional networks, long short-term memory and adjacency attention.

1) Graph Generation: A bike sharing network can be encoded in a graph $G(V, E)$ where each node denotes each bike station and the weight of each edge denotes the correlation between any two stations. We note that the designs of the correlations, characterized by the weighted adjacency matrix, are particularly important for the prediction via graph convolutional networks. The mobility trends of bike usage can be described by the dynamic difference in the POIs between stations, and hence we define the weight of the edge between any two stations as the weighted Euclidean distance between the different numbers of POIs in their surrounding areas.

Recall that we form the POI vector at each station i as $\mathbf{p}_i$. Then we design the weight of the edge between stations i and k, $e_{i,k}$, at the initial timestamp as

$$e_{i,k} = \sqrt{\sum_{c=1}^{C} \left(\mathbf{p}_i[c] - \mathbf{p}_k[c]\right)^2}. \tag{3}$$

Then we form the weighted adjacency matrix between the stations as $\mathbf{A}$, and $\mathbf{A}[i,k] = \mathbf{A}[k,i] = e_{i,k}$.

From previous study [26], we notice that most of the bike trips cover geo-distances over 500 meters, meaning that there is limited bike communication between two close stations that have similar POI distributions, and this is explicitly expressed by Eq. (3). Note that in this study we consider the station correlations as the mobility trends of bike usage, so larger POI differences result in higher mobility trends and hence stronger correlations. The adjacency matrices for the four bike sharing networks are visualized in Fig. 12.

2) Graph Convolutional Networks (`GCN`): Given the graph, G, constructed in the way described in the previous section, we adopt `GCN` [27] to capture the spatial correlations between stations. Specifically, the adjacency matrix $\mathbf{A}$ is first normalized into $\tilde{\mathbf{A}}$, *i.e.*,

$$\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{\frac{1}{2}}, \tag{4}$$

where $\mathbf{D}$ is degree matrix of $\mathbf{A}$. The convolutional operation at each level is then given by

$$\mathbf{H}_l = \tilde{\mathbf{A}}\mathbf{H}_{l-1}\mathbf{W}_{G,l-1}, \tag{5}$$

where $\mathbf{H}_{l-1} \in \mathbb{R}^{N \times s_{l-1}}$ and $\mathbf{H}_l \in \mathbb{R}^{N \times s_l}$ are projected features at the $(l-1)$-th and l-th layers, $l \in [1, L]$, and $\mathbf{W}_{G,l-1} \in \mathbb{R}^{s_{l-1} \times s_l}$ is a learnable parameter with s_l being the dimension of projected features at layer l. The input of the first layer is the original input signal at timestamp t, *i.e.*,

$$\mathbf{H}_1 = \mathbf{Z}_t \in \mathbb{R}^{N \times s}. \tag{6}$$

3) Long Short-Term Memory (`LSTM`): The `GCN` output, $\mathbf{H}_L \in \mathbb{R}^{N \times s_L}$, is flattened, and the flattened tensor $\bar{\mathbf{H}}_L$ is concatenated with external features, $\mathbf{ext} \in \mathbb{R}^w$, including weather and weekends/holidays, and fed into the `LSTM` module. Specifically, an `LSTM` cell at timestamp t [28] is formally given by

$$\begin{aligned}
\mathbf{m}_t &= \sigma\left(\mathbf{W}^m\left[\mathbf{h}_{t-1}, \bar{\mathbf{H}}_L, \mathbf{ext}\right] + \mathbf{b}^m\right), \\
\mathbf{n}_t &= \sigma\left(\mathbf{W}^n\left[\mathbf{h}_{t-1}, \bar{\mathbf{H}}_L, \mathbf{ext}\right] + \mathbf{b}^n\right), \\
\mathbf{o}_t &= \sigma\left(\mathbf{W}^o\left[\mathbf{h}_{t-1}, \bar{\mathbf{H}}_L, \mathbf{ext}\right] + \mathbf{b}^o\right), \\
\tilde{\mathbf{c}}_t &= \tanh\left(\mathbf{W}^m\left[\mathbf{h}_{t-1}, \bar{\mathbf{H}}_L, \mathbf{ext}\right] + \mathbf{b}^m\right), \\
\mathbf{c}_t &= \left(\mathbf{f}_t * \mathbf{c}_{t-1} + \mathbf{i}_t * \tilde{\mathbf{c}}_t\right), \\
\mathbf{h}_t &= \tanh\left(\mathbf{c}_t\right) * \mathbf{o}_t,
\end{aligned} \tag{7}$$

where $\mathbf{h}_t$, $\mathbf{c}_t \in \mathbb{R}^h$ are the `LSTM`'s hidden state and cell state respectively with h being the number of hidden layers, $\mathbf{W} \in \mathbb{R}^{h \times (h + N \cdot s_L + w)}$ and $\mathbf{b} \in \mathbb{R}^h$ are learnable parameters, and the rest are intermediate variables.

