Mobility Modeling and Prediction in Bike-Sharing Systems

Zidong Yang[†], Ji Hu[†], Yuanchao Shu[‡*], Peng Cheng[†], Jiming Chen[†], Thomas Moscibroda[‡]

[†] Zhejiang University, Hangzhou, China [‡] Microsoft Research Asia

ABSTRACT

As an innovative mobility strategy, public bike-sharing has grown dramatically worldwide. Though providing convenient, low-cost and environmental-friendly transportation, the unique features of bike-sharing systems give rise to problems to both users and operators. The primary issue among these problems is the uneven distribution of bicycles caused by the ever-changing usage and (available) supply. This bicycle imbalance issue necessitates efficient bike re-balancing strategies, which depends highly on bicycle mobility modeling and prediction. In this paper, for the first time, we propose a spatio-temporal bicycle mobility model based on historical bike-sharing data, and devise a traffic prediction mechanism on a per-station basis with sub-hour granularity. We extensively evaluated the performance of our design through a one-year dataset from the world's largest public bike-sharing system (BSS) with more than 2800 stations and over 103 million check in/out records. Evaluation results show an 85 percentile relative error of 0.6 for both check in and check out prediction. We believe this new mobility modeling and prediction approach can advance the bike re-balancing algorithm design and pave the way for the rapid deployment and adoption of bike-sharing systems across the globe.

Keywords

Sharing economy; Bike sharing; Mobility modeling; Flow prediction; Rebalancing

1. INTRODUCTION

Shared transportation has grown tremendously in recent years as a result of the rise of the sharing economy and growing environmental, energy and economic concerns. Among the various forms of shared-use mobility[1], public *bike-sharing systems* (BSS) have become increasingly popular with a significant growing presence over the past decade. Available figures indicate that there are more than 500 bike-sharing programs running in at least 49 countries with one million shared bikes in 2015 [1, 2].

In addition to its advantages of reducing traffic congestion and mitigating pollution, BSS feature unique characteristics compared with other forms of shared-use mobility. First, bike-sharing differs from classic ride-sharing (e.g., carpooling) and ride-sourcing (e.g., Uber and Lyft) in that bicycles are typically *unattended*. During vacant hours, bicycles are *concentrated* at a group of stations where operations of checking in or checking out are facilitated through a backbone network, i.e., an IT infrastructure that enables rent management and monitoring. Second, unlike conventional public transit (e.g., subways and buses) which follows a regular schedule and pre-determined routes, bike-sharing provides transportation on an *on-demand* basis with a *decentralized* structure. These two distinct features, however, pose characteristic challenges in BSS management and optimization. One common problem, for example, is that the system typically ends up with an uneven distribution of bikes across the different stations (due to the uncontrolled, uneven demand), often rendering the check in or check out service unavailable at some stations where bicycle docks are either fully occupied or empty.

This bicycle imbalance problem makes it necessary for bike-share cities to employ costly *bike redistribution*, which is typically performed by trucks or trailers driving around the city, moving bikes among stations. To increase service availability and minimize redistribution cost, studies have been conducted to improve these bike redistribution strategies based on bicycle mobility models and predictions. Yet, in spite of the researches on bike usage patterns and global rental volume forecasts (e.g., [3–7]), developing a fine-grained and localized prediction model for the number of bikes that should be optimally redistributed has proven to be elusive, and has remained a largely unstudied problem. The main technical challenge is that bike traffic is not only highly dynamic and inter-correlated in both the temporal and spatial domains, but also further influenced by complex issues such as timing and meteorology.

In this paper, we establish a spatio-temporal mobility model of bikes, and present a novel fine-grained traffic (i.e., check in and check out) prediction mechanism on a per-station basis by leveraging historical bike-sharing data as well as meteorological data. Our work differs fundamentally from previous approaches in that we model BSS as a dynamic network and predict the traffic by jointly considering the spatio-temporal correlations between stations and additional time factors and meteorology. To this end, we first decouple the bicycle transitions between stations from check out actions based on the counterbalance of bicycles and the mutual independence of user behaviors. Based on historical

[*] Yuanchao Shu is the corresponding author of this paper.

[1] Shared-use mobility–the shared use of transportation services–is an innovative transportation solution that enables users to have short-term access to transportation modes. It includes traditional public transit, bike-sharing, car-sharing, ride-sharing, ride-sourcing etc.

MobiSys'16, June 25-30, 2016, Singapore, Singapore

© 2016 ACM. ISBN 978-1-4503-4269-8/16/06. . . $15.00

DOI: http://dx.doi.org/10.1145/2906388.2906408

data, we then use a probabilistic model to describe the spatio-temporal movements of bikes within the network, and estimate the number and time of check in at different stations. Combined with a random-forest-based check out prediction algorithm, we are able to estimate the number of docked bicycles at each station at any given time period in the future, which lays a foundation for efficient redistribution strategy design.

This paper makes the following three main contributions:

- We identify the mobility modeling problem and establish a spatio-temporal dynamic network model for BSS by taking into account the interactions among all stations;
- To our knowledge, we conduct the first work to devise a traffic prediction mechanism on a per-station basis with sub-hour granularity by using the mobility model and historical data;
- We evaluate the performance of mobility modeling and prediction through a one-year dataset from the city of Hangzhou, the world's largest public BSS with more than 2800 stations and over 103 million check in/out records [8, 9]. Compared with two benchmark methods, the proposed approach provides the best performance with an 85 percentile relative error of 0.6 for both check in and check out prediction.

The remainder of this paper is organized as follows. We first provide an overview of our design in Section 2. We then present the mobility model in Section 3, followed by detailed illustration of bicycle check out prediction in Section 4. Section 5 describes our datasets, and Section 6 presents an in-depth evaluation of mobility modeling and prediction. Several insights and related work are discussed in Section 7 and Section 8. We conclude the paper in Section 9.

2. DESIGN OVERVIEW

This section provides a design overview, including problem formulation in Section 2.1 and design methodology in Section 2.2.

2.1 Problem Formulation

Two types of entities constitute a BSS system (see Figure 1): *Active objects* (users) and *Reactive objects* (bikes), respectively. Users shift bikes from check in to check out operations, changing the status (i.e., the number of docked bicycles) of stations located at different places. Conversely, the spatial diversity of stations and bike availabilities also influence user behaviors. In this paper, we call a sequence of operations - bicycle check out, movement and check in - a *shift instance* (SI). As can be seen from Figure 1, *Active objects* and *Reactive objects* are coupled in both the temporal and spatial domain. However, it is worth noting that user activities in *Active objects* are *mutually independent*, though subject to the change of time factors and meteorology.

The objective of this paper is two-fold. First, we aim to model the mobility patterns of bikes in BSS. The mobility model characterizes the spatio-temporal transition of bikes among stations. Second, based on the mobility model, we aim to predict the number of docked bicycles at each station at a given time in the future. Due to the correlation between *Active objects* and *Reactive objects* as well as among stations, our design methodology proceeds from a decoupling operation, which disentangles these various correlations.

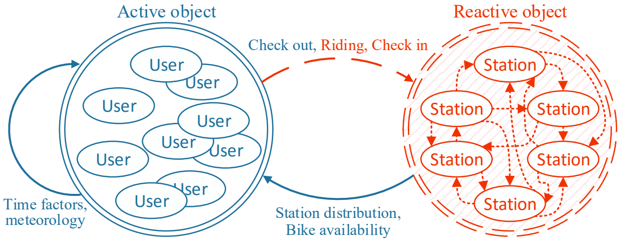


Figure 1: Components of a bike-sharing system.

2.2 Design Methodology

Despite the random bicycle check out time and location in each SI, bicycles are bound to be checked in at some station. Based on this simple observation, we decouple the system in Figure 1 into two parts (i.e., the left blue part and the right red part with dashed lines) by modeling the mobility of undocked bicycles and check out behaviors separately. Specifically, based on historical check out data, we first use a probabilistic model to describe the spatio-temporal movements of the undocked bikes, and then estimate the number and time of check in at different stations (see Section 3). We then apply the *random forest theory* to model and predict the users' check out behaviors with a joint consideration of time factors, meteorology and real-time bike availability (see Section 4). Both estimation and prediction results are updated in an online manner, and therefore can be integrated into any existing real-world BSS and used to infer and predict the number of docked bicycles at each station in real time.

3. BICYCLE MOBILITY MODELING

In this section, we develop a mobility model to capture the spatio-temporal transition of bikes. The model is established based on the uniqueness of bike flows between different pairs of stations at different times. For example, bicycles flow into stations in working areas in the morning on weekdays and flow out in the afternoon. Even for a random bicycle check out at a given station and a given time, the probabilities of the corresponding check in at different stations vary. Therefore, we propose a statistical model which uses historical check in/out data to describe the spatio-temporal shifts of bikes between pairs of stations, and estimate bicycle check in based on the online check out records. Compared with state-of-art approaches [10, 11] that treat each station independently, our model has the following two advantages:

- **Fine-grained Modeling.** A fine-grained mobility model with time-varying parameters is built based on the analysis of historical bicycle flows between stations. Check in estimations at each station can be obtained using this model.
- **Online Updating.** The estimation results are continuously updated based on recent check in and check out data, adapting to evolving user behaviors and network settings.

