Planning Bike Lanes based on Sharing-Bikes' Trajectories

Jie Bao
Microsoft Research, China
jiebao@microsoft.com

Tianfu He
Harbin Institution of Technology
Tianfu.D.He@outlook.com

Sijie Ruan
Xidian University, China
v-siruan@microsoft.com

Yanhua Li
Worcester Polytechnic Institute, USA
yli15@wpi.edu

Yu Zheng
Microsoft Research, China
yuzheng@microsoft.com

ABSTRACT

Cycling as a green transportation mode has been promoted by many governments all over the world. As a result, constructing effective bike lanes has become a crucial task for governments promoting the cycling life style, as well-planned bike paths can reduce traffic congestion and decrease safety risks for both cyclists and motor vehicle drivers. Unfortunately, existing trajectory mining approaches for bike lane planning do not consider key realistic government constraints: 1) budget limitations, 2) construction convenience, and 3) bike lane utilization.

In this paper, we propose a data-driven approach to develop bike lane construction plans based on large-scale real world bike trajectory data. We enforce these constraints to formulate our problem and introduce a flexible objective function to tune the benefit between coverage of the number of users and the length of their trajectories. We prove the NP-hardness of the problem and propose greedy-based heuristics to address it. Finally, we deploy our system on Microsoft Azure, providing extensive experiments and case studies to demonstrate the effectiveness of our approach.

CCS CONCEPTS

•Information systems →Spatial-temporal systems; Data mining;

KEYWORDS

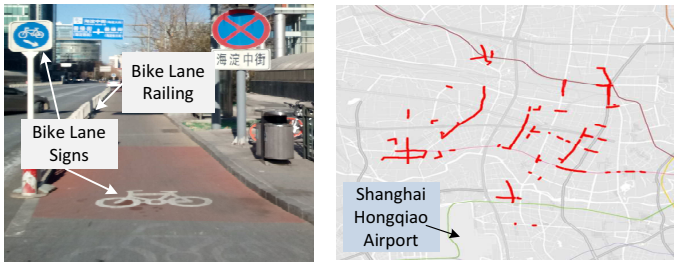
Trajectory Data Mining, Urban Planning, Urban Computing

ACM Reference format:
Jie Bao, Tianfu He, Sijie Ruan , Yanhua Li, and Yu Zheng. 2017. Planning Bike Lanes based on Sharing-Bikes' Trajectories. In *Proceedings of KDD'17, August 13–17, 2017, Halifax, NS, Canada., ,* 10 pages.
DOI: http://dx.doi.org/10.1145/3097983.3098056

This work was done when the second and third authors were interning in Microsoft Research, Beijing.

KDD'17, August 13–17, 2017, Halifax, NS, Canada.

DOI: http://dx.doi.org/10.1145/3097983.3098056



(a) An Example of Bike Lane (b) An Example of Inconvenient Plan

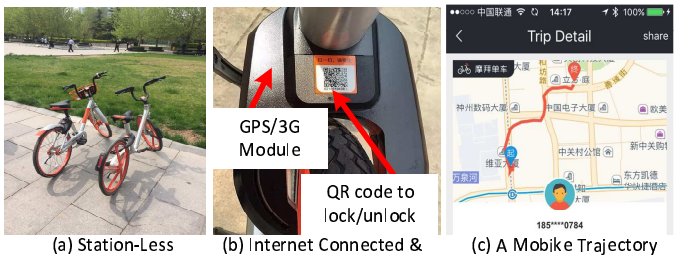
Figure 1: Motivating Examples.

1 INTRODUCTION

Cycling as a commonly used urban transit mode for daily commute has been promoted by multiple governments all over the world [1, 40] for several reasons: 1) it is an affordable and environment-friendly transportation mode for users; 2) it reduces road traffic congestion; and 3) it is a healthy lifestyle [31]. As a result, building effective bike lanes, demonstrated in Figure 1a, becomes a vital task for governments to promote the cycling lifestyle. Well planned & implemented bike lanes not only make cycling easier, but also reduce the safety risks for both cyclists and drivers of motor vehicles [30].