A fully connected layer takes the last hidden state of `LSTM` cell to generate predictions of bike pick-ups and drop-offs simultaneously. The previous hidden state of the `LSTM`, $\mathbf{h}_{t-1}$, is utilized to incorporate the dynamics of POI correlations at current timestamp t, which is represented by the adjacency attention as formulated in the following section.

4) Adjacency Attention Design: Following the manner of graph generation described previously, the spatial correlations of the bike sharing network is encoded in the graph G. However, defining graph edges as Euclidean distance between stations' surrounding POI numbers is an under-estimation, as the contributions of different POI categories to the network's spatial correlations could be different and vary with time. For example, during rush hours of a workday, stations around business areas are more active than stations near entertainment facilities. Therefore, we design an attention mechanism on the adjacency matrix $\mathbf{A}$ to capture and differentiate the varied contributions of each POI category.

Specifically, given the graph G, each entry of the adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ at the initial timestamp is a Euclidean distance of POI vectors for a station pair. $\mathbf{A}$ can be generated by calculating the Euclidean distance between POI vectors for every pair of stations. This is equivalent to first calculating the squared number difference for each POI category from which we get an intermediate matrix $\hat{\mathbf{A}} \in \mathbb{R}^{C \times N \times N}$, and then we calculate the square root of the summation of all the entries of $\hat{\mathbf{A}}$ along the first axis. Specifically, an entry of the intermediate matrix, $\hat{\mathbf{A}}$, for station i and k at the c-th POI type is given by

$$\hat{\mathbf{A}}_c[i,k] = \hat{\mathbf{A}}_c[k,i] = \left(\mathbf{p}_i[c] - \mathbf{p}_k[c]\right)^2. \tag{8}$$

The attention score for POI category c at time t is calculated by

$$\epsilon^c_t = \mathbf{v} \cdot \tanh\left(\left[\mathbf{V}\hat{\mathbf{A}}_c\mathbf{u}, \mathbf{U}\mathbf{h}_{t-1}\right]^\intercal\right), \tag{9}$$

where $\mathbf{v} \in \mathbb{R}^{v+u}$, $\mathbf{u} \in \mathbb{R}^N$, $\mathbf{V} \in \mathbb{R}^{v \times N}$ and $\mathbf{U} \in \mathbb{R}^{u \times h}$

are learnable model parameters; u, v are adjustable attention parameters; $\hat{\mathbf{A}}_c \in \mathbb{R}^{N\times N}$ is the corresponding intermediate matrix for POI category c; $[\bullet, \bullet]$ is the concatenation operation.

Then the attention weight of this category is given by a `softmax` function for the score of this category, *i.e.*,

$$\alpha_t^c = \frac{\exp\left(\epsilon_t^c\right)}{\sum_{c=1}^{C} \exp\left(\epsilon_t^c\right)}. \tag{10}$$

This way, the spatial correlation (adjacency matrix) attention captures the dynamic contribution of different POI categories at each timestamp. The weighted adjacency matrix $\tilde{\mathbf{A}}_t \in \mathbb{R}^{N\times N}$ at timestamp t is then computed by

$$\tilde{\mathbf{A}}_t = \sqrt{\alpha_t^1 \hat{\mathbf{A}}_1 + ... + \alpha_t^c \hat{\mathbf{A}}_c + ... + \alpha_t^C \hat{\mathbf{A}}_C}, \tag{11}$$

which is the updated adjacency matrix at that timestamp fed to the `GCN` following the same manner as Sec. IV-B2.

V. Experimental Studies

In this section, we first present the evaluation setup in Sec. V-A , followed by the experimental results in Sec. V-B.