3.1 Theoretical Mobility Model

We first present a theoretical mobility model which estimates the check in numbers based on the analysis of historical SI data. Notations are defined in Table 1.

Without loss of generality, we consider the check in estimation of some station i. Given previous check out records (i.e., bikes checked out before t_{now}), we aim to quantify bikes that will be checked in at station i during a target period $[t, t+\Delta]$ in the future.

Table 1: Notations of the mobility model.

Notation	Description
$A_i(t, \Delta)$	Number of bikes check in to station i during $[t, t+\Delta]$
$D_i(t, \Delta)$	Number of bikes check out from station i during $[t, t+\Delta]$
$\gamma_{ji}(t)$	The probability of bikes that check out from station j at time t will check in to station i
$F_{ji}(\tau)$	The cumulative distribution function (CDF) of trip duration from station j to station i
N	The collection of stations

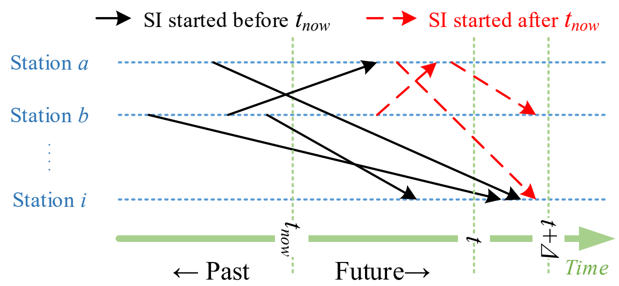


Figure 2: Basic idea of check in estimation.

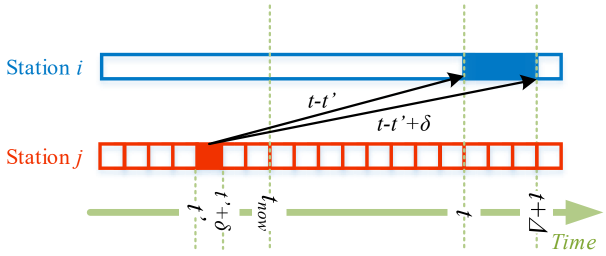


Figure 3: Traffic analysis between two stations.

To better illustrate the main idea of the estimation algorithm, we use an example shown in Figure 2. As can be seen in the figure, all bikes checked in at station i during $[t, t+\Delta]$ can be classified into two categories: bikes departed from all stations i) before t_{now} (i.e., black solid lines), and ii) after t_{now} (i.e, red dashed lines). In this section, we focus on the first category and illustrate how to quantify the bicycles arrived at station i during $[t, t+\Delta]$ from the entire group of bikes that are checked out before t_{now}. For those departed after t_{now}, the same approach can be applied once we perform the check out estimation for $[t_{now}, t+\Delta]$. The detailed algorithms of check out prediction will be presented in Section 4.

In general, the principal relationship between check in and check out can be written as

$$A_i = \sum_{j \in N} D_j \Gamma_{ji} P_t \tag{1}$$

where A_i is the number of bikes checking in to station i while D_j is the number of bikes checking out from station j. Since bikes from station j may have different destinations, Γ_{ji} is used to denote the transfer probability from station j to station i. Trip duration is another important factor in our model. The check in time may fall out of the target period if the trip duration is too short or too long. Thus, P_t is used to denote the probability that the bike will check in to station i within the target period.

Based on Equation 1, we derive a theoretical model for check in estimation. First consider SIs start from station j and end at station i. As discussed above, the value of P_t is determined by the trip duration. However, if a bicycle ends up in the destination station during the target period, the feasible range of the trip duration is subject to the check out time. For example, if the target period is $[10\colon 00 a.m., 10\colon 30 a.m.]$, the feasible trip duration for bikes checked out at 9:30 a.m. is 30~60 minutes while it is 45~75 minutes for bikes checked out at 9:15 a.m.

To cope with this problem, we perform *temporal discretization* and adopt a unified P_t for each time interval. Thus, we are able to enumerate SIs initiated in each time slot and add them up to obtain the aggregated bike flow between two stations.

As shown in Figure 3, if δ is sufficiently small, for bikes checked out from station j during $[t', t'+\delta]$, if they check in to station i within time window $[t, t+\Delta]$, the trip duration should be within $[t-t', t+\Delta-t']$. In addition, the transfer probability $\gamma_{ji}(\tau)$ can be viewed as a constant during $[t', t'+\delta]$. Then, the number of bikes from station j to station i which check in during $[t, t+\Delta]$ can be computed by adding up SIs in each time slot as

$$n_{ji} = \sum_{k=1}^{\infty} D_j(t_k, \delta)\gamma_{ji}(t_k)(F_{ji}(t+\Delta-t_k) - F_{ji}(t-t_k)) \tag{2}$$

where $[t_k, t_k+\delta]$ is the k-th interval before t_{now} and $t_k = t_{now} - k\delta$; n_{ji} is the number of SIs from station j to i. Both γ_{ji} and F_{ji} can be obtained probabilistically based on historical SI data.

So far, we are able to compute the SIs between two stations. Thus, by taking the summation of n_{ji} over all possible source stations, we can get the expression for $A_i(t, \Delta t)$ as

$$A_i(t, \Delta t) = \sum_{j \in N} n_{ji}. \tag{3}$$

Note that bikes checked out after t_{now} may end up at the destination station during the target period. One can simply ignore this part (i.e., treating it as 0) if $t \approx t_{now}$. However, omitting those SIs may cause a large error when $t \gg t_{now}$. To cope with this problem, we devise a check out prediction approach which is described in detail in Section 4. Once obtained the approximate numbers of check out bicycles (after t_{now}), we can feed them into the mobility model to compute the check in number (i.e., the red dashed lines in Figure 2).

3.2 Pruning

In Section 3.1, we derive a theoretical mobility model for check in estimation. Now we demonstrate how to compute the check in number of bicycles at each station in a practical way based on real check in/out data. Specifically, we adopt several pruning techniques by taking advantage of properties of BSS.

3.2.1 Temporal Pruning

In the theoretical mobility model, k ranges from 0 to ∞. Though it is infeasible to compute the sum of an infinite series, we find people usually use public bikes for the last miles of a commute for a short time. In other words, it is reasonable to set a cut off value on k to perform a temporal pruning.

Take the bike-sharing system in Hangzhou as an example. The distribution function of trip durations is depicted in Figure 4[2]. It demonstrates that 99.6% SIs are completed within 3 hours. Thus, instead of summing k from 0 to ∞, we can safely set a limit on k to 3 hours.

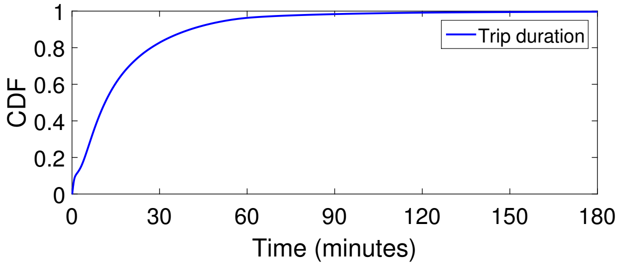


Figure 4: CDF of trip duration.

3.2.2 *Spatial Pruning*

In the theoretical model, SIs between any two stations are taken into consideration. However, we find it is not essential since the traffic between two far away stations is usually small and can be ignored. In other words, instead of taking all possible source stations into consideration, we use a subset of them to compute the potential check in bikes.

Let $M_i(n)$ denote the top-n source stations of station i in terms of the number of check in bicycles, and n_{ji} denote the number of bikes from station j to station i. We further define the top-n cumulative contribution rate $\lambda_i(n)$ as follows:

$$\lambda_i(n) = \frac{\sum_{j \in M_i(n)} n_{ji}}{\sum_{j \in N} n_{ji}}. \qquad (4)$$

The average, minimal and maximal value of $\lambda_i(n)$ over all stations is depicted in Figure 5, which demonstrates that the majority of check in bikes of a specific station can be attributed to a small group of source stations. For example, the top 200 stations contribute more than 96.6% of bikes on average. Thus, we can perform a spatial pruning by limiting the number of source stations to a small value. Specially, we rank the numbers of SIs between each pair of source station and target station[3], and select the top 200 ones (at most) in check in estimation.

3.2.3 $\gamma_{ji}(t)$ *Discretization and Calculation*

It is also worth investigating how to efficiently update the parameter $\gamma_{ji}(t)$. By examining the SI data between stations, we find that γ_{ji} is highly volatile within a day whereas it remains relatively stable across days. Also, due to the sporadic check out of bicycles, we discretize $\gamma_{ji}(t)$ into a piece-wise function, and compute its value within each time slot (e.g., one hour) based on historical bicycle check in/out data. In the proposed mobility model, we set the length of each time slot to one hour to get a tradeoff between the computational overhead and accuracy.