Traditional approaches to planning bike lanes in a city rely mainly on empirical experience and surveys [12, 18, 32]. With widespread availability of GPS embedded devices, more data-driven approaches on planning bike lanes have emerged, e.g., [10, 11, 19]. However, existing works [10, 11, 19] merely focus on summarizing commonalities of bike trajectory data while ignoring the realistic constraints and requirements faced by the government:

- **Budget Limitations.** There are costs to realizing a bike lane on a road segment, which may include: 1) the space for creating bike lanes; and 2) the price of building bike lane railing, and painting signs (demonstrated in Figure 1(a)). Unfortunately, governments often have limited budgets.
- **Construction Convenience.** To implement the bike lanes, construction teams need to be dispatched to construction zones, with the number of teams required also being a hard constraint. For the sake of ease of management, the government would like to avoid spreading teams out to construction zones in far reaching locations (e.g., red lines in Figure 1(b) highlights the top-100 segments with the most bike trajectories), and prefer to have



(a) Station-Less Deployment (b) Internet Connected & GPS Logged (c) A Mobike Trajectory Example

Figure 2: The Mobike Example.

them clustered, i.e., as a limited number of connected components in the road network.

- **Bike Lane Utilization.** As a public service, from the government's point of view, the objective of building bike lanes is to increase the usability for more bikers and cover more possible routes.

To incorporate these real world constraints, in this paper, we propose a data-driven approach for planning the bike lanes based on the massive number of trajectories collected from Mobike [1] users. Mobike is a fully station-less bike-sharing system currently deployed in many large cities in China. It is the world's largest bike operator, and recently made Shanghai the world's largest bike-share city. Compared to the traditional station-based bike sharing system, trajectories generated by Mobike users have two distinctive advantages in tackling the bike lane planning problem:

- **Realistic Travel Demands.** Unlike many existing station-based bike sharing systems, which require the users to pick up and drop off bikes from designated stations, Mobike offers a more flexible system, where the users can pick up and drop off their bikes at arbitrary locations (Figure 2(a)). As a result, the trajectories of Mobike users reflect actual urban travel demands.
- **Rich Travel Information.** A 3G communication component and a GPS module are embedded on the lock system in Mobike (demonstrated in Figure 2(b)), which enables the users to find bikes with their phones. It also keeps the track of the exact route traversed by the users (Figure 2(c)), while the traditional station-based bike sharing system can only provide the check-in/out information.

In this paper, we design, implement and deploy a data-driven bike lane planning system on Microsoft Azure, which not only leverages the massive bike trajectories generated by thousands of Mobike users, but also fulfills the constraints and objectives requested by the government. The proposed system contains two main components: 1) *Pre-Processing*, which pre-processes the trajectories from the Mobike user and maps them on the road network; and 2) *Bike Lane Planning*, which takes the user's input (i.e., requirements from the government) and provides bike lane suggestions. The main contributions are summarized as follows:

• We formulate the bike lane planning problem by considering various construction constraints, and propose a flexible tuning parameter to characterize the design trade-off between the number of covered users and the length of the continuously covered bike trips. The problem proves to be NP-hard.

• We propose a *greedy network expansion* algorithm, which provides a scalable and approximate solution to the data-driven bike lane planning problem. To achieve a better effectiveness, we also propose two different approaches to initialize the algorithm, which work well for low and high budget scenarios, respectively.

• We evaluate the proposed algorithms extensively over one month Mobike trajectory data (i.e., from 9/1/2016 - 9/30/2016) from the City of Shanghai. We also provide an extensive data analysis and discover many useful insights. Moreover, on-field case studies are conducted to evaluate the effectiveness of our bike lane recommendations.

• An online system with the real dataset is deployed and available on Microsoft Azure [2]. Finally, we collect the feedback from the government officials, from which our system received very positive reviews.

The rest of the paper is organized as follows: Section 2 describes the problem definition and the system overview. Section 3 presents the pre-processing module. Bike lane planning module is presented in Section 4. Experiments and case studies are given in Section 5. Section 6 presents the system deployment details and the expert reviews. Related works are summarized in Section 7. Section 8 concludes the paper.

2 OVERVIEW

In this section, we model and define the bike lane planning problem, and outline our solution framework.

2.1 Problem Definition

Given a road network graph $G = (V, E)$ (where a vertex set V represents intersections and an edge set $E = \{e\}$ represents all relevant road segments, our data-driven bike lane planning problem aims to discover a subset of edges $E' \subseteq E$, that follows three criteria: (i) construction budget constraint, (ii) connectivity constraint, (iii) maximum usage benefit.