A. Evaluation Setup

We compare `BikeGAAN` with the following baselines and state-of-art models for our experimental evaluation:

- `HA`: In Historical Average (`HA`) [29] we calculate the future usage based on all historical data at the same point of time.
- `ARIMA`: Auto Regressive Integrated Moving Average (`ARIMA`) is a statistical regression model for time series forecasting.
- `SES`: Simple Exponential Smoothing (`SES`) uses a weighted moving average which assigns exponentially decreasing weights for observations from newest to older.
- `MLP`: In Multi-layer Perceptron (`MLP`) we leverage the historical data to predict the usage of future timestamps.
- `RNN`: Recurrent Neural Network (`RNN`) [30] is a classic deep learning model for time series prediction.
- `LSTM`: Long Short-Term Memory (`LSTM`) neural network [28] is a variant of the recurrent neural network with inclusion of information gates.
- `GRU`: Gated Recurrent Units (`GRU`) [31] is another variant of the recurrent neural network with a different structure of information control.
- `CNN`: Convolutional Neural Network (`CNN`) leverages the bike usage of the past to give predictions of the future timestamp.
- `CNN-LSTM`/`CNN-GRU`/`CNN-RNN`: The `CNN` is combined with `LSTM`/`GRU`/`RNN` to predict future bike usage.
- `GCN`: The Graph Convolutional Network (`GCN`) developed by [10] predicts the time-series data where the adjacency matrix is treated as an adjustable parameter.

We evaluate the `BikeGAAN` and related schemes with the datasets presented in Sec. III-A. For all schemes, we leverage the bike usage data of the past 24 hours to predict the bike usage in the next following hour for the stations in the sharing networks we study. We first go through all the trip data for each city and identify the stations which have bike usage everyday and have not been relocated or removed. Other stations are not of our interest in this study. We find in 2019 totally 575 stations in NYC from January to June, 234 stations in Chicago and 270 stations in DC from April to June, and 43 stations in downtown LA in June. We use the user trip data during the above time periods for training, and evaluate the schemes on the data from July 1st to September 18th (80 days in total) for NYC, Chicago and DC and those from July 1st to July 16th (16 days) for LA.

TABLE III: Comparison between `BikeGAAN` and baselines for the four bike sharing systems.

Schemes	Citi	Divvy	Capital	Metro
`SES`	43.745	24.456	7.534	1.331
`MLP`	41.932	14.118	8.663	1.394
`ARIMA`	39.277	22.041	6.087	1.131
`HA`	19.630	11.106	3.653	0.759
`RNN`	16.973	7.580	3.070	0.797
`GRU`	16.148	7.299	3.258	0.785
`LSTM`	18.912	7.023	3.310	0.768
`CNN`	14.766	9.228	3.684	0.742
`CNN-RNN`	23.297	6.128	3.062	0.748
`CNN-LSTM`	30.898	6.329	3.245	0.783
`CNN-GRU`	23.049	6.315	3.267	0.754
`GCN`	12.271	6.550	3.065	0.736
`BikeGAAN`	**10.146**	**5.652**	**2.775**	**0.701**

We set the grid area $\mathcal{G}_{lat} \times \mathcal{G}_{lon}$ around each station to be 500m×500m, and count the POI distributions and nearby events. The bike usage of the network is combined with events to form the `GCN` inputs, such that for $\mathbf{Z}_t \in \mathbb{R}^{N\times s}$ and $s = 3$, with the first two channels being bike pick-ups and drop-offs and the last channel being stations' nearby events of all categories.

The number of projection layers of `GCN` in `BikeGAAN` is set to be two with the projected dimensions of each station being $s_1 = 1$ and $s_2 = 1$. We set the number of hidden layers of `LSTM`, h, as 256 for NYC, 128 for both Chicago and DC, and 16 for LA. The drop-out rate of `LSTM` cell is 0.5. `BikeGAAN` is trained with a learning rate of 0.001 and a batch size of 64 by Adam optimizer for 6,000 iterations for Citi Bike (NYC) and 5,000 iterations for others. All experimental evaluations are conducted upon a desktop with Intel i5-8700, 16GB RAM, Nvidia GeForce RTX2080Ti/GTX1060 and Windows 10.

We use the mean square error (MSE) as the comparison metric:

$$\textit{MSE} = \frac{1}{2N\tau} \sum_{q\in[p,d]} \sum_{i=0}^{N} \sum_{t\in\tau} \left(\hat{y}_{i,t}^{(q)} - z_{i1,t}^{(q)}\right)^2. \tag{12}$$

The models are trained and tested based on MSE. The training losses of `BikeGAAN` are shown in Fig. 13.