By utilizing the above pruning techniques, it is computationally feasible to build a bicycle mobility model and perform check in estimations. In short, our model consists of two phases, namely a training phase and an estimation phase. In the training phase, model parameters (e.g, γ_{ji} and F_{ji}) are calculated from historical data while estimations are conducted by leveraging the model and online check out records during the estimation phase. The training and estimation algorithm for one station is presented in Algorithm 1 and Algorithm 2, respectively.

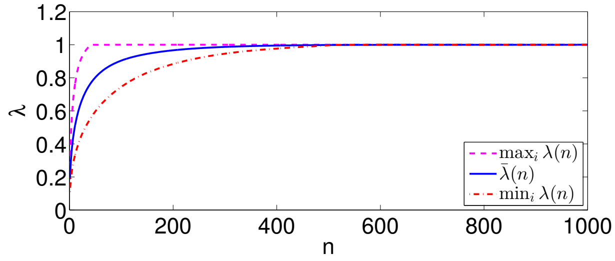


Figure 5: Cumulative contribution rate.

```
Input  : Training dataset $S$, Number of source station
         considered $m$
Output: Mobility model of station $i$
$model_i = NULL$;
for each time slot $t$ do
    $sources$ = TopSource ($m$);
    for each station $j \in sources$ do
        Compute $\gamma_{ji}$ for time slot $t$;
        Estimate trip duration CDF $F_{ji}$ from $j$ to $i$;
        Add $\gamma_{ji}$, $F_{ji}$ to $model_i$;
    end
end
return $model_i$
```

Algorithm 1: Mobility modeling training.

In the training phase, for each time slot, Algorithm 1 computes γ_{ji} and F_{ji} between top-m source stations and the target station. Thus, the computation complexity for each station is $O(nm)$ where n is the number of time slots and m is the number of considered source stations. In the estimation phase, for each time interval, Algorithm 2 estimates the traffic between each pair of source station and target station. Thus, the corresponding computation complexity is $O(pm)$ where p is the number of time intervals. Both algorithms can run in parallel, therefore being applied to all stations simultaneously in a large bike-sharing system. Due to the changes in BBS (e.g., new station deployment) and external conditions (e.g., road construction), we train the mobility model every one week by using all SI data from the past one year.

4. BICYCLE CHECK OUT PREDICTION

Section 3 presents a mobility model with check in analysis based on SIs started before t_{now}. In this part, we introduce how to predict the number of bicycle check out after t_{now}, thus completing the mobility model to obtain the total number of check in bicycles between $[t, t + \Delta]$.

Different from bicycle check in results which are correlated with stations, users' check out actions are mutually independent but subject to external factors such as time factors and meteorology. Therefore, we apply *random forest theory* [12] to model and forecast the check out behaviors based on historical SIs, corresponding time and meteorology data.

[2] The detailed dataset description is presented in Section 5.

[3] A time slot is set to 1 hour in our paper.

```
Input : Mobility model $model_i$, Estimation time period
        $[t, t+\Delta]$, $k_{max}$, $\delta$
Output: Estimation result $n$
$n = 0, k = 1$;
$p = \frac{\Delta}{\delta}$;
while $k < k_{max}$ do
    $sources$ = GetSource ($model_i$,$t + \Delta - k\delta$);
    for each station $j$ in sources do
        Compute $D_j(t + \Delta - k\delta, \delta)$;
        Get $\gamma_{ji}$ from $model_i$;
        Get $F_{ji}(k\delta)$ from $model_i$;
        if $k \geq p$ then
            // bikes depart before $t$
            Get $F_{ji}(k\delta - \Delta)$ from $model_i$;
            $n =$
            $n + D_j(t + \Delta - k\delta, \delta)\gamma_{ji}(F_{ji}(k\delta) - F_{ji}(k\delta - \Delta)$;
        else
            // bikes depart after $t$
            $n = n + D_j(t + \Delta - k\delta, \delta)\gamma_{ji}F_{ji}(k\delta)$;
        end
    end
    $k = k + 1$
end
return $n$
```

Algorithm 2: Check in estimation.

4.1 Feature Extraction

We first extract features from raw data and create feature vectors with fixed-size time window to build the random forest model. The features that affect the check out behaviors in a significant way can be categorized into two types: offline features and online features.

4.1.1 Offline Features

Time factors: Although characteristics of check out actions differ among stations, they are all closely related to time factors and show unique temporal patterns. We select the most significant four time factors: day of week, time of day, weekday and holiday.

Figure 6 shows the average check out number of a station in a tourist attraction area in different days of one week. We notice that the average check out number on weekdays is relatively smaller than that during weekends. Things are the opposite for the stations in residential areas or commercial areas. Therefore, day of week is informative for check out prediction.

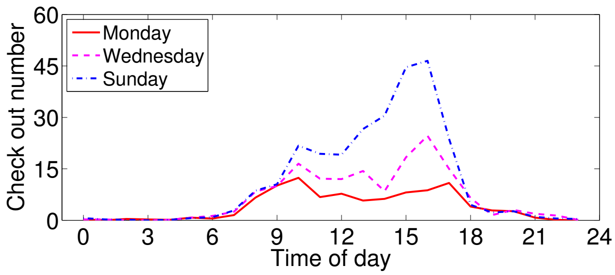


Figure 6: Check out number at one station in different days.

Figure 7 shows the check out number over time on a weekday which varies dramatically. There are two peaks at around 08:00 and 16:00. Since a significant portion of users rent bicycles for fixed purposes (e.g., commute), time of day is also considered as a feature in our model.

There are two additional features that are associated with date: weekday and holiday. These are binary indicators representing whether a particular day is a weekday or holiday.

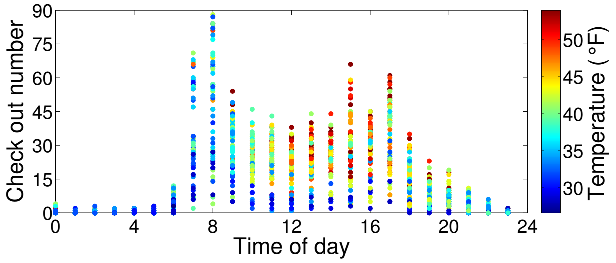


Figure 7: Check out number by time and temperature.

Meteorology: Meteorology condition has a huge influence on user behaviors in BSS [7, 10, 13]. As is shown in Figure 7, check out numbers grow when people feel more comfortable at higher temperatures. Similar patterns exist for other meteorological conditions such as humidity, visibility and wind speed. Nevertheless, these patterns vary with time and across stations. For example, users' check out behaviors are less influenced by weather conditions during peak hours.

4.1.2 Online Feature

Online check out number: Despite the tight correlations between users' check out behaviors and time factors as well as the meteorological data, there exist check out anomalies that differ significantly from the results observed at the same hours on other days. We notice that these anomalies are caused by sudden changes of bike availabilities at stations. For instance, bicycles in surrounding areas run out very quickly with large audiences coming from a stadium after a soccer game. Also, the bike-sharing service becomes unavailable when dock (or even station) failures happen. Since anomalies usually last for a short period of time, we adopt an online feature of check out number from the previous time window. By incorporating this feature, the model is capable of adapting to accidental events that dramatically change the bike availability at stations.

We combine all offline and online features mentioned above to generate a feature vector f_t for each time window t. Denote the ground truth of check out number for each time window as r_t, we combine feature f_t and r_t into a big vector $x_t = (f_t, r_t)$ to train the model.

4.2 Random Forest Model

Random forest is an ensemble learning method typically used for classification and regression. The general idea of random forest is to combine a large number of decision trees, each of which is individually built on bootstrapped samples of the data. The predictions are performed by taking the mean of outputs from each individual decision tree.

During the training, we first conduct the training set S_t by concatenating all the historical x_t mentioned above. Then we sample the training set randomly with replacement to create a subset. For each subset, a tree grows according to the following steps: at each node, it chooses some features randomly as split variables from all the predictor variables in the feature vector; it then finds the best split according to the criterion that maximizes the homogeneity of the two resulting groups. In our design, we use a mean square error function to measure the quality of a split; at the next node, the procedure is repeated and the training set is partitioned into smaller groups.

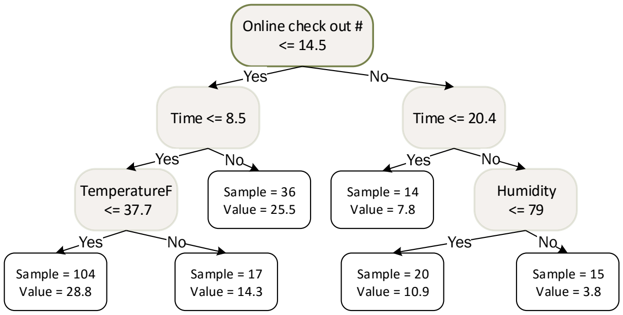


Figure 8: A decision tree in random forest.

