# D$^3$P: Data-driven Demand Prediction for Fast Expanding Electric Vehicle Sharing Systems

MAN LUO, University of Warwick and The Alan Turing Institute, UK
BOWEN DU, University of Warwick, UK
KONSTANTIN KLEMMER, University of Warwick and The Alan Turing Institute, UK
HONGMING ZHU, Tongji University, China
HAKAN FERHATOSMANOGLU and HONGKAI WEN, University of Warwick, UK

The future of urban mobility is expected to be shared and electric. It is not only a more sustainable paradigm that can reduce emissions, but can also bring societal benefits by offering a more affordable on-demand mobility option to the general public. Many car sharing service providers as well as automobile manufacturers are entering the competition by expanding both their EV fleets and renting/returning station networks, aiming to seize a share of the market and to bring car sharing to the zero emissions level. During their fast expansion, one determinant for success is the ability of predicting the demand of stations as the entire system is growing continuously. There are several challenges in this demand prediction problem: First, unlike most of the existing work which predicts demand only for static systems or at few stages of expansion, in the real world we often need to predict the demand *as* or even *before* stations are being deployed or closed, to provide information and decision support. Second, for the new stations to be deployed, there is no historical data available to help the prediction of their demand. Finally, the impact of deploying/closing stations on the other stations in the system can be complex. To address these challenges, we formulate the demand prediction problem in the context of fast expanding electric vehicle sharing systems, and propose a *data-driven demand prediction* approach which aims to model the expansion dynamics directly from the data. We use a local temporal encoding process to handle the historical data for each existing station, and a dynamic spatial encoding process to take correlations between stations into account with Graph Convolutional Neural Networks (GCN). The encoded features are fed to a multi-scale predictor, which forecasts both the long-term expected demand of the stations and their instant demand in the near future. We evaluate the proposed approach with real-world data collected from a major EV sharing platform for one year. Experimental results demonstrate that our approach significantly outperforms the state of the art, showing up to three-fold performance gain in predicting demand for the expanding EV sharing systems.

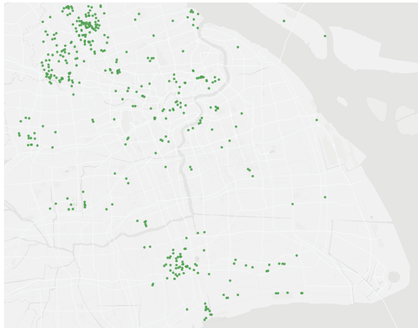
CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing systems and tools**; • **Information systems → Spatial-temporal systems**.

Additional Key Words and Phrases: Electric Vehicle Sharing; Data-driven; Demand Prediction; System Expansion

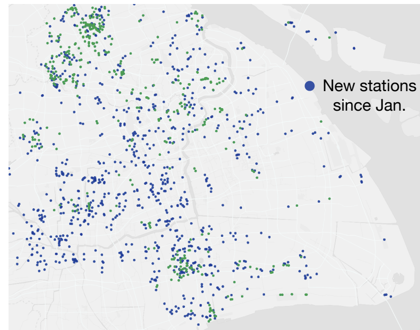
**ACM Reference Format:**
Man Luo, Bowen Du, Konstantin Klemmer, Hongming Zhu, Hakan Ferhatosmanoglu, and Hongkai Wen. 2020. D$^3$P: Data-driven Demand Prediction for Fast Expanding Electric Vehicle Sharing Systems. 1, 1 (April 2020), 21 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Authors' addresses: Man Luo, University of Warwick, The Alan Turing Institute, UK; Bowen Du, University of Warwick, UK; Konstantin Klemmer, University of Warwick, The Alan Turing Institute, UK; Hongming Zhu, Tongji University, China; Hakan Ferhatosmanoglu; Hongkai Wen, University of Warwick, UK.

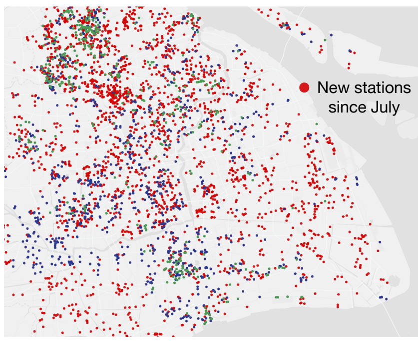

XXXX-XXXX/2020/4-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn



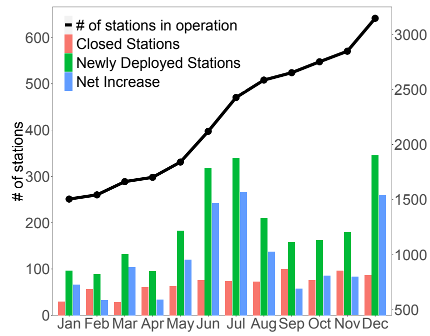
(a) Station distribution in January.




(b) Station distribution in July.




(c) Station distribution in December.




(d) Statistics of stations over 12 months.

Fig. 1. The expansion process of an EV sharing system in Shanghai during the year 2017. Images better viewed in colour.

## 1 INTRODUCTION

Cities around the globe struggle with congestion and poor air quality. Vehicle sharing systems have long been recognized as an environmentally friendly mobility option, reducing vehicles on the road while cutting out unnecessary $CO_2$ emissions. With the recent advances in battery technologies, a new generation of car sharing services is going one step further, by offering full electric vehicle (EV) fleets with fast expanding infrastructures in major cities, e.g. Bluecity [2] in London, WeShare [6] in Berlin, and BlueSG [3] in Singapore. Traditional car sharing providers have also started to populate their EV fleets, e.g., ZipCar [7] seeks to provide over 9,000 full electric vehicles across London by 2025. According to a recent study [30], the global market of EV sharing services is poised for even faster growth in the near future, due to the incentives and regulations put in place by governments across the world to encourage broader EV usages. Electric vehicle sharing services will reshape the current urban transportation paradigm and potentially offer significant societal benefits, as they provide a more efficient, sustainable, and affordable mobility option to all citizens, eliminating the necessity of individual car ownership.

Despite their increased popularity and rapid growth across the globe, there are still many challenges and open questions with respect to the practicality of current EV sharing systems. A major problem of those systems is that during their expansion processes, dynamic predictions of user demand and implementing expansion strategies become substantially more difficult. This is not only complicating decision making as to where and when to deploy new stations or close the poorly performing ones, but also of great importance to the effective operation of

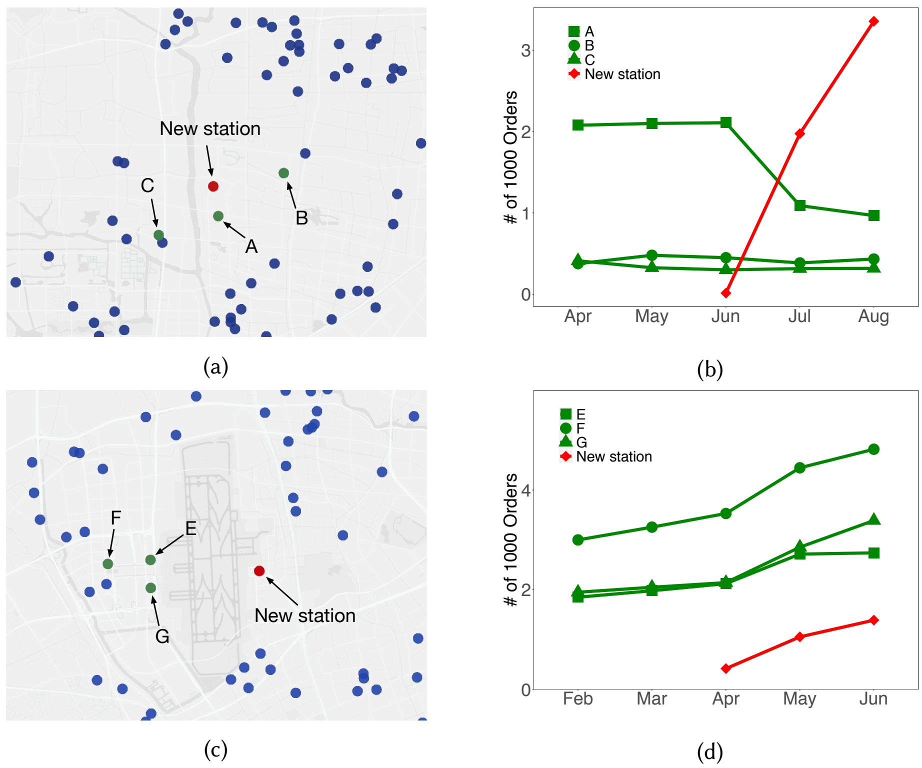



Fig. 2. Different types of impact when deploying new stations to the current station network. (a)-(b) An example showing that a new station 'steals' the user demand from one of its neighbour stations. (c)-(d) An example showing that a new station 'boosts' the demand of its neighbour stations.

currently used stations. Understanding the potential impact of proposed expansions to their demand can provide valuable insights on a number of vital tasks such as scheduling, pricing, and rebalancing.

However, in the context of such continuous expanding EV sharing systems, this demand prediction problem is not trivial. Most of the existing work on demand prediction [11, 17, 22, 23, 27, 39] assumes the stations in the system are static and different forms of historical data is available for all stations, or only predicts demand after fixed (one or two) expansion stages where stations are only deployed in batches [24]. These assumptions often collapse in the real world. For instance, Fig. 1(a)-(c) visualize the expansion process of a major EV sharing platform in Shanghai during 2017. We see that in the beginning stations are only scattered within limited areas, while at the end of the year the entire city has been densely covered. As shown in Fig. 1(d), within 12 months the total number of stations in operation has doubled (from roughly 1500 to more than 3000). In addition, we see that in each month there are continuously hundreds of stations being deployed or closed. In this case, predicting demand at those newly deployed or to be deployed stations is challenging, since there is no sufficient historical data available as prior knowledge.