Construction budget constraint. There is a monetary cost $e_i.c$ associated with each road segment e_i, to convert a road segment into a bike lane (e.g., building the railings and clearing the space). On the other hand, the government has an overall budget constraint B to building bike lanes, and the total cost of the bike lane construction cannot exceed the overall construction budget B, as highlighted in eq.(1) below.

$$\sum_{e_i \in E'} e_i.c \leq B. \tag{1}$$

Connectivity constraint. As has been outlined in the introduction section, for the construction and management convenience, the government prefers to deploy bike lanes with up to k connected components (e.g., to be assigned to k construction teams). The following inequality eq.(2) reflects such a constraint:

$$C(E') \leq k, \tag{2}$$

[1] https://en.wikipedia.org/wiki/Mobike

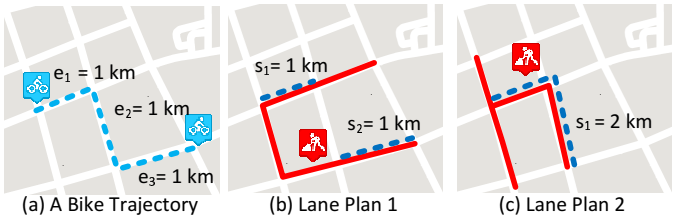


Figure 3: Motivation of Trajectory Score Function.

where $C(E')$ denotes the operator that counts the number of connected components from an edge set E'.

Maximum usage benefit. The goal here is to maximize overall usage of deployed bike lanes, which should 1) facilitate as many users as possible, and 2) cover more continuous road segments along their trip routes. Note that continuous road coverage in bike lane planning is crucial, as it increases the users' quality of experience (QoE). For example, a bike travels on a path (i.e., $e_1 \rightarrow e_2 \rightarrow e_3$), shown as blue dotted lines in Figure 3(a)). Though the two planned bike lanes (Figure 3(b) & (c)) cover the same lengths of the trajectory, *Lane Plan 2* in Figure 3(c) is preferred by users as it provides a longer continuous path, while the trajectory coverage of *Lane Plan 1* in Figure 3(b) is broken into two disconnected segments s_1 & s_2.

Unfortunately, these two objectives (i.e., serving more users vs. covering longer and continuous trips) usually conflict with each other, as user trips usually have different destinations. Hence, we propose a flexible score function for decision makers to adjust their preference between the two objectives for a trajectory tr_i:

$$tr_i.g = \sum_{s_j \in S_i} \alpha^{\frac{s_j.\ell}{min(e.\ell)}} \times \frac{s_j.\ell}{min(e.\ell)}, \alpha \geq 1. \quad (3)$$

where $tr_i.g$ is the beneficial score for trajectory tr_i, S_i is the set of continuous road segments that overlap with trajectory tr_i in the path plan E', s_j is one continuous road segments in set S_i, $\frac{s_j.\ell}{min(e.\ell)}$ normalizes the length of the continuous road segment $s_j \in S_i$ (where $min(e.\ell)$ is the minimum length of the road segment in the network), with the guarantee that its value is no less than 1, and α is the tuning parameter to set the preference on the number of covered users vs the length of continuous coverage. The reason for designing a score function using the exponential function of the normalized length is that when $\alpha > 1$, the continuous segment gets a higher score. Otherwise, without the exponential function $\alpha^{\frac{s_j.\ell}{min(e.\ell)}}$, *Lane Plan 1* and *Lane Plan 2* will have the same score. A smaller α indicates that more preference is given to the amount of user coverage (e.g., $\alpha = 1$ means that we do not care about the continuous length coverage, and two path plans in Figure 3 have the same beneficial score), while a larger α means that the longer continuous length coverage of the user trips is preferred.

Then, the overall beneficial score of a edge set (or a bike lane plan) $E'.g$ can be calculated by aggregating the scores of all the trajectories Tr that overlap with road segment set E':

$$E'.g = \sum_{tr_i \in Tr \& tr_i \cap E' \neq \emptyset} tr_i.g. \quad (4)$$

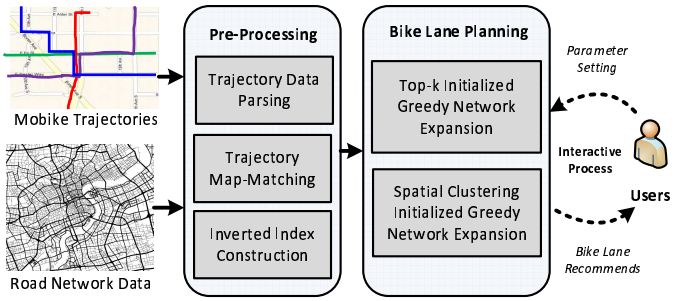


Figure 4: An Overview of System.