Figure 8 shows an example decision tree with four layers. To predict the check out number of time $t+1$, the input feature vector f_{t+1} must contain the following fields: *day of week*, *hour of day*, *holiday*, *weekday*, the *meteorology forecast value* at time $t+1$ and *online check out number*. When f_{t+1} is entered into the tree, it needs to make a decision at each node based on the split variable and makes its way down till it reaches a leaf node. For example, in this figure, if the *online check out number* is smaller than or equal to 14.5, the next step is to decide whether *time* is larger than 8.5. If not, it will reach a leaf node where sample = 36 and value = 25.5. It means that there are 36 samples in the historical SIs that also meet the above conditions and the value 25.5 is the average check out number of those 36 samples. Let y_n be the prediction of the n_{th} single decision tree. Then $p = \frac{1}{N_{tree}} \sum_{n=1}^{N_{tree}} y_n$ is the final check out prediction.

For online check out prediction, the time window is set to 30 minutes. At time window t, we predict check out number for K steps to feed the mobility model, i.e., computing $\{p_{step}\}_{step=1}^{K}$. Under this situation, the input sequence depends on the ground truth of check out numbers from time window t to time window $t+K-1$, which is unknown by now. Hence we consider the check out prediction at time t as the ground truth and use it as an input feature for prediction in the next step. The detailed algorithm is shown in Algorithm 3. When it comes to the next time window, the preceding prediction sequence is discarded and we update prediction with new observations. Similar to the mobility model, we update the random forest model every one week.

```
Input  : Training dataset $S_t$, prediction horizon length K
Output: Check out prediction $\{p_{step}\}_{step=1}^{K}$
Initialize step = 1;
for each station $j$ do
    Use $S_t$ to train the random forest model;
    Compute feature importance for station $j$;
    while step < $K$ do
        predict the check out number $p_{step}$ given observation $f_{step}$ ;
        combine the perdition $p_{step}$ and other offline features to $f_{step+1}$;
        step = step + 1;
    end
end
```

Algorithm 3: Check out prediction for K steps.

Compared with other algorithms, the proposed random forest-based model has the following advantages:

- It can deal with both categorical and numerical variables without normalization. In our original data, some meteorology features such as temperature and wind speed are numerical variables while holiday and weekday are binary variables. We can just take these features as input without additional conversion.
- It provides importance of features based on the training dataset, which sheds light on the check out patterns. For example, meteorology features may have higher importance on stations near tourist attractions.
- It can deal with huge volumes of data and can be easily parallelized, and thus is very suitable for check out behavior modeling based on millions of SIs.

Denote the length of the training set S_t as N, the length of the test dataset as T. The complexity of feature extraction is $O(N+KT)$. With regard to the random forest building with N_{tree} number of trees, the complexity is $O(N_{tree} * Nlog(N))$. Hence, the total computation complexity is $O(Nlog(N))$.

4.3 Putting It All Together

In Section 3 and Section 4, we are able to predict SIs from two aspects: mobility modeling with check in estimation, and check out prediction. The combination of two parts presents a complete picture of the bike-sharing system. Specifically, for check out prediction in Section 4, we utilize a random forest model to capture the relationship between check out behaviors and key factors, including time/date, meteorology, and recent check out status. Then, given these features, we are able to predict check out instances in a future period.

In Section 3, we leverage the spatio-temporal relationships between pairs of stations and derive a probabilistic model to describe the bike mobility. By feeding both check out records in the past (i.e., the black lines in Figure 2) and the predicted results in the future (i.e., the red dashed lines in Figure 2) into the mobility model, we are able to estimate the check in numbers at each station at any given period in the future. In addition, the number of docked bicycles at each station can be inferred by integrating the check in and check out prediction results, which is of great value to both customers and system operators.

5. DATASET DESCRIPTION

Before presenting performance evaluations, in this section, we first describe the two datasets used in this paper: the *BSS Dataset* and the *Meteorology Dataset*. The BSS dataset is provided by Hangzhou Public Bicycle Transport Service Development Co., Ltd., and the meteorology dataset was collected from an online weather service provider Weather Underground[4].

5.1 BSS Dataset

The Chinese city of Hangzhou has the world's largest public BSS with more than 3300 stations and over 84,000 shared bicycles [8, 9]. The system adopts smart-card technology and automated check in and check out. Each check in and check out instance will be recorded in the back-end database with corresponding user ID, bicycle ID, time and etc.

The BSS dataset we use includes all bicycle sharing records in 2013. In summary, it contains 43705 bikes and 2806 stations,

[4] http://www.wunderground.com

Table 2: Primary fields in the BSS dataset.

user_id	rent_netid	tran_date	tran_time
6114381	4051	20130101	000152
return_netid	**return_date**	**return_time**	**bike_id**
4015	20130101	001547	913672

among which 103,661,080 trips are recorded. Each bike-sharing record has 47 fields, of which the primary fields are shown in Table 2. In 2013, the operational hour for check out service ranges from 6:00am to 9:00pm while the check in service is available until 11:00pm. In order to provide a better overview, we report statistical results of the dataset in terms of station distribution, station capacity and usage amount.

Station distribution: Bike stations in Hangzhou are located within the urban area spanning over 600 square kilometers; the average distance to the closest neighboring station being 300 meters [8]. The probability distribution function (PDF) of the number of stations within a certain range of one station is depicted in Figure 9. As we can see, half the stations have more than 3 neighbors within the range of 300 meters, and they typically have 20 neighbors within the range of 800 meters.

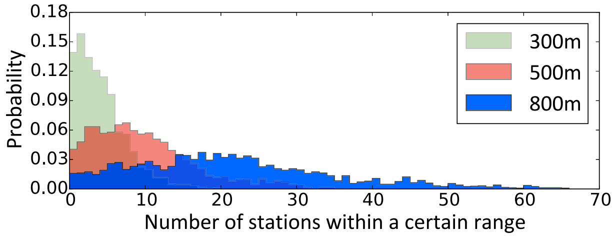


Figure 9: PDF of the number of stations within a certain range of one station.

Station capacity: Station capacity is measured as the number of stocks. As is shown in Figure 10, there are basically two types of stations in Hangzhou: normal stations with 21 docks and large station with around 33 docks.

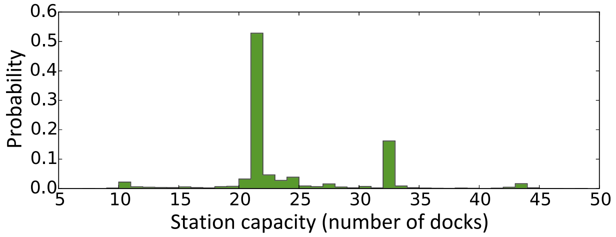


Figure 10: Distribution of station capacity.

Usage amount: Figure 11 presents the distribution of monthly usage amount (i.e., check in numbers) across all 2806 stations. We find that there are more than 100 busy stations with extremely high usage amount up to 30000 (check in/month). However, the median value of the usage amount is around 2000. The skewed distribution in Figure 11 indicates a high diversity of usage amount across different stations. We observe a similar pattern for check out numbers.

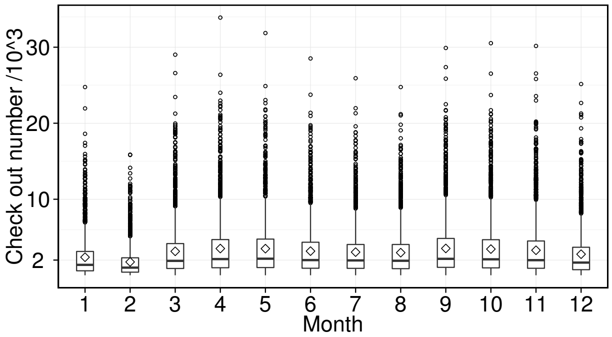


Figure 11: Distribution of monthly usage amount.

Table 3: Fields in the meteorology dataset.

Time (CST)	Temp (°F)	Dew Point (°F)
12:30 PM	100.4	69.8
Pressure (hPa)	**Humidity (%)**	**Visibility (MPH)**
29.65	37	6.2
Wind Dir	**Wind Speed (MPH)**	**Conditions**
WSW	8.9	Partly Cloudy

5.2 Meteorology Dataset

The meteorology dataset contains weather conditions of Hangzhou with totally $48 \times 365 = 17,520$ records. Meteorological observations were updated every half hour and the data format of each record is shown in Table 3.

6. EVALUATION

We conduct extensive data-driven simulations to evaluate the performance of our design. In the following, we first present the performance of check out prediction, and then show the check in estimation results based on the mobility model and check out prediction. Both case studies and overall performance are shown to provide a more comprehensive analysis.

6.1 Baseline Approaches

We first introduce three prediction techniques that comprise the baselines of our model.

- **Historical Average (HA)** uses the average of historical observations for the same time and location to forecast the future data [11]. Specially, when conducting prediction for a specific time period, we first find out historical days with same day/time values, and then average the check in/out results from these periods.