On the other hand, the new dynamics caused by the expansion process may have complex effects on the entire EV sharing system. For example, as shown in Fig. 2, deploying stations at various places may have completely different effects. For example, the new station (denoted as the red dot) in Fig. 2(a) 'steals' the demand from one of its neighbors (station A) since its deployment in June (see the changes of their order numbers in Fig. 2(b)).

We found from the data that the new station was deployed in a major shopping center, and therefore it likely attracted the users who originally preferred to rent/return cars from/to station A, which is just one block away. In contrast as shown in Fig. 2(c) and (d), deploying a new station has increased the orders of its neighbor stations E, F and G collectively. In particular, we found from the data that a large portion of their increased orders have the new station as the destination. This means that after the new station is deployed, many users tend to rent EVs from E, F, and G because together with the new station, they offer convenient short-range connections for them to get to the other side of the airport. In the presence of such dynamics, accurate demand prediction for the remaining stations becomes very challenging, due to the non-trivial system dynamics caused by the continuous expansion process.

To address those challenges, we propose a novel data-driven demand prediction approach, which models the expansion of EV sharing systems with time-varying graphs, and is able to predict the demand of stations throughout the expansion process. Specifically, for each station that comes in operation, we employ a local temporal encoding module to capture the correlations within the historical data. The extracted features from all stations are then compiled by a dynamic spatial encoding module, which considers the spatial dependencies between them as multiple graphs, and fuses the station-level features with Graph Convolutional Neural Networks (GCN). Based on the encoded information and future expansion plan (i.e., which stations to be deployed or closed), we consider a multi-scale predictor which forecasts station demand at different scales: From instant demand in the immediate near future to the long term expected demand, for both stations to be deployed and the ones remaining. The technical contributions of this paper are as follows:

- To the best of our knowledge, this is the first work that investigates the demand prediction problem in the context of continuous expanding electric vehicle sharing systems. We conduct a comprehensive study with the operational data from a fast growing EV sharing system in the real world, and identify the needs and benefits of forecasting the accurate user demand as the system continuously expands, which have not been studied before.
- We propose a novel data-driven approach for demand prediction which is capable of modelling the complex dynamics caused by the continuous system expansion. The key idea is to model the evolving station network of the EV sharing system as multiple time-varying graphs, which describe the different types of correlations between the stations. With those graphs, we propose new encoding approaches which perform the local temporal encoding and global spatial encoding processes in tandem, to jointly incorporate the historical knowledge at individual stations and the spatial dependencies between them.
- We design a new multi-scale predictor on top of the encoding processes, which is able to forecast the user demand of both stations to be deployed and those already existing in the current system. In addition, our predictor can predict not only the expected future demand of the stations, but also their instant demand in subsequent timestamps, which allows us to better understand both short and long term impact of the system expansion.
- We evaluate the proposed demand prediction approach on both real and simulated data collected from a major EV sharing platform in Shanghai for one year, including data from over 3000 stations and 16,000 electric vehicles in operation. Extensive experiments have shown that our approach significantly outperforms the state of the art, offering up to three-fold improvement in prediction accuracy and is robust to different levels of expansion dynamics.

The rest of this paper is organized as follows. We first formulate the problem of demand prediction for continuously expanding EV sharing systems in Section 2. Then we present the proposed data-driven demand prediction approach, where Sections 3 and 4 discuss the local temporal and dynamic spatial encoding techniques respectively, and Section 5 describes the design of the multi-scale predictor. Section 6 evaluates the performance of our approach with real-world EV sharing system data. We review the related work in Section 7 and conclude the paper in Section 8.

## 2 PROBLEM FORMULATION

In this section, we first introduce some key concepts used throughout the paper, then we formulate the problem of demand prediction for expanding EV sharing systems and provide an overview of the proposed framework.

### 2.1 Preliminaries

*2.1.1 Electric Vehicle (EV) Stations.* Let $s_i$ be a station in the Electric Vehicle (EV) sharing system. In this paper, we assume $s_i$ can be represented as a tuple $(\boldsymbol{x}_i, m_i)$, where $\boldsymbol{x}_i$ are the geographic coordinates (e.g. latitude and longitude) of station $s_i$, and $m_i$ is the number of charging docks within $s_i$. We also assume that for a given $s_i$, we can extract a number of geospatial features based on its location $\boldsymbol{x}_i$, such as nearby Points of Interest (POI) or the distribution of road networks within a certain radius.

*2.1.2 Instant Station Demand.* We define the instant demand of a station $s_i$ at timestamp $t$ as the rent/return frequency of $s_i$, denoted as $d_i(t)$. In this paper the granularity of timestamp $t$ is days, i.e., we focus on daily station demand, but the proposed approach can be extended to adopt other time granularity levels.

*2.1.3 Expected Station Demand.* For a station $s_i$, the expected demand $\bar{d}_i$ over a period $[t_s, t_e]$ can be defined as the mean $\bar{d}_i(t_s, t_e) = |t_e - t_s|^{-1} \sum_{t=t_s}^{t_e} d_i(t)$. We consider the expected demand from the current time $t$ towards the future, and aggregate it according to some index, e.g., days of the week. Without loss of generality, in the following text we denote the future expected demand of station $s_i$ as a vector $\bar{\boldsymbol{d}}_i = [\bar{d}_i^{\mathrm{Mo}}, \bar{d}_i^{\mathrm{Tu}}, ..., \bar{d}_i^{\mathrm{Su}}]$, $\bar{\boldsymbol{d}}_i \in \mathbb{R}^7$ for different days of the week.

*2.1.4 Station Network.* We model the stations of the EV sharing network as a graph $G = (S, A)$, where the nodes $s_i \in S$ are stations as defined above. An edge $a_{ij} \in A$ may encode a certain type of correlation between two stations $s_i$ and $s_j$, e.g., the spatial distance between them, or similarity between their POI/road network features. Section 4.1 will discuss how we construct multiple graphs to capture such inter-station relationships in more details.

*2.1.5 Station Network Dynamics.* Unlike existing work, in this paper we assume the station network is *evolving* over time, i.e., $G = (S, A)$ is a time-varying graph. More specifically, let $G_{t-1} = (S_{t-1}, A_{t-1})$ represents the station network at time $t-1$. Without loss of generality, we assume that at time $t-1$, there is an expansion plan to be implemented at time $t$, which shall expand the current station network from $G_{t-1}$ to the *planned network* $G_t^{\mathrm{P}}$. Let's assume during this a set of new stations $S^+$ will be deployed, while existing stations $S^-$ will be closed. If the expansion plan goes through, then at time $t$ the actual station network $G_t = (S_t, A_t)$ becomes the planned $G_t^{\mathrm{P}}$, where

$$S_t = (S_{t-1} - S^-) \cup S^+ \tag{1a}$$

$$A_t = (A_{t-1} - \{a_{ij} | s_i \in S^- \text{ or } s_j \in S^-\}) \cup \{a_{ij} | s_i \in S^+ \text{ or } s_j \in S^+\} \tag{1b}$$

### 2.2 The Demand Prediction Problem

Suppose that at time $t$, we have the topology $G_1, ...G_t$ and demand $D_1, ..., D_t$ of the station network, where $D_t = \{d_i(t) | s_i \in G_t\}$. Let $G_{t+1}^{\mathrm{P}}$ be the planned station network at the future timestamp $t+1$. The demand prediction problem addressed in this paper is that given the historical data, for an arbitrary station in the planned network $s_i \in G_{t+1}^{\mathrm{P}}$ (deployed or not yet deployed) we aim to estimate both its expected future demand $\hat{\bar{\boldsymbol{d}}}_i$ and the subsequent $k$ instant demand $[\hat{d}_i(t+1), \hat{d}_i(t+2), ..., \hat{d}_i(t+k)]$, which minimise the mean square errors with

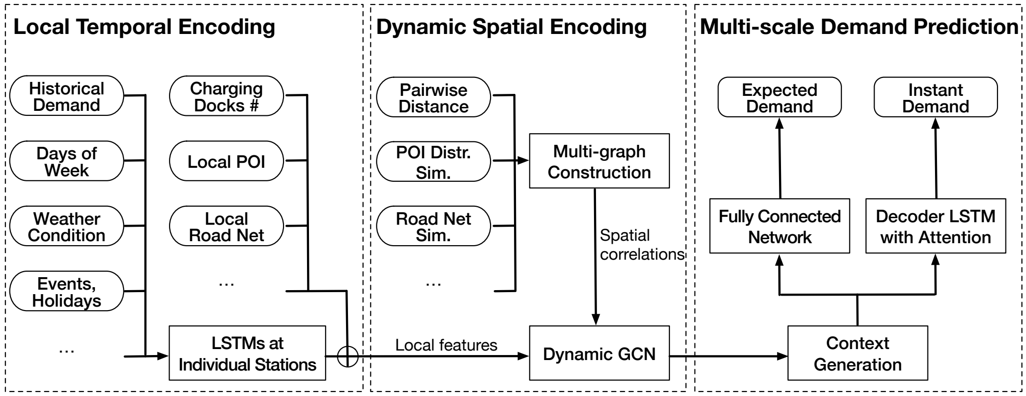



Fig. 3. Overview of the proposed data-driven demand prediction framework.

respect to the ground truth $\bar{\boldsymbol{d}}_i$ and $d_i$:

$$\delta_{\bar{\boldsymbol{d}}_i} = |\bar{\boldsymbol{d}}_i|^{-1} \|\hat{\bar{\boldsymbol{d}}}_i - \bar{\boldsymbol{d}}_i\|^2 \tag{2a}$$

$$\delta_{d_i} = k^{-1} \sum_{\tau=t+1}^{t+k} \|\hat{d}_i(\tau) - d_i(\tau)\|^2 \tag{2b}$$

In practice, the expected demand $\hat{\bar{\boldsymbol{d}}}_i$ can be viewed as a metric for the *long-term performance* of stations $s_i$, e.g., if $s_i$ is a station to be deployed, $\hat{\bar{\boldsymbol{d}}}_i$ quantifies the average level of demand it may be able to attract. On the other hand, the sequence of instant demand $[\hat{d}_i(t+1), \hat{d}_i(t+2), ..., \hat{d}_i(t+k)]$ describes the *immediate trend* of station demand under the impact of the expansion plan, which can help to optimise key future operation strategies such as marketing and resource allocation.