B. Evaluation Results

1) Model Performance: We first show the prediction accuracy of bike usage of the four bike sharing systems in Table III. From the table we can see that `BikeGAAN` has overall better performance than other baselines and state-of-art models, which demonstrates the effectiveness of the proposed representation of bike sharing networks. The `BikeGAAN` model captures the generation of bike usage as people's preference of travelling to a neighborhood with different points of their

interest from the original places. Conventional models like HA, ARIMA, RNN, GRU and LSTM can only capture temporal characteristics of bike usage, while GCN mainly learns spatial correlations, so those models cannot achieve high accuracy.

The CNN only encodes the bike usage sequence rather than geographic information, and hence CNN and its combinations with the RNN, GRU and LSTM fail to capture the spatial correlations between stations and cannot provide high prediction accuracy. BikeGAAN, however, outperforms them in terms of MSE due to its comprehensive structure in capturing the bike flow dynamics.

Table III shows that there is a great difference in model performance between different cities. This is because of the large difference in the total number of bike usage between the four cities as described in III-A, where NYC has the most bike trips, followed by Chicago, DC and LA. For a city with more bike usage, the mobility pattern of pick-ups/drop-offs is more complex, making the predictions of the models less accurate.

The importance of different POIs may vary across different cities due to different functional zones of the cities. We further illustrate the weights of adjacency attention in Fig. 14 with respect to the seven POI categories. For each city, we normalize the attention weights, and present the stacked bar, where larger proportion represents more importance in the model learning. From experimental study, we notice that the softmax function in Eq. (10) eliminates the time dependency of the attention scores. Therefore, the attention weights for each city keep the same with varied time periods.

From Fig. 14 we can see that in NYC and Chicago healthcare POIs have the largest contribution to the station correlations, probably because in these two cities the healthcare POIs are mainly located near residential areas, leading to a large POI difference between business center/workplaces and residential areas. We consider that commute is one of the major reasons for traveling of urban residents. For DC the transportation POIs and POIs such as marketplaces, childcare and post offices (denoted as "other"), have the most contributions to the station correlations. This is because of the special functionality of DC where the government facilities/workplaces are located in the center of the city where there are limited numbers of transportation and "other" facilities such as parking lots, gas stations and marketplaces. For LA the entertainment POIs play an important part in the station correlations along with education and transportation POIs. A large number of attractions around LA may lead to different mobility trends from the other three cities.

2) Visualization: Fig. 15 shows bike usage as well as BikeGAAN's predictions of two stations from NYC and DC during July 1st to July 7th. We note that the Independence Day (the 4th of July) is on Thursday, and the bike usage on the following Friday and even on Wednesday follows the same pattern as the holiday, indicating that people are taking those two workdays off. BikeGAAN takes into account this factor (as in Sec. III-B4) and hence achieves accurate predictions. The highly accurate predictions can benefit the bike sharing system operation such as bike re-balancing.

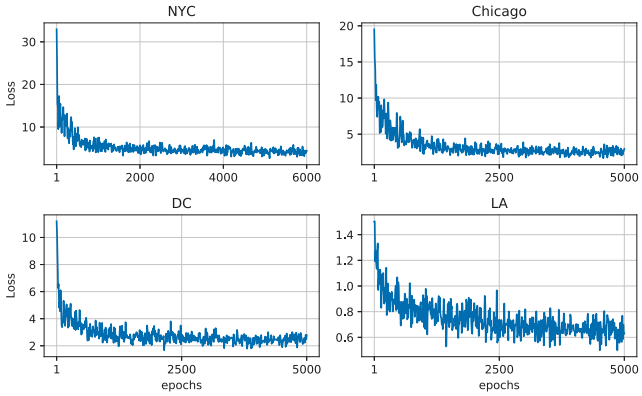


Fig. 13: Training losses (MSE) of BikeGAAN for the four systems.