We now formalize our bike lane planning problem as follows.

Problem definition. *Given a set of trajectories Tr, a road network $G = (V, E)$ with a cost value $e_i.c$ on each edge e_i, a tuning parameter α, a value k, and a total construction budget B, we want to find a set of edges $E' \subseteq E$, which maximizes the total beneficial score g, and fulfills two constraints: 1) the total budget is no more than B; and 2) the number of connected components in E' is less than k.* Formally, it is represented as an integer programming problem:

$$\text{max: } E'.g, \qquad \text{s.t.: } \sum_{e_i \in E'} e_i.c \leq B, \qquad C(E') \leq k. \quad (5)$$

Such a problem of finding k budget constrained connected components with maximum beneficial score is NP-hard as proven in Lemma 1 below.

Lemma 1 (NP-difficulty). *Finding k budget constrained connected components with maximal beneficial score is NP-hard.*

Proof. We reduce our problem of finding k budget constrained connected components with a maximum beneficial score from the $0-1$ Knapsack problem. We can view each road segment $e_i \in E$ as an item, with an item size (i.e., construction cost), and an item profit (e.g., a beneficial score contribution). The set E' of selected road segments is viewed as a knapsack, with a fixed size B (i.e., total budget constraint). If we set $\alpha = 1$, i.e., we do not care about the continuous length coverage, and $k = |E|$, i.e., the maximum number of disconnected components is unbounded. Our problem boils down to a $0-1$ Knapsack problem.

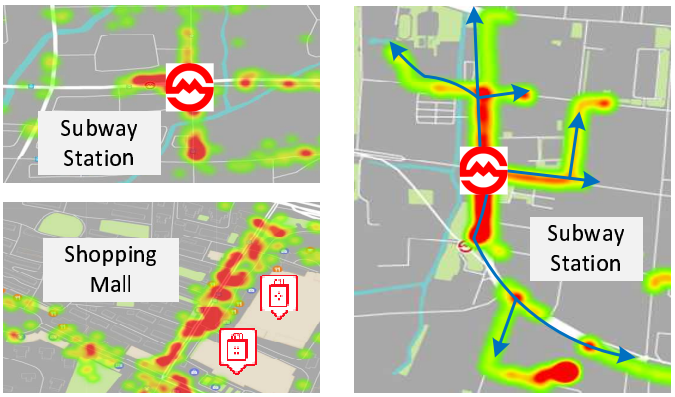
Thus, for any instance of the decision version of the $0-1$ Knapsack problem, we can find an instance of the decision version of the problem of finding k budget constrained connected components with the maximum beneficial score by setting $k = |E|$ and $\alpha = 1$, and their answers are the same. Thus, the decision version of the $0-1$ Knapsack problem is reducible to the decision version of our problem, which completes the proof of NP-difficulty. □

Given it is an NP-hard problem, we develop a greedy-algorithm based heuristic to tackle the issue.

2.2 System Framework

Figure 4 gives an overview of our system, which consists of two main components:

Pre-Processing. This component takes the bike trajectories and the road network and performs three main tasks: 1) *Trajectory Data*



(a) Spatial Hotspots (b) Star Like Mobility Pattern

Figure 5: Spatial Insights of Mobike Data.

Parsing, which removes the outlier GPS points; 2) *Trajectory Map-Matching*, which projects the bike trajectories onto the corresponding road segment; and 3) *Inverted Index Construction*, which builds an index to speed up the lookup process of retrieving trajectories based on road segment IDs (detailed in Section 3).

Bike Lane Planning. This component takes the user's parameters, e.g., the total budget, number of connected components, and the α value, and outputs the bike lane recommendation results. If the user is satisfied by the results, parameters can be tuned to get a new set of recommendations. We propose two different approaches for bike lane recommendation (detailed in Section 4).

3 PRE-PROCESSING

Pre-processing takes the road network and the trajectories as input, and performs the following three tasks to prepare the data for further processing:

Trajectory Parsing. This step cleans the raw trajectories from Mobike users by filtering the noisy GPS points with a heuristic-based outlier detection method [37].