## 2.3 Framework Overview

Fig. 3 shows the overview of the proposed data-driven demand prediction framework, which consists of three major components:

*2.3.1 Local Temporal Encoding.* During the life cycle of a station $s_i$ (from being deployed to shut down), its demand can be viewed as a time series, where the current demand $d_i(t)$ should correlate with the local historical demand $d_i(t-1), ..., d_i(1)$. In addition, there may exist other temporal factors that can influence the demand of individual stations, such as weather conditions, air pollution levels, days of the week and public holidays etc. To model such temporal dependencies, we assign a Long Short-Term Memory (LSTM) network at each individual station when being deployed, and use them to encode local temporal information at station level.

*2.3.2 Dynamic Spatial Encoding.* Intuitively, the demand of a station $s_i$ can be affected also by the other stations in the network. To capture the spatial correlations, at each time $t$ we construct multiple graphs to encode different spatial relationships between the stations, e.g., inter-station distances, POI similarity, and road network metrics. Then we use Graph Convolutional Neural Networks (GCN) to fuse those graphs and encode the previously computed local features of individual stations. In particular, as the station network is evolving over time, we develop a dynamic GCN (DGCN) which is able to process such time-varying graphs.

*2.3.3 Multi-scale Demand Prediction.* Based on the results of the above temporal and spatial encoding, we aim to predict both the expected demand and subsequent instant demand of stations after the planned expansion. To achieve that, we design a multi-scale prediction network, which firstly compiles the previously learned features into a context vector. For expected demand, it uses a fully connected branch to perform the prediction, while on the other hand, it considers a decoder LSTM network with attention mechanism to forecast instant demand at multiple future timestamps.

We are now in a position to elaborate the proposed data-driven demand prediction approach in more detail.

## 3 LOCAL TEMPORAL ENCODING

Like in many other shared mobility services, we observe that the demand of stations in the EV sharing network exhibits strong temporal correlations, as shown later in Fig. 6(b). For instance, although it fluctuates largely over time, the demand at an individual station approximates certain periodical patterns at different days across the week. In that sense, exploiting such knowledge can help significantly in estimating the future demand of the existing stations, which will have a positive knock-on effect when predicting demand for the new stations during expansion. However, those demand patterns are typically influenced by multiple factors such as weather, air quality and events, and individual stations may react to those factors differently. Therefore, it is often not optimal to only incorporate the temporal information globally for the station network, but instead we model such microdynamics at the station level.

Concretely, when a station $s_i$ is deployed, we instantiate an LSTM network which keeps processing its demand records and the additional temporal information available, e.g. weather, days of the week and public holiday/events. In our implementation, we encode such temporal information as feature vectors, e.g. the weather data can be discretized and represented as one hot vectors, and all the vectors are concatenated as the input to the LSTMs. To avoid over-fitting, we train the LSTMs with shared weights across stations. At time $t$, the LSTM encodes the station's historical demand $d_i(t), d_i(t-1), \ldots$ as well as the auxiliary information into a temporal feature vector $\mathbf{f}_i(t)$. Moreover, in this paper we also condition $\mathbf{f}_i(t)$ with a static station feature vector $\mathbf{c}_i$, which describes key attributes of the station $s_i$ such as its number of available charging docks $m_i$, nearby POIs and environmental characteristics, etc. We encode the static feature $\mathbf{c}_i$ in a similar way with the temporal information, e.g. the POI data can be represented as vectors where each element indicates the number of a particular type of POIs that are close to the station. Therefore, $\mathbf{f}_i(t)$ and $\mathbf{c}_i$ carry important local information about individual stations since they started operating, which are concatenated and passed on as the input for the later spatial encoding. Fig. 4 shows the workflow of the proposed approach, where at each timestamp we maintain a collection of local LSTMs to encode information of individual stations.

## 4 DYNAMIC SPATIAL ENCODING

### 4.1 Constructing Multiple Graphs

As discussed in Section 2.1, at a given time $t$ we represent the station network as a graph $G_t = (S_t, A_t)$, where $S_t$ are the set of current stations and $A_t$ is the adjacent matrix describing the pairwise correlations between them. In practice there are often more than one types of correlations, which cannot be effectively captured by a single graph. Therefore in this paper we construct multiple graphs to encode the complex inter-station relationships [16] particularly the *distance graph*, the *functional similarity graph*, and the *road accessibility graph* (see Fig. 4).

*4.1.1 Distance:* In most cases, we observe that the demand of stations close to each other are highly correlated, e.g., they may be deployed around the same shopping centre, and thus tend to be used interchangeably. We capture such correlations with a distance graph $A^{\mathrm{D}}$, whose elements are the reciprocal of station distance:

$$a_{ij}^{\mathrm{D}} = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^{-1} \tag{3}$$

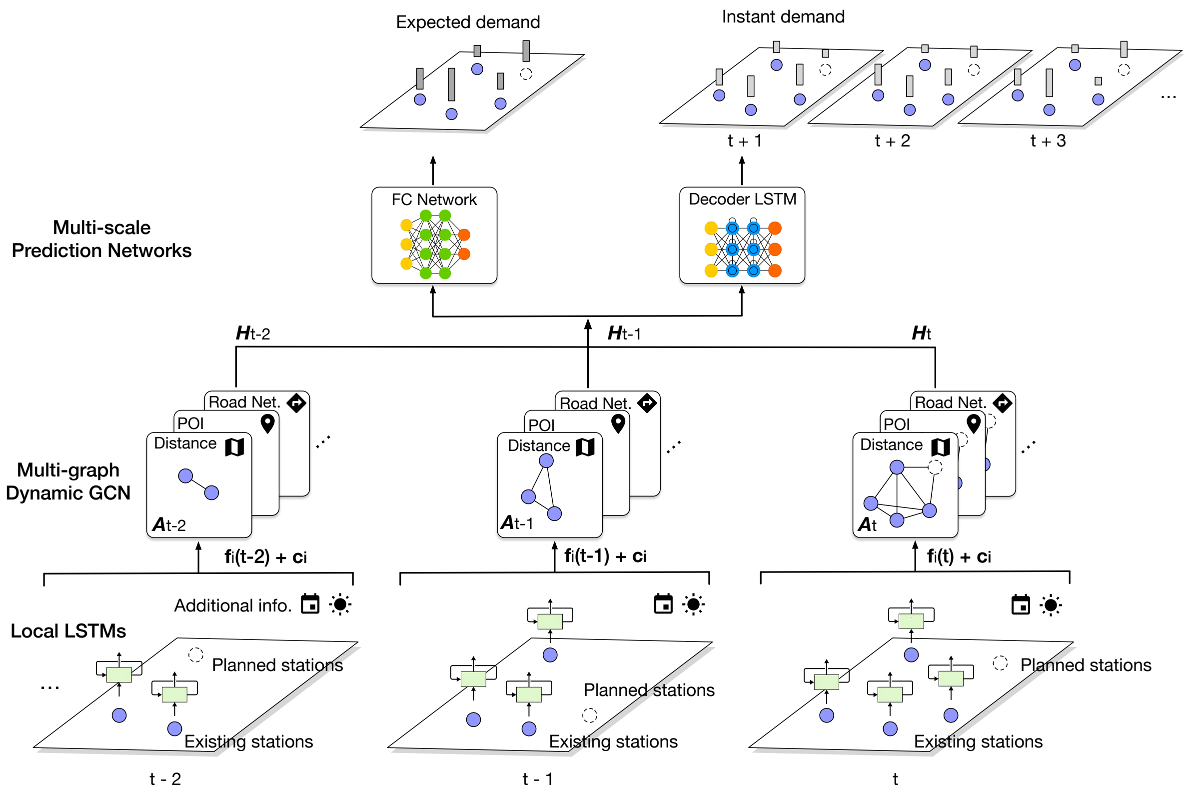



Fig. 4. The workflow of the proposed data-driven demand prediction approach. The local temporal encoding process produces the temporal and static features $\mathbf{f}_i(t)$ and $\mathbf{c}_i$, which are concatenated as the input to the spatial encoding process with the multi-graph dynamic GCN. The output of the GCN $\boldsymbol{H}_t$ encodes both the spatial and temporal features across different stations, and are fed into the multi-scale prediction networks.

where $\boldsymbol{x}_i$,$\boldsymbol{x}_j$ are the station coordinates, and $\|\cdot\|_2$ is the Euclidean distance. We also set diag $(A^{\mathrm{D}})$ to 1 to include self loops in the graph.