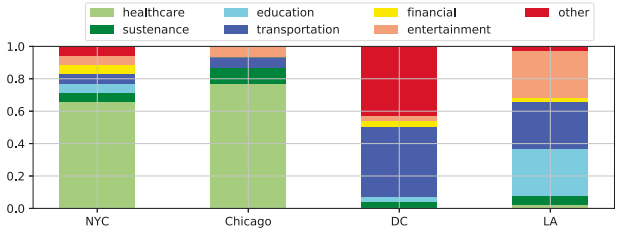
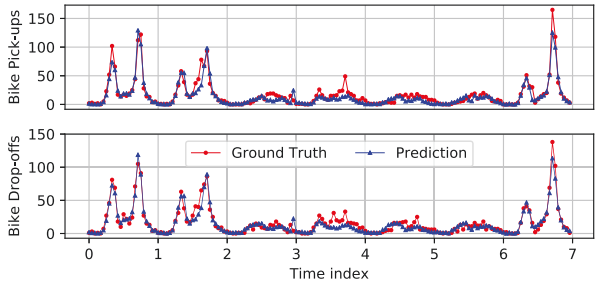
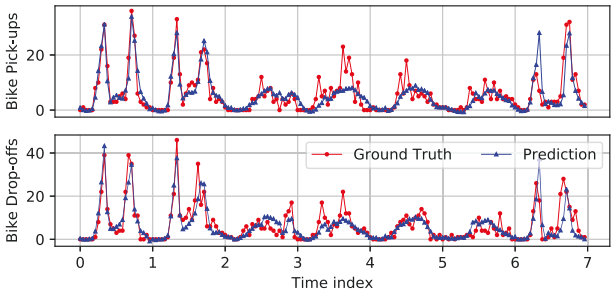


Fig. 14: Adjacency attention weights for the four systems.



(a) A station at NYC (40.751873°N, 73.977706°W).



(b) A station at DC (38.89696°N, 77.00493°W).

Fig. 15: Hourly bike usage prediction from July 1st to July 7th for a station at (a) Citi Bike and (b) Capital Bikeshare, respectively.

VI. System Deployment Discussion

We briefly discuss some system deployment discussions related to BikeGAAN.

1) Bike sharing network redeployment: In this work, only stations that have not been removed or relocated are studied. However, bike sharing networks usually change with resident's evolving demand. Bike stations might be reconfigured [32] and the systems might be expanded [26], [33]–[35]. Further exploration on how to encode such dynamics of the systems into the network structure will be considered in our future

work.

2) Incorporating transportation networks: The bike sharing system along with other means of transportation such as bus, subway, ride sharing, etc. form a transportation network of a city [36]. There exist traffic correlations between these transportation systems. Therefore, incorporating such correlations may help improve not only the accuracy of bike usage station, but also the efficiency of urban transportation for city management, which will be considered in our future work.

VII. Conclusion

In this work, we develop `BikeGAAN` with a novel representation of bike sharing networks and adjacency attention mechanism to predict the bike usage (pick-ups/drop-offs). We have conducted comprehensive and detailed analysis on real-world datasets of four major cities in the U.S., NYC, Chicago, DC and LA, and demonstrated how the weather conditions, station locations and surrounding POIs, weekends/holidays and events impact the bike usage. We leverage the nearby POI distributions between stations as their spatial correlations. The adjacency attention mechanism captures and differentiates the contributions of each POI category to the spatial correlations between stations. We have conducted extensive experimental study of our model on real-world datasets of the four cities. The results show that `BikeGAAN` outperforms other baselines and state-of-the-art models in predictions of bike usage for the entire systems.

Acknowledgment

This project is supported in part by the University of Connecticut Research Excellence Program (FY20-21 REP Award).

References

[1] M. H. Chen, A. Jauhri, and J. P. Shen, "Data driven analysis of the potentials of dynamic ride pooling," in *Proc. ACM SIGSPATIAL*, 2017, pp. 7–12.

[2] Z. Lin, J. Feng, Z. Lu, Y. Li, and D. Jin, "DeepSTN+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis," in *Proc. AAAI*, vol. 33, 2019, pp. 1020–1027.

[3] Y. Li, Z. Zhu, D. Kong, M. Xu, and Y. Zhao, "Learning heterogeneous spatial-temporal representation for bike-sharing demand prediction," in *Proc. AAAI*, vol. 33, 2019, pp. 1004–1011.

[4] R. Giot and R. Cherrier, "Predicting bikeshare system usage up to one day ahead," in *Proc. IEEE CIVTS*, 2014, pp. 22–29.

[5] Y. Li, Y. Zheng, H. Zhang, and L. Chen, "Traffic prediction in a bike-sharing system," in *Proc. ACM SIGSPATIAL*, 2015, p. 33.

[6] X. Zhou, "Understanding spatiotemporal patterns of biking behavior by analyzing massive bike sharing data in chicago," *PloS One*, vol. 10, no. 10, p. e0137922, 2015.

[7] J. Bao, C. Xu, P. Liu, and W. Wang, "Exploring bikesharing travel patterns and trip purposes using smart card data and online point of interests," *Networks and Spatial Economics*, vol. 17, no. 4, pp. 1231–1253, 2017.