- **Auto-Regressive and Moving Average (ARMA)** is widely used for time series prediction and was adopted for BBS check in estimation [5]. It leverages check in/out information of the most recent p time windows for future prediction. Parameters are determined using historical data and the least squares method.

- **HP-MSI and P-TD** are the most recent studies of traffic prediction in bike-sharing system [7] where HP-MSI and P-TD are designed for check out and check in prediction, respectively. Since these two methods are applied to predict traffic amount between clusters of stations, they cannot be directly compared with our station-basis design. As an approximation, we treat each station as an individual cluster in our evaluation.

For performance metrics, we adopt CDFs of both absolute and relative prediction error. In addition, we also use Root Mean Squared Logarithmic Error (RMLSE) [7] for evaluation. RMLSE is computed as

$$RMSLE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(log(\hat{y}+1) - log(y+1))^2} \qquad (5)$$

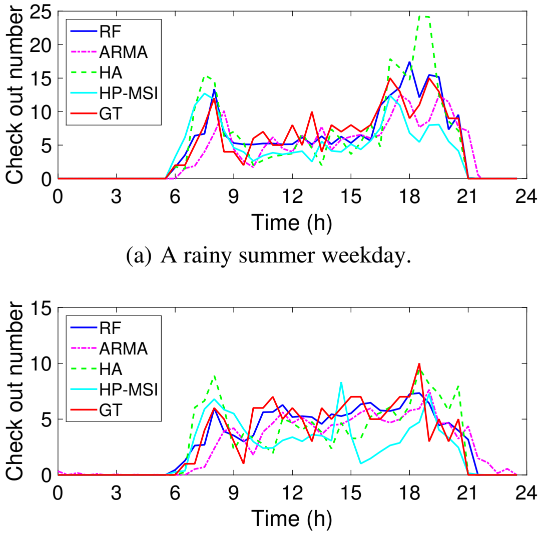
where $\hat{y}$ and y are prediction and ground truth respectively while n is number of predictions.

6.2 Check Out Prediction

In this part, we evaluate the check out prediction model presented in Section 4, which serves as a component for check in estimation.

6.2.1 *Case Studies*

We first present a case study of station 3648 under different scenarios, and predict its check out number for 24 hours adopting HA, ARMA, HP-MSI and the proposed random-forest-based method (RF). The *ground truth* (GT) is also provided in Figure 12. Figure 12(a) presents the check out prediction results every 30 minutes in a rainy summer weekday while Figure 12(b) is generated in a sunny winter weekend.



(a) A rainy summer weekday.

(b) A sunny winter weekend.

Figure 12: Check out prediction at station 3648.

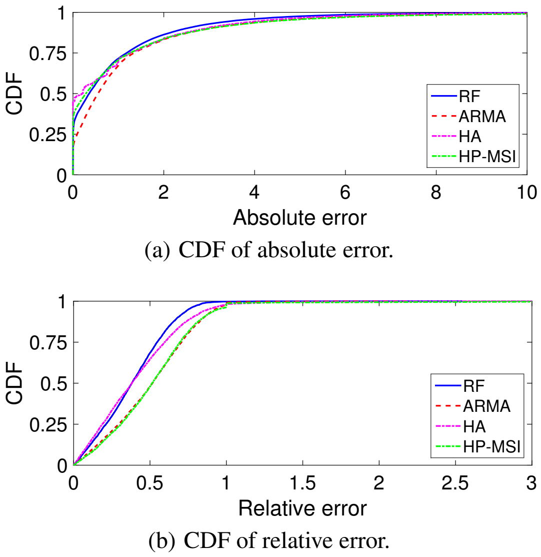
From Figure 12 we find that check out number of station 3648 in summer are much larger than those in winter, especially during rush hours. It means that the feature "hour" is an influential factor to users' check out behaviors. We further record the importance of the features while growing trees in the random forest model in Table 4. For each internal node that splits on feature i, feature importance is computed by the impurity decrease from that feature. For regression trees, the impurity criterion that we use is variance. For station 3648, the importance of "hour" is as high as 0.1434. In addition, we find that the "online check out number" is the most influential factor in this case. It represents the randomness of check out patterns at this station. Another evidence of that is the bad performance of HA which depends highly on previous observations. For example, the prediction error of HA at 19:00 in Figure 12(a) is as much as 24.14, 60.93% more than the ground truth. On the other hand, ARMA exhibits relatively large prediction delays in both figures. This is due to the sliding window technique used in the model. Among all three approaches, RF demonstrates the best performance both in terms of accuracy and delay.

Table 4: Feature importance.

Day of week	Hour	Temperature	Humidity
0.0288	0.1434	0.0846	0.0514
Visibility	Wind speed	Holiday	Workday
0.0332	0.0211	0.0030	0.0064
Online check out number			
0.6282			

6.2.2 *Overall Performance*

Figure 13 presents the overall prediction performance across all stations. We adopt the first 20 days of each month to train the random forest model, and predict check out numbers in remaining days. Compare the prediction results with ground truth, we can obtain the CDF of the errors. Figure 13(a) demonstrates the absolute error (i.e., numbers of bikes) while Figure 13(b) shows the relative error calculated by dividing the absolute error by the ground truth.



(a) CDF of absolute error.

(b) CDF of relative error.

Figure 13: Overall check out prediction performance.

From Figure 13 we can see that 90% of the absolute error is less than 2.477 in our approach, outperforming three baselines. Table 5 also gives evidence of the advantages of our approach while the RMSLE of RF is as low as 0.4287. Similar patterns exist for relative error. For example, RF achieves an 85 percentile relative error of 0.62659 while the corresponding value for ARMA is 0.7039. It is even worse for HA and HP-MSI[5]. Both Table 5 and

[5] Note that the relative error is the ratio of absolute error to the

Table 5: RMSLE of check out prediction.

RF	ARMA	HA	HP-MSI
0.4287	0.4855	0.4600	0.4662

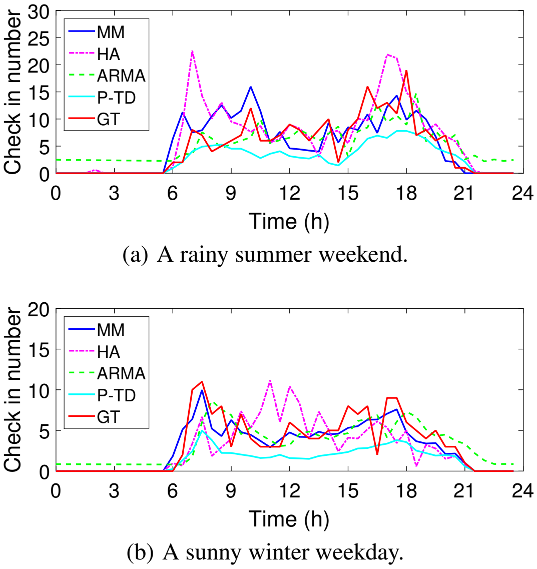
Figure 13 prove the effectiveness of the proposed random-forest-based model which takes both time factors, meteorology and real-time check out behaviors into consideration.

6.3 Check In Estimation

In this section, we evaluate the effectiveness of the proposed mobility model (MM) by demonstrating the results of check in estimations.

6.3.1 Case Studies

Similar to the previous section, we first present two case studies of a rainy summer weekend and a sunny winter weekday to provide intuitive feelings of performances of different approaches.



(a) A rainy summer weekend.

(b) A sunny winter weekday.

Figure 14: Check in prediction at station 3648.

As can be seen from Figure 14, MM is close to the ground truth. However, HA brings larger estimation error because it lacks online information. Similar to Figure 12, AR suffers from an estimation delay. As for P-TD, it tends to underestimate the result.

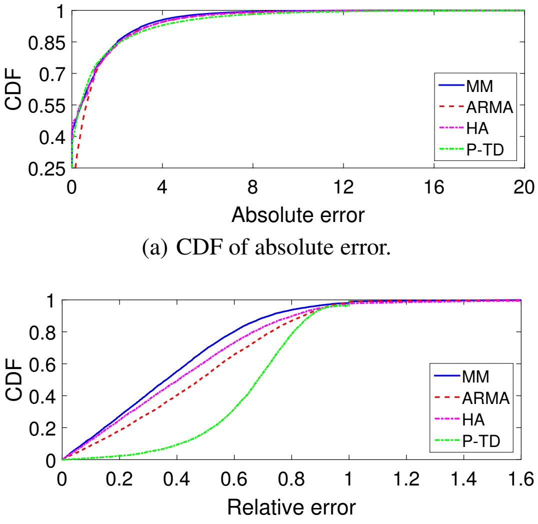
6.3.2 Overall Performance

We also evaluate the overall performance of check in estimation of all three approaches. Specifically, each approach is required to estimate check in number in the following 30 minutes (i.e., $\Delta = 30$). Similar to the random-forest-based model, we train each mobility model using SIs from the first 20 days of each month, and use the remaining days for testing. The CDF of absolute error and relative error is depicted in Figure 15, while RMSLE results are presented in Table 6.