*4.1.2 Functional Similarity.* Intuitively, stations deployed in areas with similar functionalities should share comparable demand patterns. For instance, stations close to university campuses typically have significantly higher demand during weekends. We characterize the functionalities of stations by considering the distributions of their surrounding POIs. Suppose we have $P$ different categories of POIs in total, and let $\boldsymbol{p}_i$ be the distribution of the $P$ types of POIs within a certain radius of station $s_i$. The functional similarity graph $A^{\mathrm{F}}$ is then defined as:

$$a_{ij}^{\mathrm{F}} = \mathrm{sim}\left(\boldsymbol{p}_i, \boldsymbol{p}_j\right) \tag{4}$$

where $\mathrm{sim}(,) \in [0,1]$ is a similarity measure which quantifies the distance between feature vectors. In our experiments, we use the soft cosine function.

*4.1.3 Road Accessibility.* Another factor that affects station demand is the accessibility to road networks. Intuitively, stations close to major ring roads, or within areas that have densely connected streets would have higher demand. To model this, we consider the drivable streets in the vicinity of a station $s_i$ as a local road network, containing different types of road segments and their junctions. We extract a feature vector $\boldsymbol{r}_i$ from the local road network, which encodes information such as the road segments density, average junction degree and mean centrality. Given those features, the road accessibility graph can be defined with a similarity function sim (, ):

$$a_{ij}^{\mathrm{R}} = \mathrm{sim}\,(\boldsymbol{r}_i, \boldsymbol{r}_j) \tag{5}$$

where we also use soft cosine as the similarity function.

## 4.2 Dynamic Multi-graph Convolution

At time $t-1$, given the constructed graphs $\boldsymbol{A}_{t-1} = \{A_{t-1}^D, A_{t-1}^F, A_{t-1}^R\}$ which describe the inter-station relationships, we propose a dynamic multi-graph GCN (DGCN) to fuse such spatial knowledge with local features $\mathbf{f}_i(t-1)$ and $\mathbf{c}_i$ computed by the station-level temporal encoding. In the proposed DGCN, we perform multi-graph convolution as follows:

$$\boldsymbol{H}_{t-1}^{(l)} = \sigma\Bigg(\sum_{A_{t-1} \in \boldsymbol{A}_{t-1}} f(A_{t-1}) \boldsymbol{H}_{t-1}^{(l-1)} \boldsymbol{W}_{t-1}^{(l-1)}\Bigg) \tag{6}$$

where $\boldsymbol{H}_{t-1}^{l-1}$ and $\boldsymbol{H}_{t-1}^{l}$ are the hidden features of layers $l-1$ and $l$ respectively, while $\boldsymbol{W}_{t-1}^{l-1} \in \mathbb{R}^{U_{l-1} \times U_l}$ is the feature transformation matrix learned through end-to-end training. In particular, the input $\boldsymbol{H}_{t-1}^{(0)}$ is the collection of local features computed at individual stations. $f(A_{t-1})$ is a function on graphs $A_{t-1}$, e.g., the symmetric normalized Laplacian [19] or $k$-order polynomial function of Laplacian [16], and $\sigma$ is a non-linear activation function such as ReLU.

As discussed before, in our case the station network evolves over time, i.e. new/existing stations can be opened or closed at any time. For simplicity, suppose at $t$ there is only one new station $s^N$ has been deployed. To capture this event, we recalculate the inter-station graphs $\boldsymbol{A}_{t-1}$ by appending new rows and columns to them, where the new graphs $\boldsymbol{A}_t$ now contain pairwise correlations between the new $s^+$ and each existing stations. Note that the DGCN input also changes, i.e. now $\boldsymbol{H}_t^{(0)}$ has an extra feature for this newly deployed station $s^N$, computed by the local encoding process.

On the other hand, let $s_j$ be the station that has been closed at time $t$. In our implementation, instead of removing elements from the graphs, we simply apply a mask of zeros to the corresponding rows and columns of $A_t$, and set the $j$-th row of the input $\boldsymbol{H}_t^{(0)}$ to zeros since there won't be local features generated from $s_j$ anymore. The intuition is that in our graph representation, $a_{:,j} = 0$ means station $s_j$ has no correlation with any other station at all, and thus won't be able to propagate information in the graph convolution. Therefore in our case, at different timestamps the dimension of the input to our GCN can be different, i.e., the dimensions of the graphs $\boldsymbol{A}_t$ and input features $\boldsymbol{H}_t^{(0)}$ are varying. However this won't affect the learning process, since the learnable parameters $\boldsymbol{W}_t^l$ at each layer $l$ have fixed dimensions. In addition, note that although $f(A_t)$ produces filters with the same size of the feature $\boldsymbol{H}_t^{(l)}$ at each layer $l$, Eq. (6) can still be viewed as a local convolution given the graphs $\boldsymbol{A}_t$. The reason is that by definition many elements in $\boldsymbol{A}_t$ are near zero (e.g. in the distance graph $A_t^{\mathrm{D}}$), i.e. for a given station, it will be only affected by the features of stations with sufficiently high correlations with it (having large non-zero elements in $\boldsymbol{A}_t$). Conceptually, the dynamic GCN (DGCN) operates on snapshots of the inter-station graphs which are constructed on-the-fly, and fuses the local temporal features at individual stations with the spatial dependencies encoded in those graphs.

# 5 MULTI-SCALE DEMAND PREDICTION

As discussed in Section 2.2, the demand prediction problem addressed in this paper is to forecast the future demand of arbitrary stations in the EV sharing system under the planned expansion, given the historical data and

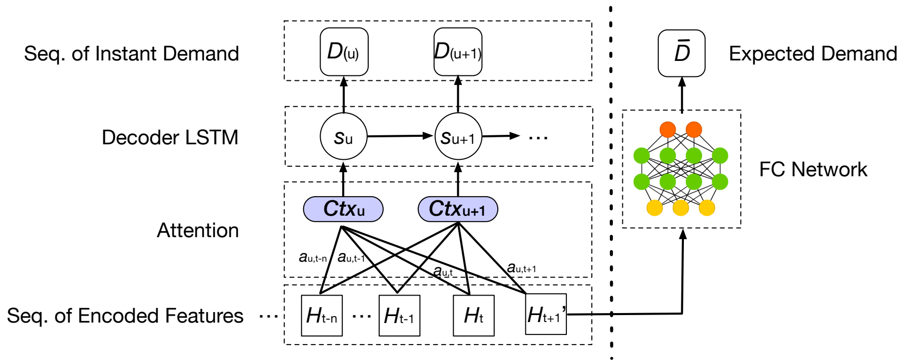



Fig. 5. The proposed multi-scale demand predictor. Left: Decoder LSTM with attention mechanism for instant demand prediction. Here the decoder runs from timestamp $u = [t+1, ..., t+k]$. Right: Fully connected network for expected demand prediction. Note that $\boldsymbol{H}'_{t+1}$ is computed using the planned station network $G^{\mathrm{P}}_{t+1}$, while $\boldsymbol{H}_t$, $\boldsymbol{H}_{t-1}$,... are derived from the actual data until time $t$.

previous dynamics of the station network. We have shown in the previous sections how we use local LSTMs and dynamic GCN (DGCN) to encode the spatial-temporal dynamics of the system, and in this section we explain how to make predictions of the user demand at multiple scales based on the knowledge extracted from the encoding processes. Fig. 5 shows the architecture of the proposed multi-scale demand prediction network.

## 5.1 Predicting Expected Demand

Let $G_t$ be the current station network at time $t$. Without loss of generality, we assume that at the next timestamp we plan to deploy a candidate new station $s^N$, while will close an existing station $s_j$. Therefore, the goal is to predict the future demand of each individual station in this planned station network $G^{\mathrm{P}}_{t+1}$. To achieve that, for each station in $G^{\mathrm{P}}_{t+1}$, we run the LSTMs in the local temporal encoding process (Section 3) to generate an additional feature for time $t+1$, and create the new input feature $\boldsymbol{H}^{(0)'}_{t+1}$ for the GCN. Note that at this moment there is no historical data for the planned new station $s^N$ since it is not deployed yet, and therefore here we only include its static features $\mathbf{c}_{s^N}$ while keeping its temporal features $\mathbf{f}_{s^N}(t+1)$ as zeros. We also mask the row corresponding to station $s_j$ with zeros in $\boldsymbol{H}^{(0)'}_{t+1}$, to mute features from $s_j$ which will be removed at $t+1$. Then we process the planned station network $G^{\mathrm{P}}_{t+1}$ by applying the same update to the inter-station graphs as discussed in Section 4.2, i.e., adding and masking the rows and columns corresponding to $s^N$ and $s_j$. The generated feature $\boldsymbol{H}^{(0)'}_{t+1}$ is then passed through the multi-graph DGCN, producing an output $\boldsymbol{H}'_{t+1}$. We consider this $\boldsymbol{H}'_{t+1}$ as the *context* for prediction, because it not only encodes the current information about the new candidate station $s^N$ and the spatial dependencies between stations, but is also relevant to the available historical information, since the underlying temporal encoding process uses LSTMs to preserve the temporal correlations.