Trajectory Map-Matching. In this step, the system maps each GPS point onto the corresponding road segment. We use a revised version of an interactive-voting based map matching algorithm [36], where the speed constraint of the road segments is not used, to perform map-matching.

Inverted Index Construction. In this step, the system builds the *inverted index* for each road segment, recording the trajectory IDs passing it. In this way, we can speed up the road segment-based trajectory look-up. The index construction process is done in parallel on Microsoft Azure [4].

4 BIKE LANE PLANNING

In this section, we first describe the overall framework of the *greedy network expansion algorithm* for planning bike lanes. After that, we describe the different approaches to initialize the network expansion.

Algorithm 1 Framework of Greedy Network Expansion

Input: Road Network $G = (V, E)$, Inverted index I, Trajectory Dataset Tr, Total budget B, tuning parameter α, and a value k.
Output: Result road segment set E'.
//Stage 1: Initialization
1: Road Segment Set $E' \leftarrow k$ starting road segments
2: Candidate set $C \leftarrow$ adjacent road segments of E'
3: Remaining Budget $B \leftarrow B - \sum_{e_i \in E'} e_i.c$
//Stage 2: Network Expansion
4: **while** Budget $B > 0$ **do**
5: $MaxGain \leftarrow 0$; $e_{next} \leftarrow \emptyset$
6: **for** $e_i \in$ Candidate set C **do**
7: **if** $e_i.c < B$ **then**
8: Retrieve trajectories Tr' from I based on $E' \cup e_i$
9: Calculate beneficial score difference per cost $\Delta g = \frac{g' - g}{e_i.c}$
10: **if** $MaxGain < \Delta g$ **then**
11: $MaxGain = \Delta g$; $e_{next} \leftarrow e_i$
12: $E' \leftarrow E' \cup e_{next}$; $B \leftarrow B - e_{next}.c$
13: Candidate Set $C \leftarrow C \cup$ none-selected adjacent edges of e_{next}
//Stage 3: Termination
14: return E'

4.1 Greedy Network Expansion Framework

Main Idea. The intuition of the greedy network expansion algorithm is to expand a set of k starting road segments in the network. This is inspired by the two key insights discovered in the dataset, namely *spatial hot spots* and *star-like mobility patterns*:

Spatial hot spots. Figure 5(a) shows the two hot spots with the highest number of trip starting locations, where the upper side reflects a subway terminal station (i.e., Jinyun Road Station of Subway Line 13), and the lower side illustrates a very popular shopping mall (i.e., Bailian Zhonghuan Commerce Plaza). The intuition behind the observation is straightforward: although the mall is very popular, it is not close to any subway stations, which makes cycling the best option; similarly for the terminal station, the fastest & most economic option to get home from there is cycling.

Star-like mobility patterns. We further investigate travel directions around spatial hot spots, and we discover that the bike trips go to different destinations from the same starting location, just like multiple edges with one shared end, namely, a star-like mobility pattern, as demonstrated by the arrows in Figure 5(b).

Taking these observations into considerations, our greedy-based bike lane planning algorithm extends the incremental network expansion algorithm in road network, e.g., [3, 28]. The algorithm has three phases:

- **Stage 1: Initialization.** The algorithm starts by selecting k starting road segments. In this way, we can guarantee that the final road segment recommendation produced by the algorithm fulfills the connectivity constraint, i.e., does not generate more than k connected components.
- **Stage 2: Network Expansion.** In this stage, the algorithm runs iteratively. In each iteration, the algorithm selects the best road segment (i.e., with the highest the beneficial score gain per cost, which equivalents to the ratio of item profit to size, in the classic 0 - 1 Knapsack problem)

Edge ID	e_1	e_2	e_4	e_5	e_6	e_7	e_8	e_{10}	e_{11}
Δ Gain	1	4	5	7	8	4	5	9	5
Cost	2	2	2	2	5	2	1	5	2

(a) A Candidate Set Example

(b) An Initialization Example

(c) An Network Expansion Iteration

Figure 6: Greedy Network Expansion Example.

to the result set E' and adds its none-selected adjacent segments to the candidate set.