As one can see from Figure 15(a), for absolute error, all approaches are quite close when absolute error is small. However, as the error increases, MM outperforms other methods. This trend becomes more obvious in Figure 15(b). Recall that we only consider time slots with actual check in number larger than 5, which implies that MM performs well when the traffic increases. This is because when the total traffic rises, the expectation number calculated by MM suffers less from randomness and becomes more accurate.

Table 6: RMSLE of check in prediction.

MM	ARMA	HA	P-TD
0.4736	0.5296	0.4865	0.5042



(a) CDF of absolute error.

(b) CDF of relative error.

Figure 15: Overall Performance of check in Estimation.

Note that in Figure 15(b), we observe significant performance degradation of PT-D. This is caused by the approximation of single-station-clustering where relative errors will be amplified with smaller ground truth values. Also, PT-D has to fit the trip duration distribution function in log-normal form, and the lack of check in/check out records between certain stations results in larger fitting errors.

6.3.3 Impact of Settings

To better understand the performance of the proposed mobility model and the check in estimation approach under different settings, we further conduct three sets of evaluation by varying several key parameters in the mobility model. We present the results of both absolute error CDFs and RMSLE.

- **Check out prediction:** As mentioned in the Section 3, part of the check in bikes between $[t, t+\Delta]$ may depart later than t_{now}, which is unknown at the time of check in estimation. Thus, we utilize the check out prediction result. Figure 16 demonstrates the effectiveness of this approach. Specially, we compare three different strategies to estimate bikes depart after t_{now}: i) Check out prediction, which is the method used in this paper; ii) Ground truth, which uses the real number of bikes depart after t_{now}. It is infeasible to obtain this figure in practice hence the results are viewed as an upper bound; iii) Ignore this part, which simply treats the number as zero.

  From Figure 16 and Table 7, we can see that the CDF

ground truth, rather than the ratio of prediction result to the ground truth, therefore Figure 13(b) does not imply that the algorithm underestimates the check out number.

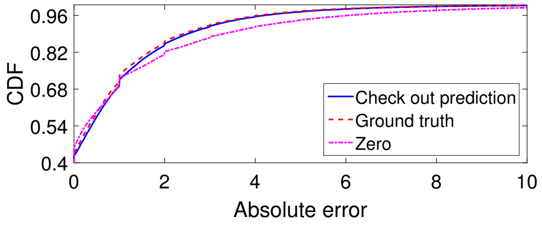


Figure 16: Impact of check out prediction.

Table 7: RMSLE of different strategies.

Ground truth	Check out prediction	Zero
0.4556	0.4736	0.5767

curves of the check out strategy and the Ground truth strategy are very close. In other words, the gap between our result and the theoretical upper bounding is quite small. In addition, we notice that ignoring this part degrades the system performance significantly, which justifies the necessity of adopting check out prediction result.

- **Granularity of time interval** (δ)**:** In order to build a theoretical mobility model for BBS, we discretize time into intervals. Intuitively, shorter intervals will bring better estimation accuracy, but add to the cost of computational complexity. Therefore, we adopt three different time intervals 2, 5 and 10 minutes to compare their modeling and prediction results.

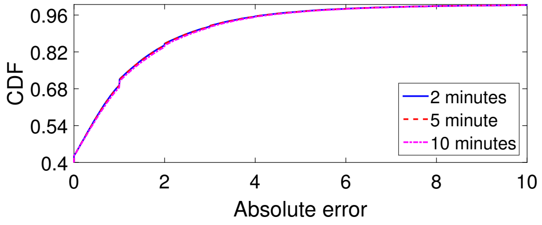


Figure 17: Impact of the granularity of time interval.

From Figure 17 and Table 8 we find that a smaller time interval results in a slightly better performance. However, gaps between each curve are all very small. Hence we use a larger time interval to reduce the computational overhead in current implementation.

- **Number of source stations:** In the mobility model, number of source stations is limited to a threshold to tradeoff between the accuracy and computational overhead. However, we are also curious to study the impacts of number of source stations on the estimation result. Figure 18 shows the CDF of absolute error of three different thresholds, 100, 200 and 400.

  In Figure 18, it can be seen that algorithms perform better when more source stations are taken into consideration. However, one may also notice that the gap between 400 stations and 200 stations is smaller than that between 200 and 100. It is also verified in Table 9 where the RMSLE values of 200 stations and 400 stations are identical. Thus, using a proper number of source stations (i.e., 200) achieves a good balance between estimation performance and computational overhead.

Table 8: RMSLE of different time intervals.

2 minutes	5 minutes	10 minutes
0.4736	0.4750	0.4844

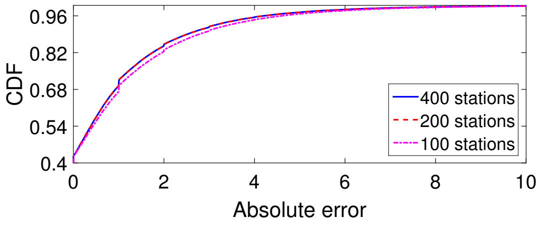


Figure 18: Impact of the number of source stations.

6.4 Evaluation on Other Datasets

To further demonstrate the effectiveness of proposed algorithms, we also perform small-scale evaluation on a bike-sharing dataset from New York City, U.S. [14]. We adopt same settings used in previous evaluation, and summarize the prediction error of check out and check in in Figure 19 and Figure 20, respectively.

As can be seen from both figures, all approaches demonstrate similar performance compared with the dataset from Hangzhou. The proposed algorithms still own the best prediction results, followed by HA, AMRA and P-TD. Taking check in prediction as an example, 80th percentile relative errors are 0.568, 0.5909, 0.6664, 0.8973 for MM, HA, AMRA and P-TD from the New York City dataset, while they are 0.598, 0.6667, 0.7276, 0.8061 from the Hangzhou dataset.

7. DISCUSSION AND FUTURE WORK

We provide several insights into the modeling and prediction results, and provide directions for future work in this part.

7.1 Insights

We first give some insights from both different scenarios and modeling approaches.

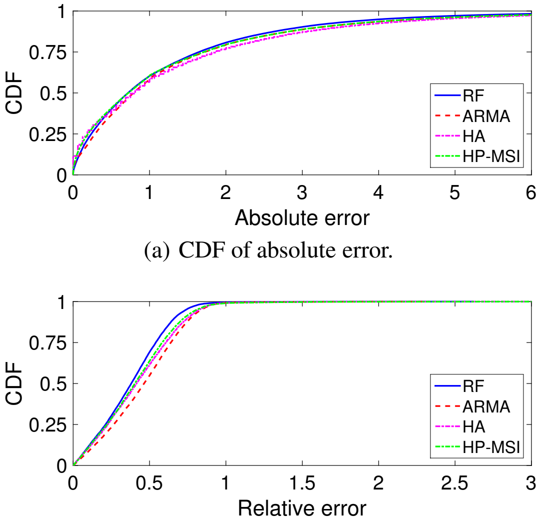
7.1.1 Variation Among Different Scenarios

We conduct studies in different scenarios to get a better understand of human mobility in BSS. CDFs of relative error of check in prediction are presented in Figure 21. The results of check out prediction are similar and omitted due to space limitation.

In Figure 21(a), we compare prediction results between stations in business area and tourist area. As we can see, stations in business area are more predictable due to users' regular mobility patterns (e.g., from home to office). Differences between rainy days and sunny days are shown in Figure 21(b). Consistent with our intuition, fewer people use public bikes in rainy days, which

Table 9: RMSLE of different numbers of source stations.

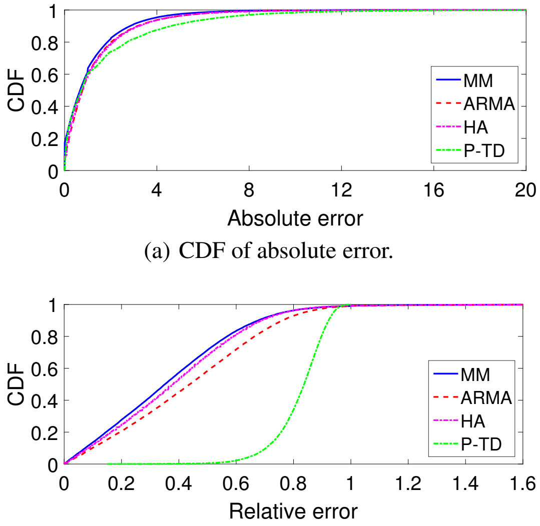
400 stations	200 stations	100 stations
0.4736	0.4736	0.5021



(a) CDF of absolute error.

(b) CDF of relative error.

Figure 19: Overall check out prediction performance in NYC.