In this paper, we consider the expected demand of station $s_i$ over different days of the week, indicating the mean demand that the station can attract in the future at each week day, i.e., i.e. $\bar{\boldsymbol{d}}_i = [\bar{d}^{\mathrm{Mo}}_i, \bar{d}^{\mathrm{Tu}}_i, ..., \bar{d}^{\mathrm{Su}}_i]$. To predict $\bar{\boldsymbol{d}}_i$, we plug in a fully connected network to the context vector $\boldsymbol{H}'_{t+1}$, which is trained to output the future expected demand (7 values indicating demand on different week days) for each station in the network $G^{\mathrm{P}}_{t+1}$. For the station $s^N$, the predicted expected demand of itself and nearby stations indicate the potential benefits of deploying $s^N$ to the current station network. In Section 6.4 we will show that in real-world experiments our approach significantly outperforms the existing techniques in prediction accuracy.

## 5.2 Predicting Instant Demand

We also predict the future instant demand of stations in the planned network $G_{t+1}^{\mathrm{P}}$ over a certain time window. This is also of great importance in practice, especially for the planned new station $s^N$, since it forecasts the immediate impact and future trends of the station network once $s^N$ is in operation. However it is more challenging than predicting the expected demand, because essentially for each station we need to predict a sequence of concrete demand instead of the aggregated values.

To address that, we design a decoder LSTM network with attention architecture, which takes a sequence of previous features computed by the dynamic multi-graph GCN as input, and estimates the future $k$ instant demand. In this case, conceptually the prediction framework becomes an encoder-decoder structure, where the processes of local temporal encoding and dynamic spatial encoding serve together as the encoder. Let $[\boldsymbol{H}_{t-n}, ..., \boldsymbol{H}_t, \boldsymbol{H}'_{t+1}]$ be the sequence of $n+1$ previous features generated by our DGCN. Unlike in the previous case where we only consider the last output feature $\boldsymbol{H}'_{t+1}$ as the context for prediction, here for each timestamp $u$ in the prediction window of length $k$, i.e. $u = [t+1, ..., t+k]$, we construct the context vectors by fusing the feature sequence with attention mechanism:

$$\mathbf{Ctx}_u = \sum_{v=t-n}^{t+1} \alpha_{uv} \boldsymbol{H}_v \tag{7}$$

where $\alpha_{uv}$ are the attention weights determining the contribution of a feature $\boldsymbol{H}_v$ ($v \in [t-n, t+1]$) in predicting the demand at time $u$. Those weights $\alpha_{uv}$ are trained through back propagation in the end-to-end optimization. Then the decoder LSTM consumes the context vectors and predicts the $k$ subsequent future demand. We found in our experiments that the attention mechanism is very helpful, since the station demand patterns tend to have strong periodic components, e.g., demand on this Monday is highly correlated with previous Mondays, and a single context vector is too compressed to encode such correlation. In our implementation we typically set $n = k$ or $n = 2k$ to better capture such periodical pattern in the station demand.

# 6 EVALUATION

In this section, we evaluate the performance of the proposed data-driven demand prediction approach on data from a real electric vehicle sharing platform in Shanghai, China. We first describe the datasets, baseline approaches and implementation details of our experiments (Sections 6.1, 6.2 and 6.3), and then discuss the experimental results in Section 6.4.

## 6.1 Datasets

*6.1.1 Electric Vehicle (EV) Sharing Data.* Our EV data is collected from real-world operational records of an EV sharing platform for one year (January to December 2017), containing two sets of data: i) the renting/returning orders at each stations, and ii)the detailed expansion process of the station network (i.e., when and where a station was deployed/closed). In particular, there were 1705 stations and 4725 electric vehicles at the beginning of 2017, while as of December 2017 there were 3127 stations with a fleet of 16148 vehicles in operation. In total, the raw data contains 6,843,737 records, which were generated by approximately 0.36 million active users. Fig. 6(a) visualizes the spatial distribution of the orders (represented as lines between pick up and return stations) in a month. Fig. 6(b) shows the numbers of orders at different days-of-month, which exhibit clear periodic patterns with peaks on weekends.

*6.1.2 Simulated System Expansion Data.* In addition to the actual EV data, we also simulate additional datasets for training purposes with different patterns of system expansion. The rationale is that the real expansion data only represents one sample (run) of system expansion, and it is not sufficient for our models to learn how to react to the general expansion process. Specifically, given the real EV data (both orders and expansion data), at

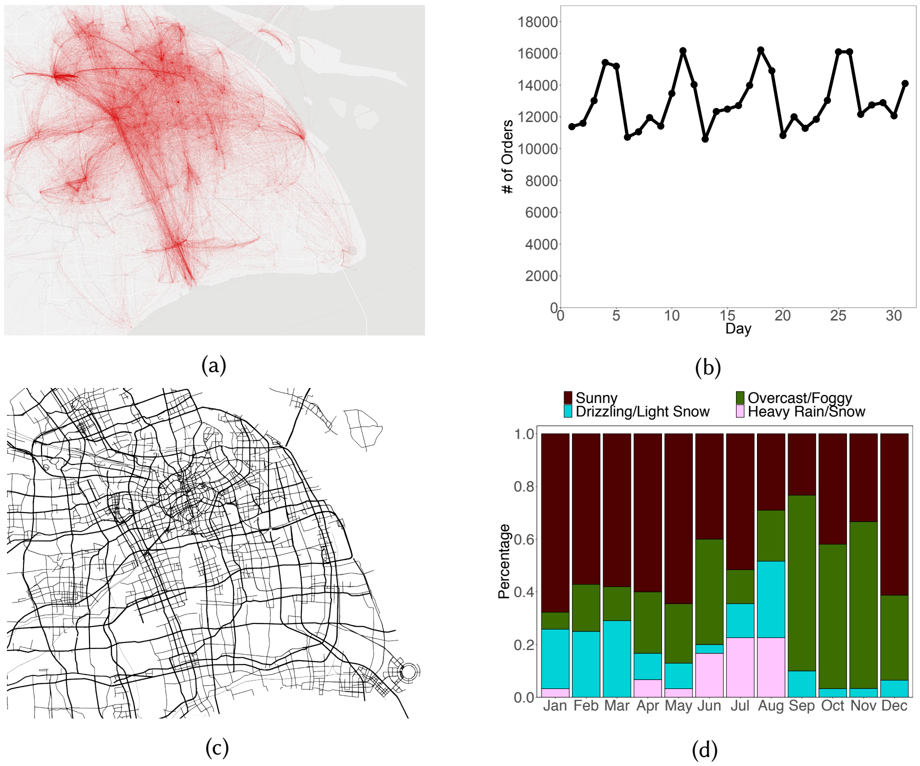



Fig. 6. Visualization of data used in the experiments. (a) Spatial distribution of orders (only showing the most frequent orders over the year). (b) Number of orders in one month. (c) Road network in Shanghai in a graph format. (d) Weather distribution of Shanghai in each month of the year 2017.

each timestamp $t$ we randomly pick a subset of existing stations according to a probability $p$, and treat those stations as newly deployed, i.e., assuming they don't have any previous order data. Note that here we select the stations randomly instead of following certain rules in order to make learning more generalizable. We vary the probability $p$ from 0 to 1, generating multiple simulated datasets ($p = 0.1$, ..., $p = 1$) with different expansion dynamics. Intuitively, the case where $p = 0$ is the real EV data without any extra injected expansion dynamics, while $p = 1$ is the extreme case where all stations at every timestamp are supposed to be newly deployed. Note that we only use the simulated data in training, and for testing we always use the real data. As shown later in Section 6.4, the simulated datasets effectively augment the real EV data, which help the proposed demand prediction approach to generalize better.

*6.1.3 POI Data .* We also collect Point Of Interest (POI) data from an online map service provider [1] in China. In total we have extracted 4,126,844 POI entries in Shanghai, each of which consists of a GPS coordinate and a category label. The label indicates the particular type and function of the POI, e.g. hospitals, subway stations, schools etc. In our experiments, for each station we only consider the POIs within 1km radius. Table. 1 shows the statistics of some POI categories. In our implementation we use one hot vectors to represent the POI features of the stations, i.e. the elements encode the numbers of particular types of POIs which are close to the station.

Table 1. Statistics of some POI categories in our data.

| POI Type | Number | POI Type | Number |
|---|---|---|---|
| Hospitals | 4745 | Banks | 2988 |
| Tourist attractions | 2696 | Companies | 89,747 |
| Gov. organizations | 16,425 | Higher education | 6922 |
| Airport services | 126 | Residences | 51,089 |
| Subway stations | 1,729 | Hotels | 18,234 |
| Bus stations | 41,475 | ... | ... |

*6.1.4 Road Network Data .* We extract road network data in Shanghai using OSMnx [9] from OpenStreetMap [5], which is formatted as a graph (visualized in Fig. 6(c)). Similar with the POIs, we consider the subgraphs within 1km radius of the stations, and compile key statistics such as mean degree, length of road segments etc. into the feature vectors. In our data, on average a subgraph contains road segments of length 13.85km and approximately 39 junctions, with a mean degree of 4.28.

*6.1.5 Meteorology Data .* Finally, we collect the historical daily weather data in Shanghai for the year 2017 from a publicly available source [4]. Each record describes weather conditions of the day, which falls into four different categories: *sunny*, *overcast/foggy*, *drizzling/light snow* and *heavy rain/snow*. Then natually we encode the data using one hot vectors as the weather features. Fig. 6(d) shows the distribution of weather conditions in Shanghai over the 12 months.

## 6.2 Baselines and Metric

We evaluate two variants of the proposed data-driven demand prediction approach respectively: 1) **D$^3$P-Exp**, which predicts the future *expected demand* of stations; and 2) **D$^3$P-Seq**, which forecasts the *instant demand* of stations in a subsequent time window. Both of the two variants share the same local temporal and dynamic spatial encoding processes, but they implement the two different branches in our multi-scale demand predictor and forecast future demand at different scales (as discussed in Section 5).