[8] P. Hulot, D. Aloise, and S. D. Jena, "Towards station-level demand prediction for effective rebalancing in bike-sharing systems," in *Proc. ACM SIGKDD*, 2018, pp. 378–386.

[9] P.-C. Chen, H.-Y. Hsieh, X. K. Sigalingging, Y.-R. Chen, and J.-S. Leu, "Prediction of station level demand in a bike sharing system using recurrent neural networks," in *Proc. IEEE VTC Spring*, 2017, pp. 1–5.

[10] D. Chai, L. Wang, and Q. Yang, "Bike flow prediction with multi-graph convolutional networks," in *Proc. ACM SIGSPATIAL*, 2018, pp. 397–400.

[11] 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," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 258–276, 2018.

[12] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[13] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[14] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. NIPS*, 2015, pp. 577–585.

[15] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Proc. IEEE ICASSP*, 2016, pp. 4945–4949.

[16] A. Zeyer, K. Irie, R. Schlüter, and H. Ney, "Improved training of end-to-end attention models for speech recognition," *arXiv preprint arXiv:1805.03294*, 2018.

[17] Z. Ji, K. Xiong, Y. Pang, and X. Li, "Video summarization with attention-based encoder-decoder networks," *IEEE TCSVT*, 2019.

[18] J. Song, Z. Guo, L. Gao, W. Liu, D. Zhang, and H. T. Shen, "Hierarchical LSTM with adjusted temporal attention for video captioning," *arXiv preprint arXiv:1706.01231*, 2017.

[19] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proc. AAAI*, 2017.

[20] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "GeoMAN: Multi-level attention networks for geo-sensory time series prediction." in *Proc. IJCAI*, 2018, pp. 3428–3434.

[21] S. He and K. G. Shin, "Towards fine-grained flow forecasting: A graph attention approach for bike sharing systems," in *Proc. WWW*, 2020, pp. 88–98.

[22] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *Proc. WWW*, 2019, pp. 2022–2032.

[23] J. Zhang, X. Shi, J. Xie, H. Ma, I. King, and D.-Y. Yeung, "GaAn: Gated attention networks for learning on large and spatiotemporal graphs," *arXiv preprint arXiv:1803.07294*, 2018.

[24] C. Zheng, X. Fan, C. Wang, and J. Qi, "GMAN: A graph multi-attention network for traffic prediction," in *Proc. AAAI*, vol. 34, no. 01, 2020, pp. 1234–1241.

[25] L. Chen, D. Zhang, L. Wang, D. Yang, X. Ma, S. Li, Z. Wu, G. Pan, T.-M.-T. Nguyen, and J. Jakubowicz, "Dynamic cluster-based over-demand prediction in bike sharing systems," in *Proc. UbiComp*, 2016, pp. 841–852.

[26] X. Yang and S. He, "Towards dynamic urban bike usage prediction for station network reconfiguration," in *Proc. ACM UrbComp*, 2020.

[27] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[28] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, "Learning long-term dependencies in narx recurrent neural networks," *IEEE TNNLS*, vol. 7, no. 6, pp. 1329–1338, 1996.

[29] J. E. Froehlich, J. Neumann, and N. Oliver, "Sensing and predicting the pulse of the city through shared bicycling," in *Proc. IJCAI*, 2009.

[30] Y. Pan, R. C. Zheng, J. Zhang, and X. Yao, "Predicting bike sharing demand using recurrent neural networks," *Procedia Computer Science*, vol. 147, pp. 562–566, 2019.

[31] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[32] S. He and K. G. Shin, "(Re) Configuring Bike Station Network via Crowdsourced Information Fusion and Joint Optimization," in *Proc. ACM MobiHoc*, 2018, pp. 1–10.

[33] J. Zhang, X. Pan, M. Li, and P. S. Yu, "Bicycle-sharing systems expansion: station re-deployment through crowd planning," in *Proc. ACM SIGSPATIAL*, 2016, pp. 1–10.

[34] S. He and K. G. Shin, "Dynamic flow distribution prediction for urban dockless e-scooter sharing reconfiguration," in *Proc. WWW*, 2020, pp. 133–143.

[35] J. Liu, L. Sun, Q. Li, J. Ming, Y. Liu, and H. Xiong, "Functional zone based hierarchical demand prediction for bike system expansion," in *Proc. ACM SIGKDD*, 2017, pp. 957–966.

[36] S. He and K. G. Shin, "Spatio-temporal capsule-based reinforcement learning for mobility-on-demand network coordination," in *Proc. WWW*, 2019, p. 2806–2813.