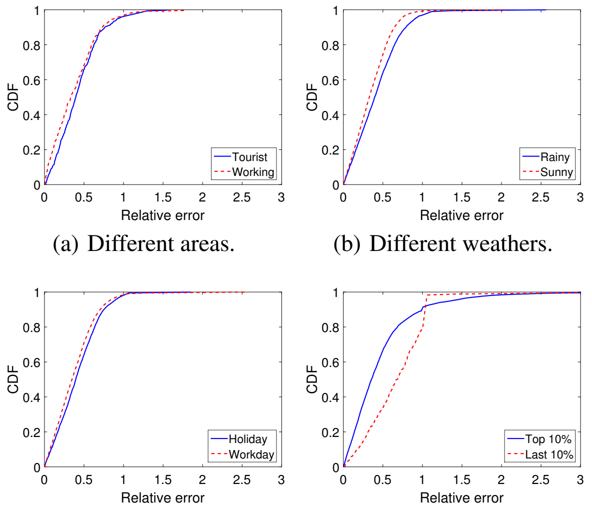
(a) CDF of absolute error.

(b) CDF of relative error.

Figure 20: Overall check in prediction performance in NYC.

increases randomness and degrades prediction performance. As can be seen from Figure 21(c), workdays own better prediction results than holidays or weekends for similar reasons. Finally, stations with high utilization exhibit high predictability over stations with low throughputs in Figure 21(d).

In addition to the variations among days and stations, it is also interesting to see the impacts of special events. As can be seen from Figure 22, absolute error of check out prediction for one station near a stadium is 16.735 at 17:00, when an opening ceremony is over, which is around 4 times higher than that from the past half hour. When such special events happen, there tends to be a lot more people renting bikes than the historical average, inevitably causing bike usage anomalies and poor performance of prediction. Since these events happen less frequently and it is difficult to develop a static model to capture their impacts on BSS, we choose to leverage online features to compensate the impacts of special events in Section 4.



(a) Different areas.

(b) Different weathers.

(c) Different days.

(d) Different stations.

Figure 21: Overall check in prediction performance (CDF of relative error).

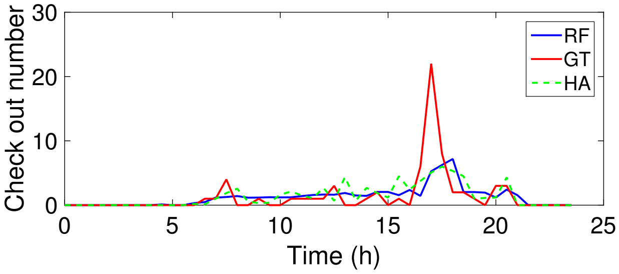


Figure 22: Check out prediction when the Games' opening ceremony are being held.

7.1.2 User-centric Modeling and Prediction

In this paper, we model and predict users' mobility in a station basis. However, one may also consider to do that in a user-centric way. We conduct some preliminary research in this direction and present the results here. Specifically, we aim to identify regular users who have fixed routes (e.g., from home to school), and exploit their profiles for modeling and prediction.

A user is categorized as a regular user if he or she has generated n routes of which the check out stations as well as check in stations fall into two small circles with 1km radius. In this study, we vary n and compute the percentage of regular users based on the data from January, 2013.

From Table 10, we can see that as n increases, the percentage of regular user decreases. However, even with a smaller n of 6, only 12.55% users can be classified as regular users. Therefore, the user-centric mobility modeling and prediction, while hopeful, left challenging problems due to the low percentages of regular users.

Table 10: Regular user percentage.

n	regular user(%)	n	regular user(%)
6	12.55%	14	3.85%
8	9.19%	16	2.87%
10	6.65%	18	2.02%
12	5.02%	20	1.23%

7.2 Open Research Issues

From a mobile system point of view, we summarize some open research issues related to the emerging bike-sharing systems.

7.2.1 Mobility Model Fusion with Multi-source Data

Study of human mobility has drawn significant attention in the mobile community. One intuitive idea is to improve the existing models by integrating bike-sharing data. In [15], authors have demonstrated the reduced bias of mobility modeling by exploiting the inherent diversities from multi-source data (i.e., taxi, bus, subway and smartphone CDR).

7.2.2 Bike Rebalancing

Rebalancing, a reality for pretty much every bike-sharing system, can benefit from the accurate mobility modeling and prediction. However, how to design an efficient and practical rebalancing algorithm is non-trivial. For example, one needs to calculate the number of shuffled bikes at each station and plan a route at the same time. Apart from operator's "passive" rebalancing, how to devise an incentive and price mechanism enabling user-based "proactive" rebalancing is also an interesting subject to pursue.

7.2.3 Service Optimizations

In addition to bike rebalancing, future work on service optimization includes station location optimization, service hour optimization, pricing strategy design, bicycle utilization balancing etc. From a customer perspective, prompt bike stock information delivery and user-friendly interaction design is also of great help. To achieve this goal, we have developed a demo application [16], which not only provides early stock warnings for operators, but also serves as a tour guide for bike users.

8. RELATED WORK

Extensive research has been done to describe the nature of bike-sharing systems, business models, how they have spread in time and space and why they have been adopted [8, 17–25]. For example, Shaheen *et al.* reviews the history, advantages and inadequacies of bike-sharing systems across the globe [17, 18, 24]. Martin *et al.* evaluates transit modal shift dynamics with the emergence of public bike-sharing [22]. Parkes *et al.* [19] uses diffusion theory to compare the adoption process of bike-sharing in Europe and North America. Comprehensive analysis and survey of city-scale bike-sharing systems in Paris [20], Hangzhou [8], New York [25], Washington D.C. [21] and Montreal [23] have also been conducted.

Adoption of bike-sharing systems have motivated studies on system design optimization. A first line of work focuses on the sensing of dynamics from bike-sharing system data [4–7, 10, 13, 20, 26–28], which broadly consider on two topics, namely clustering and prediction. Most clustering approaches identify mobility patterns in bike usage and partition the stations into clusters based on their usage profiles [3, 7, 10, 26, 27]. For instance, in [3], two clustering techniques using activity statistics derived either from the evolution of station occupancy or the number of available bicycles along the day. In [26], authors use graphs to describe the similarity of usage profiles between pairs of stations for weekdays and weekends, which is then analyzed using a community detection algorithm for clustering. In contrast to clustering, the aim of prediction is to forecast the occupancy of the stations or the network state over time by means of time series analysis [4–6, 28], Bayesian networks [3] and supervised regression model [13]. For instance, Borgnat *et al.* [6] forecasts the global rental volume, whereas Li *et al.* [7] infers the bike rental/return demand of a cluster of stations based on historical check in and check out data. To our knowledge, we are the first to generate a global spatio-temporal mobility model considering flows between stations, and to provide fine-grained prediction results on a per-station basis. Such a spatio-temporal mobility model is the key to any improvements of the bike redistribution strategy.

Based on insights into usage patterns and bike trip demand analysis, research has also been conducted to optimize the placement of stations in bike-sharing systems [10, 13, 29–31], and design strategies for bicycle re-balancing [32–35]. For example, Chen *et al.* [13] and García-Palomares *et al.* [29] solve the station placement problem by estimating the potential trip demand using a semi-supervised learning algorithm and a GIS-based method, respectively. Authors in [35] use a clustering-based heuristic for truck routing whereas in [32], Raviv *et al.* find truck routes by minimizing an objective function tied to both the operating cost of the vehicles as well as penalty functions relating to station imbalance. The mobility model and prediction mechanism derived in our work can be easily applied to other bike-sharing systems and lay a solid foundation for the upper layer design and optimization.

In addition to bike-sharing data, researchers analyzed human mobility based on other empirical data from taxicabs [36], buses [37], subways [38], private cars [39], WiFi APs [40, 41], cellular carriers [15, 42] and social networks [43]. Due to the unique intrinsic properties such as the decentralized structure, on-demand usage and unattended vehicles in BSS, our work provides a fundamentally different model from these designs.

9. CONCLUSION

This paper focuses on the mobility modeling and prediction in bike-sharing systems. Based on historical bicycle sharing data, we first use statistical methods to model the spatio-temporal shifts of bikes between stations, and then estimate bike check in results based on the model and online check out records. A random-forest-based prediction mechanism is further proposed to model and forecast the users' check out behaviors. The mobility modeling and prediction algorithms provide insights into the operations of bike-sharing systems, and have proved useful for the prediction of bike availability at each station, laying a foundation for further efficient re-balancing design in bike-sharing systems.

Acknowledgment

This work is supported in part by the National Basic Research Program (973 Program) under Grant 2015CB352500, and National Program for Special Support of Top Notch Young Professionals. We thank our partner BSS operator, Hangzhou Public Bicycle Transport Service Development Co., Ltd., and the anonymous reviewers and our shepherd Qin Lv for their constructive comments and support.

References

[1] LLC MetroBike. The Bike Sharing World - 2014 -Year End Data. http://bike-sharing.blogspot.com/2015/01/the-bike-sharing-world-2014-year-end.html.

[2] Wikipedia. List of Bicycle-sharing Systems. https://en.wikipedia.org/wiki/List_of_bicycle-sharing_systems.