In particular, for predicting the expected demand, we compare our **D$^3$P-Exp** approach with the following baselines:

- **KNN**, which uses a linear regressor to predict the expected demand of existing stations. For the planned stations, it estimates their demand with standard KNN, based on the similarity of features (e.g. POIs) between them and the existing stations.
- **Random Forest (RF)**, which shares the similar idea as KNN, but trains a random forest as the predictor.
- **Functional Zone (FZ)**, which implements the state of the art demand prediction approach for system expansion in [24]. Note that we don't have taxi records in our data, but instead we directly feed the ground truth check-in/out to favour this approach.

For **D$^3$P-Seq** which computes the instant demand, we consider three competing algorithms:

- **ARIMA + KNN**, which uses Auto-Regressive Integrated Moving Average (ARIMA) [36] to forecast multi-step demand at existing stations, and then uses KNN to estimate demand at new station based on station features such as POIs.
- **LSTM + KNN**, which is similar with A-KNN, but trains LSTM networks for temporal modelling.
- **Multi-graph GCN (MGCN)**, which implements a similar framework as the state of the art in [11], whose implementation is not publicly available. More importantly, the original framework in [11] is not able to

work with time-varying graphs. Therefore, to perform fair comparison, here we use our dynamic multi-graph GCN implementations that can handle new/closed stations, and consider the same data sources as in our approach.

For all approaches, we adopt the *Root Mean Squared Error* (RMSE) and the *Error Rate* (ER) as the performance metric:

$$\begin{aligned} RMSE &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{z}_i - z_i)^2} \\ ER &= \frac{\sum_{i=1}^{N}|\hat{z}_i - z_i|}{\sum_{i=1}^{N} z_i} \end{aligned} \tag{8}$$

where $\hat{z}_i$ and $z_i$ are the predicted and ground truth values respectively.

## 6.3 Implementation Details

We implement the deep neural networks in the proposed approach with TensorFlow [8] 1.10.0, and use the Adam optimiser [18] with the learning rate of 0.001. The networks are trained on a single Titan Xp GPU from scratch. To preserve the temporal dependencies in the data, we partition the data into multiple batches where each of them contains data of consecutive three months. For example, the first batch includes data from January, February and March, while the second has data of February, March and April. For each batch, we use both real and simulated data ($p = 0, ..., p = 1$ as discussed above) from the first two months for training, and the real data of the third month for testing. We train the two branches of our predictor networks separately, where the ground truth labels are obtained from the real world data. We repeat training on all batches and report the best average performance.

## 6.4 Evaluation Results

*6.4.1 Accuracy of Predicting Expected Demand.* The first set of experiments evaluate the overall accuracy when predicting the expected demand of stations. Fig. 7(a) and (b) show the RMSE and ER of the proposed approach ($D^3$P-Exp) and competing algorithms over different days of the week. We see that comparing to naive KNN, the random forest based approach (RF) can reduce the RMSE by about 30% while ER by 20%. However, our approach ($D^3$P-Exp) performs significantly better, and can achieve up to three times improvement in both RMSE and ER. In particular, on average the RMSE of $D^3$P-Exp is approximately 1.961, which means when predicting a station's expected demand, the value estimated by our approach is only about ±2 with respect to the ground truth. This confirms that the proposed approach can effectively model the complex temporal and spatial dependencies within the evolving station network, and exploits that to make more accurate predictions. In addition, we observe that the RMSE tends to increase on weekends compared to weekdays for all algorithms. This is because in practice the absolute demand on weekends is larger, which often leads to bigger RMSE. Note that the ER remains relatively consistent across different days.

*6.4.2 Planned vs. Existing Stations.* This experiment investigates the prediction performance of different approaches on the planned new stations which haven't been deployed yet, and existing stations which are already been in operation. Fig. 7(c) and (d) show the average RMSE and ER of the proposed approach ($D^3$P-Exp) and the competing algorithms on the planned, existing, and all stations respectively. We see that all of approaches perform better on the existing stations than the planned. This is expected because for existing stations we have access to their historical demand data, which is not available for planned stations. We also observe that although the functional zone based approach (FZ) performs better than the baselines for the planned stations, it fails on the existing stations (performs worse than RF). This is because by design FZ is tuned to predict demand of new stations in the context of system expansion, but not for existing ones. Finally, we see that for both planned and

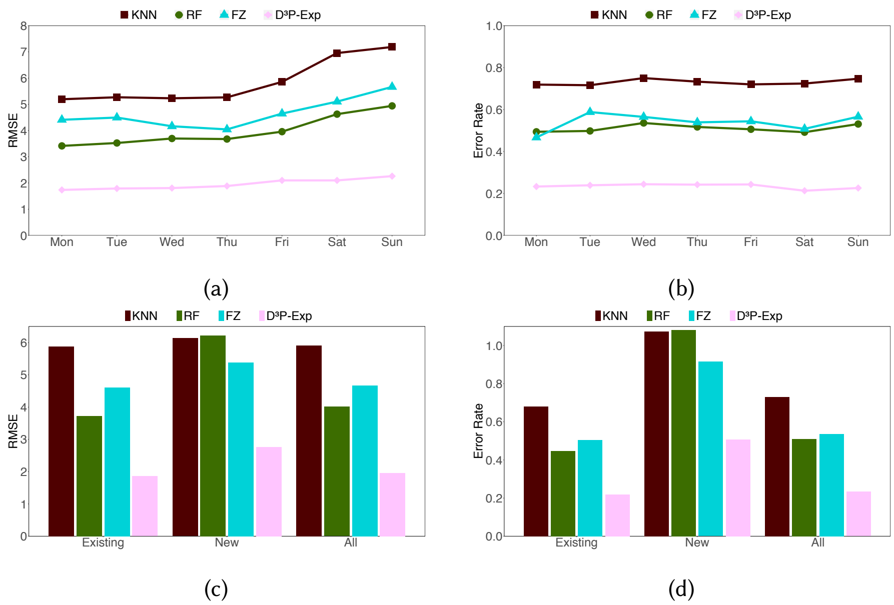



Fig. 7. Performance on predicting the expected demand. (a) RMSE and (b) ER of all stations across different days in the week. (c) RMSE and (d) ER of existing vs. newly deployed stations vs. all stations averaged over all days of the week.

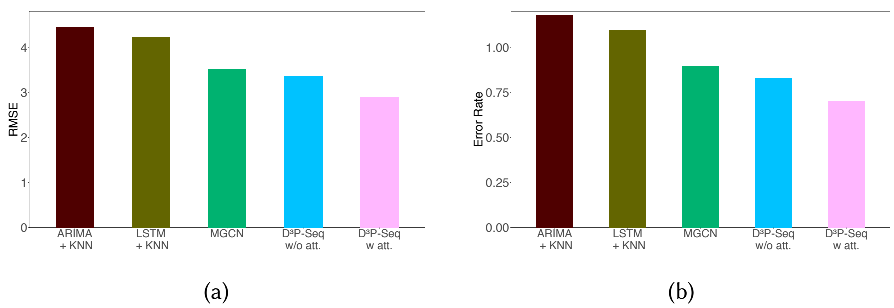
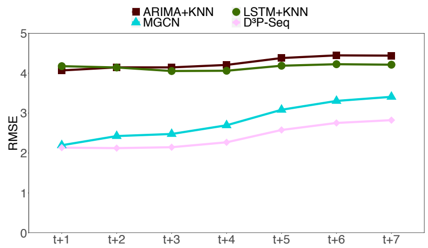


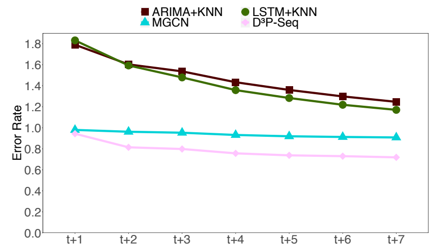

Fig. 8. Performance on predicting the instant demand. (a) RMSE and (b) ER of the competing approaches.

existing stations our approach (D$^3$P-Exp) performs consistently the best. For the planned stations, it halves the errors comparing to the state of the art approach FZ, while for the existing stations, it offers about three-fold improvement over the baselines.

*6.4.3 Accuracy of Predicting Instant Demand.* This set of experiments evaluates the performance of different approaches when predicting the future instant demand. Here we only consider the planned stations, since it is straightforward to predict for the existing stations given their historical data. We ask all approaches to predict the instant demand over the next seven days, and report the average accuracy. Fig. 8 shows the RMSE and ER of the proposed approach (D$^3$P-Seq) and the competing algorithms. We see that in this challenging case, our approach (D$^3$P-Seq) can still achieve an average RMSE of 2.903, which is over 30% lower than the baselines (similar gap can be observed in ER). It is also superior to the state of the art MGCN approach which also uses




(a)




(b)

Fig. 9. (a) RMSE and (b) ER of the predicted instant demand for different prediction lengths.

multi-graph GCN, with about 20% reduction in RMSE and ER. This confirms that even for the planned stations without historical data, our approach can still accurately predict their future instant demand within a certain time window. In addition, we find that the attention mechanism in our approach is very effective. Without using attention architecture in the decoder, the performance of our approach drops by approximately 15%, which is still better than the state of the art.