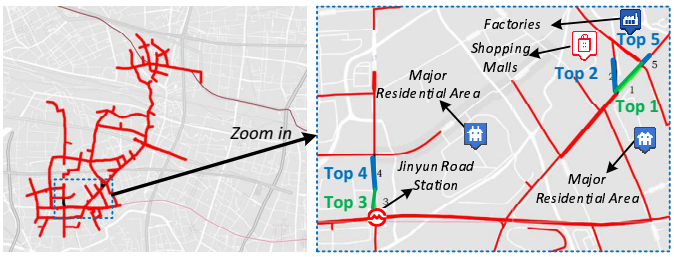
- **Stage 3: Termination.** The algorithm terminates when budget limit B is met, and then returns the resulting road segment set E' as the recommended bike lane plan.

Algorithm Design. Algorithm 1 gives the pseudo-code of our greedy network expansion algorithm. In the *initialization stage*, the algorithm first selects k starting road segments in the resulting set E', puts all the adjacent road segments of the start segments in candidate set C, and updates the budget value by subtracting the total cost of the starting road segments (Line 1-3).

In each iteration of the *network expansion stage* (Line 5-13), the algorithm checks each road segment e_i in the candidate set C. If the cost of the road segment is smaller than the remaining budget, the algorithm retrieves all the trajectories Tr that has been covered by the road segment e_i and the result road segment set E' (Line 8). After all the covered trajectories are retrieved, we calculate an updated beneficial score g' based on Equation 4. Then, we calculate the corresponding beneficial score gain per cost (Line 9). During the process, we keep track of the road segment e_{next}, which has the maximum beneficial score gain per cost in the iteration. e_{next} is inserted in to the resulting road segment set E', the remaining budget is updated by subtracting the cost of the selected road segment $e_{next}.c$. Road segment e_{next} is removed from candidate set C, and all of its none-selected adjacent segments of e_{next} are inserted in the candidate set C for further iterations (Line 10- 13).

Finally, when all the budget is used up, the algorithm terminates, and the road segment set E' is returned as the recommended plan.

Example. Figure 6 gives an example of the greedy network expansion algorithm. In the *initialization stage*, two starting road segments are selected (marked in red), and all of their adjacent segments are inserted in the candidate set (marked in blue). During the *network expansion stage*, in the iteration, we calculate the beneficial score gain difference for each segment in the candidate set (illustrated in Figure 6(a)), based on Equation 4. After that, we divide the beneficial score difference by the cost of each segment and select the highest one to expand the network, which is e_8 in our example. Then, the adjacent segments of e_8 are added as the



(a) Result of Top-k Initialization (b) Top-k Start Segments

Figure 7: Top-k Initialization Example.

new candidates (i.e., e_{12} and e_{13} in Figure 6(c)). The algorithm terminates when the budget is used up.

Analysis. As demonstrated in the example, it is clear that the performance of the final results E' is highly determined by the selection of the starting road segments. As a consequence, finding an effective method to perform initialization becomes a vital task in our greedy network expansion algorithm.

4.2 Top-k based Initialization

Main Idea. The most straightforward method is *Top-k Initialization*, which essentially selects the highest ranked k segments based on the beneficial score per cost (i.e., $\frac{e_i.g}{e_i.c}$), as the starting segments for network expansion. The intuition behind this approach is that these road segments usually represent the spatial hot spots, which should always be included in the final result.

Example. Figure 7(a) gives an example result of greedy network expansion with top-k based initialization, with $k = 5$. The recommended bike lanes are marked in red in the figure, which form one large set of connected components. The reason the result contains only one connected component, rather than five (i.e., k value) is that the top-5 highest ranked segments are connected with each other. Figure 7(b) is the detailed view of the boxed area in Figure 7(a), where the selected five starting road segments are marked in green and blue, which form two groups (i.e., {Top 1, Top 2, Top 5} and {Top 3, Top 4}). The first group contains the road segments between a major residential area and nearby shopping malls/factories, while the second group contains the road segments near the terminal station for subway Line 13. The reason the top ranked segments are usually connect to each other, is a large amount of trajectories may share a lot of road segments, as they traverse from or to the same location (e.g., a subway station or a shopping mall).

Analysis. The top-k based initialization approach guarantees that the algorithm will never miss any segment with the highest beneficial score per cost. However, as most of the top-k ranked segments are very close to each other, it can only expand with a much lower number of connected components in the network, which limits the search space in the candidate set and may miss some important areas, especially when the budget B is large.