[3] Jon Froehlich, Joachim Neumann, and Nuria Oliver. Sensing and Predicting the Pulse of the City through Shared Bicycling. In *IJCAI*, 2009.

[4] Andreas Kaltenbrunner, Rodrigo Meza, Jens Grivolla, Joan Codina, and Rafael Banchs. Urban Cycles and Mobility Patterns: Exploring and Predicting Trends in a Bicycle-based Public Transport System. *Pervasive and Mobile Computing*, 6(4):455–466, 2010.

[5] Patrick Vogel and Dirk C. Mattfeld. Strategic and Operational Planning of Bike-Sharing Systems by Data Mining - A Case Study. In *Computational Logistics*, pages 127–141. 2011.

[6] Pierre Borgnat, Eric Fleury, Céline Robardet, and Antoine Scherrer. Spatial Analysis of Dynamic Movements of Vélo'v, Lyon's Shared Bicycle Program. In *European Conference on Complex Systems (ECCS)*, 2009.

[7] Yexin Li, Yu Zheng, Huichu Zhang, and Lei Chen. Traffic Prediction in a Bike Sharing System. In *ACM SIGSPATIAL*, 2015.

[8] Susan a. Shaheen, Hua Zhang, Elliot Martin, and Stacey Guzman. China's Hangzhou Public Bicycle. *Transportation Research Record: Journal of the Transportation Research Board*, 2247(1):33–41, 2011.

[9] Wikipedia. Hangzhou Public Bicycle. https://en.wikipedia.org/wiki/Hangzhou_Public_Bicycle.

[10] Eoin O Mahony and David B Shmoys. Data Analysis and Optimization for (Citi) Bike Sharing. In *AAAI*, 2015.

[11] Nicolas Gast, Guillaume Massonnet, Daniël Reijsbergen, and Mirco Tribastone. Probabilistic forecasts of bike-sharing systems for journey planning. In *ACM CIKM*, 2015.

[12] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

[13] Longbiao Chen, Daqing Zhang, Gang Pan, Xiaojuan Ma, Dingqi Yang, Kostadin Kushlev, Wangsheng Zhang, and Shijian Li. Bike Sharing Station Placement Leveraging Heterogeneous Urban Open Data. In *ACM Ubicomp*, 2015.

[14] Citi Bike. New York City Bike-sharing System Data. https://www.citibikenyc.com/system-data.

[15] Desheng Zhang, Jun Huang, Ye Li, Fan Zhang, Chengzhong Xu, and Tian He. Exploring Human Mobility with Multi-source Data at Extremely Large Metropolitan Scales. In *ACM MobiCom*, 2014.

[16] Lihuan Zhang, Siyuan Tang, Zidong Yang, Ji Hu, Yuanchao Shu, Peng Cheng, and Jiming Chen. Demo: Data Analysis and Visualization in Bike-Sharing Systems. http://www.sensornet.cn/bikevis/.

[17] Susan a. Shaheen, Stacey Guzman, and Hua Zhang. Bikesharing in Europe, the Americas, and Asia. *Transportation Research Record: Journal of the Transportation Research Board*, 2143:159–167, 2010.

[18] Susan a. Shaheen, Adam P. Cohen, and Elliot W. Martin. Public Bikesharing in North America: Early Operator Understanding and Emerging Trends. *Transportation Research Record: Journal of the Transportation Research Board*, 2387:83–92, 2013.

[19] Stephen D. Parkes, Greg Marsden, Susan A. Shaheen, and Adam P. Cohen. Understanding the Diffusion of Public Bikesharing Systems: Evidence from Europe and North America. *Journal of Transport Geography*, 31:94–103, 2013.

[20] Rahul Nair, Elise Miller-Hooks, Robert C. Hampshire, and Ana Bušić. Large-Scale Vehicle Sharing Systems: Analysis of Vélib'. *International Journal of Sustainable Transportation*, 7(1):85–106, 2013.

[21] LDA Consulting Washington. 2013 Capital Bikeshare Member Survey Report. Technical Report 202, 2013.

[22] Elliot W. Martin and Susan A. Shaheen. Evaluating Public Transit Modal Shift Dynamics in Response to Bikesharing: A Tale of Two U.S. Cities. *Journal of Transport Geography*, 41:315–324, 2014.

[23] Ahmadreza Faghih Imani, Naveen Eluru, Ahmed M. El-Geneidy, Michael Rabbat, and Usama Haq. How does Land-use and Urban Form Impact Bicycle Flows: Evidence from the Bicycle-sharing System (BIXI) in Montreal. *Transport Geography*, (February):1–20, 2014.

[24] Paul DeMaio. Bike-sharing: History, Impacts, Models of Provision, and Future. *Journal of Public Transportation*, 12(DeMaio 2004):41–56, 2009.

[25] Department for City Planning New York. Bike-Share. Opportunities in New York City. Technical report, 2009.

[26] P. Borgnat, C. Robardet, P. Abry, P. Flandrin, J. Rouquier, and N. Tremblay. A Dynamical Network View of Lyon's Vélo'v Shared Bicycle System. In *Dynamics On and Of Complex Networks*, volume 2, chapter A Dynamical, pages 267–284. Springer Berlin Heidelberg, 2013.

[27] C Ome and Oukhellou Latifa. Model-Based Count Series Clustering for Bike Sharing System Usage Mining : A Case Study with the Vélib System of Paris. *ACM Transactions on Intelligent Systems and Technology*, 5(3):1–21, 2014.

[28] Ji Won Yoon, Fabio Pinelli, and Francesco Calabrese. Cityride: A Predictive Bike Sharing Journey Advisor. In *IEEE ICMDM*, 2012.

[29] Juan Carlos García-Palomares, Javier Gutiénrrez, and Marta Latorre. Optimizing the Location of Stations in Bike-sharing Programs: A GIS Approach. *Applied Geography*, 35(1–2):235–246, 2012.

[30] Juan P. Romero, Angel Ibeas, Jose L. Moura, Juan Benavente, and Borja Alonso. A Simulation-optimization Approach to Design Efficient Systems of Bike-sharing. *Meeting of the EURO Working Group on Transportation*, 54:646–655, 2012.

[31] Jenn-Rong Lin and Ta-Hui Yang. Strategic Design of Public Bicycle Sharing Systems with Service Level Constraints. *Transportation Research Part E: Logistics and Transportation Review*, 47(2):284–294, 2011.

[32] Tal Raviv, Michal Tzur, and IrisA. Forma. Static Repositioning in a Bike-sharing System: Models and Solution Approaches. *EURO Journal on Transportation and Logistics*, 2(3):187–229, 2013.

[33] Jia Shu, Mabel C. Chou, Qizhang Liu, Chung-Piaw Teo, and I-Lin Wang. Models for Effective Deployment and Redistribution of Bicycles Within Public Bicycle-Sharing Systems. *Operations Research*, 61(6):1346–1359, 2013.

[34] Contardo, Claudio, Catherine Morency, and Louis-Martin Rousseau. Balancing a Dynamic Public Bike-sharing System. Technical report, 2012.

[35] Jasper Schuijbroek, Robert Hampshire, and Willem-Jan van Hoeve. Inventory Rebalancing and Vehicle Routing in Bike Sharing Systems. Technical report, 2013.

[36] Raghu Ganti, Mudhakar Srivatsa, Anand Ranganathan, and Jiawei Han. Inferring Human Mobility Patterns from Taxicab Location Traces. In *ACM UbiComp*, 2013.

[37] Sourav Bhattacharya, Santi Phithakkitnukoon, Petteri Nurmi, Arto Klami, Marco Veloso, and Carlos Bento. Gaussian Process-based Predictive Modeling for Bus Ridership. In *ACM UbiComp*, 2013.

[38] Neal Lathia and Licia Capra. How Smart is Your Smartcard?: Measuring Travel Behaviours, Perceptions, and Incentives. In *ACM UbiComp*, 2011.

[39] Fosca Giannotti, Mirco Nanni, Dino Pedreschi, Fabio Pinelli, Chiara Renso, Salvatore Rinzivillo, and Roberto Trasarti. Unveiling the Complexity of Human Mobility by Querying and Mining Massive Trajectory Data. *The VLDB Journal*, 20(5):695–719, October 2011.

[40] Jungkeun Yoon, Brian D. Noble, Mingyan Liu, and Minkyong Kim. Building Realistic Mobility Models from Coarse-grained Traces. In *ACM MobiSys*, 2006.

[41] Jennie Steshenko, Vasanta G. Chaganti, and James Kurose. Mobility in a Large-scale WiFi Network: From Syslog Events to Mobile User Sessions. In *ACM MSWiM*, 2014.

[42] Sibren Isaacman, Richard Becker, Ramón Cáceres, Margaret Martonosi, James Rowland, Alexander Varshavsky, and Walter Willinger. Human Mobility Modeling at Metropolitan Scales. In *ACM MobiSys*, 2012.

[43] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility: User movement in location-based social networks. In *ACM KDD*, 2011.