*6.4.4 Accuracy vs. Prediction Length.* This experiment studies the accuracy of competing approaches when predicting instant demand over different time intervals. As in the previous experiment, here we also only consider prediction performance for the planned station. We vary the length of the prediction time window from 1 to 7, i.e. from predicting the demand of stations on the immediate next day $t + 1$, to that on the subsequent seven days $t + 7$. Fig. 9 shows the RMSE and ER of the approaches under different time windows. We observe that in general, the RMSE increases as the length of the time window grows, especially for our approach (D$^3$P-Seq) and the state of the art MGCN. This makes sense because clearly predicting demand over a longer time window is more difficult. On the other hand, we see that the ER of baselines are higher for short window lengths comparing to the MGCN or our approach. We find that this is because the baselines tend to report random estimations on the future demand, where for shorter windows this can lead to larger ER, but will be averaged out for longer time windows as the ground truth demand grows in later days. Finally, we see that MGCN can offer comparable performance with our approach (D$^3$P-Seq) when predicting for the immediate next timestamp. However as the prediction length increases, our approach consistently outperforms MGCN, with a performance gap of up to 26%.

*6.4.5 Impact of Different System Expansion Dynamics.* The last set of experiments investigates the impact of different levels of system expansion dynamics on the proposed demand prediction approach, and the validity of using the simulated data for training. As discussed in Section 6.3, the reason why we use simulated data in addition to real data for training is that the real data only represents one sample (or run) of system expansion, which is not sufficient for our models to pick up the general expansion process. Therefore, we simulate more datasets by randomly selecting a subset of existing stations according to a probability $p$, and assume those stations as newly deployed. This allows us to generate datasets with different levels of extra injected expansion dynamics. In particular, we varied $p$ from 0 to 1, where $p = 0$ is the real data without any extra dynamics, and $p = 1$ leads to the extreme case where all the stations are considered as newly deployed at each timestamp. Therefore, we train our models with both the real ($p = 0$) and simulated ($p = [0.1, ..., 0.9]$) data, but in all experiments we evaluated our approach with the real data. As shown in Fig. 10, we see that as $p$ increases from zero, our approach tends to make more accurate predictions for both expected and instant demand (lower errors). This confirms the validity of our simulation approach, in that by artificially injecting the simulated dynamics, we essentially force the GCN to learn how to better react to the deployment of new stations. We also observed that for larger $p$ values, the

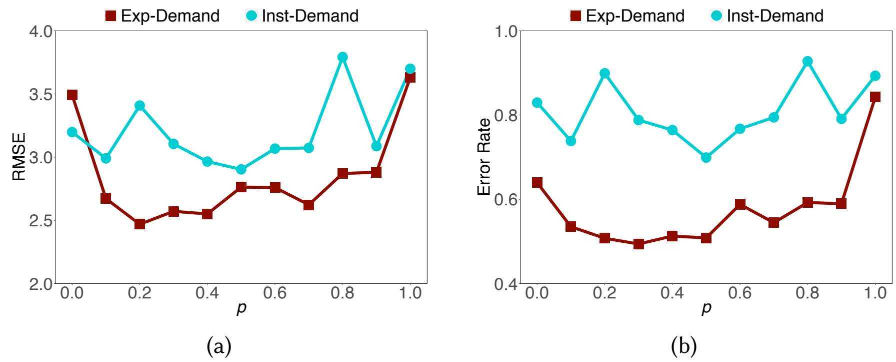



Fig. 10. Prediction performance of our approach trained on datasets with different levels of system expansion dynamics. (a) RMSE and (b) ER.

errors (both RMSE and ER) tend to increase for both types of demand. This is also expected because in those cases the excessive injected dynamics would mute the useful information coming from local LSTMs at individual stations and confuse the GCN, leading to deterioration of performance. Therefore, this means with carefully selected $p$, the simulated data can help the learning process to capture the dynamics caused by system expansion, leading to more accurate predictions. Empirically we find that $p$ with values around 0.4∼0.6 would achieve the desired balance between incorporating the historical information and learning from the expansion dynamics.

## 7 RELATED WORK

### 7.1 Electric Vehicle (EV) Systems

The electrification of urban mobility systems has introduced a number of new challenges, such as route planning and optimization [29], vehicle charging scheduling [33, 38, 42], and infrastructure planning [29]. The work in [32] conducted a comprehensive measurement investigation to study the long-term evolving mobility and charging patterns of electric taxis in a city, using real-world data collected over five years. Our work complements those existing studies which primarily consider the electric taxis or buses, in that we focus on the EV sharing systems, which operate in a very different way. For instance, for electric taxis or buses their networks of stations [32] are mainly used for charging, whose service coverage also depends on the fixed bus routes or individual taxi drivers. However in our case, the users can only access (renting, returning and charging) vehicles at the available stations, and thus the expansion of the station network will have much more direct and complex impact on the entire system.

### 7.2 Demand Prediction for Shared Mobility Services

Predicting user demand in shared mobility services (e.g. taxis and bike- or vehicle-sharing systems) has received considerable interest in various research communities. Most of the existing work takes the historical usage (e.g. picking-up and returning records), geospatial data such as POIs, and other auxiliary information (e.g. weather) into account, and builds prediction models that can forecast demand over certain periods or aggregated time slots. They also predict the demand at different spatial granularity, e.g. over the entire systems [35, 41], grids/regions [16], station clusters [15, 22, 27], or individual stations [11–13, 17, 23, 34, 39, 43]. This paper falls into the last category since we aim to predict station-level demand of EV sharing systems. However, our work is fundamentally different in that we assume the station network is not static, but dynamically evolving, i.e. stations can be deployed or closed at arbitrary times. This has not been investigated by the existing work, and in this case, state of the art station-level demand predictors (e.g. [17]) do not address our problem because they rely heavily on station historical data to make predictions, which are not available for those newly deployed stations.

### 7.3 Shared Mobility System Expansion

There is also a solid body of work focusing on modeling the expansion process of shared mobility systems, e.g. planning for optimal new stations [12, 23, 37], or increasing the capacity of existing stations [14]. However, most of the existing work assumes that historical information on demand of the stations (renting and returning) are either known, or can be estimated from other data sources such as taxi records, which is fundamentally different from our work. On the other hand, the work in [24] proposes a functional zone based hierarchical demand predictor for shared bike systems, which can estimate the average demand at newly deployed stations across different expansion stages. Our work shares similar assumptions with [24], yet differs substantially: 1) instead of fixed stages, we can predict demand while the entire station network is dynamically expanding; 2) we are able to estimate both the instant and expected demand of new or existing stations, while [24] can only predict aggregated demand patterns; and finally 3) we do not require historical mobility data in the newly expanded areas, like the taxi trip records required in [24].

### 7.4 Urban Computing and Analysis

The demand prediction problem for EV sharing studied in this paper also falls in the broader area of urban computing, which investigates the acquisition, integration, and analysis of the heterogeneous and large amount of data generated by various sources in urban spaces [45]. In that sense, our work shares similar challenges and settings with the other urban computing and analysis problems, such as crowd prediction [31, 44], traffic analysis [20], anomalies detection [28, 46], etc. However, this paper is fundamentally different from those work. First of all, most of the existing work considers events or data generated across fixed 2D space or road networks, while our problem aims to understand the future demand at the EV stations within a station network, which is a graph dynamically growing over time. In addition, the technical approaches proposed in our paper is also different from those considered in the existing work. We consider a novel dynamic GCN approach that can handle the time-varying graph of the growing EV station network), while most of the recent work, e.g. those on crowd prediction [31, 44] assume the space (modeled as graphs where different regions are nodes) is stationary. Essentially, the graphs considered in the crowd prediction work can be viewed as static containers of the crowd, as they represent regions of the space. Thus their approaches cannot be applied to our demand prediction problem where the graphs themselves are variables evolving over time.

### 7.5 Graph-based Deep Learning

Due to their non-Euclidean nature, many real-world problems such as demand/traffic/air quality forecasting that require spatio-temporal analysis have been tackled with the emerging graph-based deep learning techniques [11, 16, 21, 25, 26, 40]. In particular, existing work often employs the graph convolutional neural network [10] to capture the spatial correlations, where temporal dependencies are typically modelled with recurrent neural networks. For instance, [21] models the traffic flow as a diffusion process on directed graphs for traffic forecasting, while [40] and [16] propose frameworks that use multi-graph convolutional neural networks (CNNs) to predict demand for taxi and ride-hailing services. Another work in [11] uses an encoder-decoder structure on top of multi-graph CNNs to estimate flow between stations in bike sharing systems, which bears a close resemblance to this paper. However, unlike [11] who only output demand at the immediate next timestamp, our work considers a sequence to sequence model with attention mechanism to perform multi-step forecasting towards future demand. In addition, none of the above approaches can work on new stations where historical data is not available, i.e. they do not address continuous system expansion.

## 8 CONCLUSION AND FUTURE WORK

In this paper, we propose a data-driven demand prediction approach for continuously growing Electric Vehicle (EV) sharing systems. The proposed approach is able to capture the complex spatial and temporal dynamics from the system expansion, and can predict demand for both existing stations and the planned new stations. To achieve that, we first encode the local temporal information at the individual station level, and then fuse the extracted features with a novel Dynamic Graph Convolutional Neural Network (DGCN) to take the spatial dependencies between different stations into account. The demand of stations is then estimated by a multi-scale prediction network, which forecasts both the long-term expected demand and the instant future demand. We evaluate our approach on data collected from a real-world EV sharing platform in Shanghai for a year. Extensive experiments on real and simulated data have shown that our approach consistently and substantially outperforms the state of the art in predicting both the long-term expected and the immediate future demand of the expanding system. Our proposed method may help businesses or public operators to reliably evaluate planned system changes and expansions, offering valuable decision support. For future work, we would like to study the effectiveness and generalizability of our approach in more cities, and explore new approaches that can transfer the learned knowledge to new settings.

## ACKNOWLEDGMENTS

This work was supported in part by The Alan Turing Institute under the EPSRC grant EP/N510129/1.

## REFERENCES

[1] 2019. AMap API. https://lbs.amap.com/api. Accessed: 2019-04-29.
[2] 2019. Bluecity Car Sharing. https://www.blue-city.co.uk/. Accessed: 2019-04-29.
[3] 2019. BlueSG. https://www.bluesg.com.sg/. Accessed: 2019-04-29.
[4] 2019. Juhe API. https://www.juhe.cn/docs/api/id/277. Accessed: 2019-04-29.
[5] 2019. OpenStreetMap API. https://www.openstreetmap.org. Accessed: 2019-04-29.
[6] 2019. Volkswagen starts 'We Share' e-mobility car sharing in Berlin. https://www.volkswagenag.com/en/news/2018/08/VW_Brand_We_Share.html. Accessed: 2019-04-29.
[7] 2019. Zipcar Flex. https://www.zipcar.co.uk/electric. Accessed: 2019-04-29.
[8] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. TensorFlow: A System for Large-scale Machine Learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation (OSDI'16)*. USENIX Association, Berkeley, CA, USA, 265–283. http://dl.acm.org/citation.cfm?id=3026877.3026899
[9] Geoff Boeing. 2017. OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems* 65 (2017), 126 – 139. https://doi.org/10.1016/j.compenvurbsys.2017.05.004
[10] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203* (2013).
[11] Di Chai, Leye Wang, and Qiang Yang. 2018. Bike flow prediction with multi-graph convolutional networks. In *Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 397–400.
[12] Longbiao Chen, Daqing Zhang, Gang Pan, Xiaojuan Ma, Dingqi Yang, Kostadin Kushlev, Wangsheng Zhang, and Shijian Li. 2015. Bike sharing station placement leveraging heterogeneous urban open data. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 571–575.
[13] Longbiao Chen, Daqing Zhang, Leye Wang, Dingqi Yang, Xiaojuan Ma, Shijian Li, Zhaohui Wu, Gang Pan, Thi-Mai-Trang Nguyen, and Jérémie Jakubowicz. 2016. Dynamic cluster-based over-demand prediction in bike sharing systems. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 841–852.
[14] Bowen Du, Yongxin Tong, Zimu Zhou, Qian Tao, and Wenjun Zhou. 2018. Demand-Aware Charger Planning for Electric Vehicle Sharing. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1330–1338.
[15] Jon Froehlich, Joachim Neumann, Nuria Oliver, et al. 2009. Sensing and predicting the pulse of the city through shared bicycling.. In *IJCAI*, Vol. 9. 1420–1426.

[16] Xu Geng, Yaguang Li, Leye Wang, Lingyu Zhang, Qiang Yang, Jieping Ye, and Yan Liu. 2019. Spatiotemporal Multi-Graph Convolution Network for Ride-hailing Demand Forecasting. In *2019 AAAI Conference on Artificial Intelligence (AAAI'19)*.

[17] Pierre Hulot, Daniel Aloise, and Sanjay Dominik Jena. 2018. Towards Station-Level Demand Prediction for Effective Rebalancing in Bike-Sharing Systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 378–386.

[18] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[19] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*.

[20] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).

[21] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations (ICLR'18)*.

[22] Yexin Li, Yu Zheng, Huichu Zhang, and Lei Chen. 2015. Traffic prediction in a bike-sharing system. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 33.

[23] Junming Liu, Qiao Li, Meng Qu, Weiwei Chen, Jingyuan Yang, Hui Xiong, Hao Zhong, and Yanjie Fu. 2015. Station site optimization in bike sharing systems. In *Data Mining (ICDM), 2015 IEEE International Conference on*. IEEE, 883–888.

[24] Junming Liu, Leilei Sun, Qiao Li, Jingci Ming, Yanchi Liu, and Hui Xiong. 2017. Functional zone based hierarchical demand prediction for bike system expansion. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 957–966.

[25] Man Luo, Hongkai Wen, Yi Luo, Bowen Du, Konstantin Klemmer, and Hongming Zhu. 2019. Demand Prediction for Electric Vehicle Sharing. *arXiv preprint arXiv:1903.04051* (2019).

[26] Yi Luo, Qin Liu, Hongming Zhu, Hongfei Fan, Tianyou Song, Chang Wu Yu, and Bowen Du. 2019. Multistep Flow Prediction on Car-Sharing Systems: A Multi-Graph Convolutional Neural Network with Attention Mechanism. (2019).

[27] Eoin O'Mahony and David B Shmoys. 2015. Data Analysis and Optimization for (Citi) Bike Sharing.. In *AAAI*. 687–694.

[28] Bei Pan, Yu Zheng, David Wilkie, and Cyrus Shahabi. 2013. Crowd sensing of traffic anomalies based on human mobility and social media. In *Proceedings of the 21st ACM SIGSPATIAL international conference on advances in geographic information systems*. ACM, 344–353.

[29] Ankur Sarker, Haiying Shen, and John A. Stankovic. 2018. MORP: Data-Driven Multi-Objective Route Planning and Optimization for Electric Vehicles. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 162 (Jan. 2018), 35 pages. https://doi.org/10.1145/3161408

[30] Susan Shaheen, Adam Cohen, and Mark Jaffee. 2018. Innovative Mobility: Carsharing Outlook. (2018).

[31] Junkai Sun, Junbo Zhang, Qiaofei Li, Xiuwen Yi, and Yu Zheng. 2019. Predicting Citywide Crowd Flows in Irregular Regions Using Multi-View Graph Convolutional Networks. *arXiv preprint arXiv:1903.07789* (2019).

[32] Guang Wang, Xiuyuan Chen, Fan Zhang, Yang Wang, and Desheng Zhang. 2019. Experience: Understanding Long-Term Evolving Patterns of Shared Electric Vehicle Networks. In *MobiCOM*.

[33] Guang Wang, W. Li, J. Zhang, Y. Ge, Z. Fu, F. Zhang, Y. Wang, and Desheng Zhang. 2019. sharedCharging: Data-Driven Shared Charging for Large-Scale Heterogeneous Electric Vehicles. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* (May 2019).

[34] Shuai Wang, Tian He, Desheng Zhang, Yuanchao Shu, Yunhuai Liu, Yu Gu, Cong Liu, Haengju Lee, and Sang H Son. 2018. BRAVO: Improving the rebalancing operation in bike sharing with rebalancing range prediction. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 44.

[35] Wen Wang. 2016. *Forecasting Bike Rental Demand Using New York Citi Bike Data.* Master's thesis. Dublin Institute of Technology.

[36] Billy M Williams and Lester A Hoel. 2003. Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: Theoretical basis and empirical results. *Journal of transportation engineering* 129, 6 (2003), 664–672.

[37] Yanhai Xiong, Jiarui Gan, Bo An, Chunyan Miao, and Ana LC Bazzan. 2015. Optimal Electric Vehicle Charging Station Placement.. In *IJCAI*. 2662–2668.

[38] Li Yan, Haiying Shen, Zhuozhao Li, Ankur Sarker, John A. Stankovic, Chenxi Qiu, Juanjuan Zhao, and Chengzhong Xu. 2018. Employing Opportunistic Charging for Electric Taxicabs to Reduce Idle Time. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 47 (March 2018), 25 pages. https://doi.org/10.1145/3191779

[39] Zidong Yang, Ji Hu, Yuanchao Shu, Peng Cheng, Jiming Chen, and Thomas Moscibroda. 2016. Mobility modeling and prediction in bike-sharing systems. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 165–178.

[40] Huaxiu Yao, Fei Wu, Jintao Ke, Xianfeng Tang, Yitian Jia, Siyu Lu, Pinghua Gong, Jieping Ye, and Zhenhui Li. 2018. Deep Multi-View Spatial-Temporal Network for Taxi Demand Prediction. In *2018 AAAI Conference on Artificial Intelligence (AAAI'18)*.

[41] Yu-Chun Yin, Chi-Shuen Lee, and Yu-Po Wong. 2014. Demand Prediction of Bicycle Sharing Systems. (2014).

[42] Yukun Yuan, Desheng Zhang, Fei Miao, Jiming Chen, Tian He, and Shan Lin. 2019. $p^2$ Charging: Proactive Partial Charging for Electric Taxi Systems. In *ICDCS*.

[43] Ming Zeng, Tong Yu, Xiao Wang, Vincent Su, Le T Nguyen, and Ole J Mengshoel. 2016. Improving Demand Prediction in Bike Sharing System by Learning Global Features. *Machine Learning for Large Scale Transportation Systems (LSTS)@ KDD-16* (2016).

[44] Junbo Zhang, Yu Zheng, Dekang Qi, Ruiyuan Li, Xiuwen Yi, and Tianrui Li. 2018. Predicting citywide crowd flows using deep spatio-temporal residual networks. *Artificial Intelligence* 259 (2018), 147–166.

[45] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. 2014. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 3 (2014), 38.

[46] Yu Zheng, Huichu Zhang, and Yong Yu. 2015. Detecting collective anomalies from multiple spatio-temporal datasets across different domains. In *Proceedings of the 23rd SIGSPATIAL international conference on advances in geographic information systems*. ACM, 2.