4.3 Spatial Clustering-based Initialization

In order to include more spatially diversified starting locations in the initialization stage and be more effective when the budget is

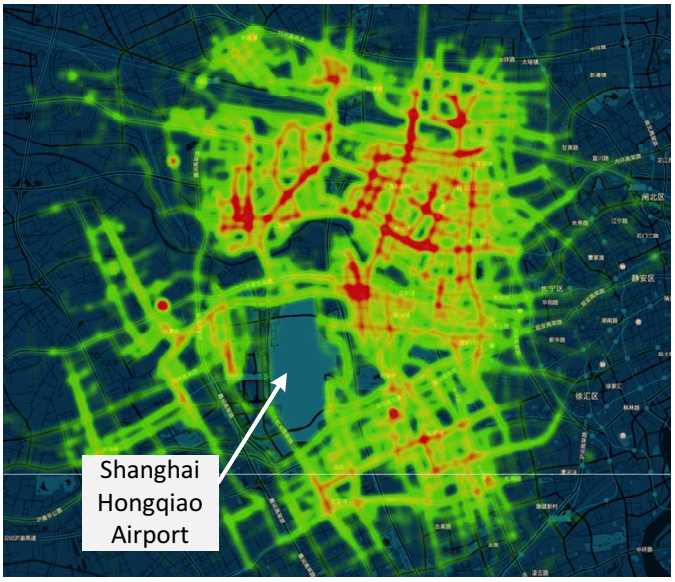


Figure 8: Mobike Trajectory Data Distribution.

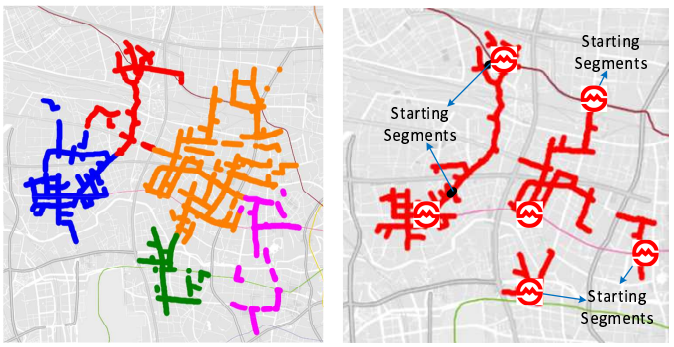
larger, we take advantage of spatial clustering techniques to select the starting road segments.

Main Idea. The intuition behind the *spatial clustering-based initialization* is from observing of the trajectory heat map (i.e., Figure 8), which visually has some rough clusters over the space. In this way, we can avoid the drawbacks of the top-k based initialization, which has the starting segments connected to each other and limits the search space. This method has two main steps:

- **Candidate Selection.** In this step, we select a subset of road segments with high ranks (e.g., top 1% ranked segments in our implementation based on the score per cost), as the candidates for clustering.
- **Spatial Clustering.** In this step, the candidate road segments are clustered based on an agglomeration hierarchical clustering method, e.g., [33]. After that, the highest ranked road segment in each cluster is selected as the starting segments.

The reason for selecting a subset of road segments for clustering is to remove the road segments that will never be in the final result and reduce computational cost. The hierarchical-based clustering method is employed in our system, as it does not need to tune the clustering parameters (e.g., in DBSCAN [9]) and it always generates stable results (unlike it is in K Means [14]). Thus, it is more intuitive for government users.

Example. Figure 9 gives an example of the execution results of spatial clustering-based initialization, where $k = 5$. In the first step, we compute the clusters generated by our algorithm, i.e., Figure 9(a). After that, the highest ranked road segments are selected as the starting segments, which are the black segments in Figure 9(b). It is interesting to note that four of the starting segments are at subway stations. The recommended paths actually cover the neighborhood of six subway stations, as illustrated in the figure.



(a) Results of Hierarchical based Spatial Clustering

(b) Results of Spatial Clustering based Greedy Network Expansion

Figure 9: Spatial Clustering based Initialization.

Analysis. Compared to the results generated by the top-k initialization method, spatial clustering based initialization clearly has better diversity and coverage. The main reason is that after the spatial clustering step, the starting segments are no longer connected with each other. As we will show in our experiments, with more budgets, the spatial clustering-based initialization method is more effective.

5 EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the effectiveness of our system. We first describe the dataset used in the paper. Then, we provide a detailed analysis on the mobility statistics of the Mobike trajectories. After that, we provide experiment results with different parameters. Finally, we present a set of real case studies to demonstrate the effectiveness of our system.

5.1 Datasets

Road Network. We use the road network of Shanghai, China from Bing Map, which contains 333, 766 intersections and 440, 922 road segments.

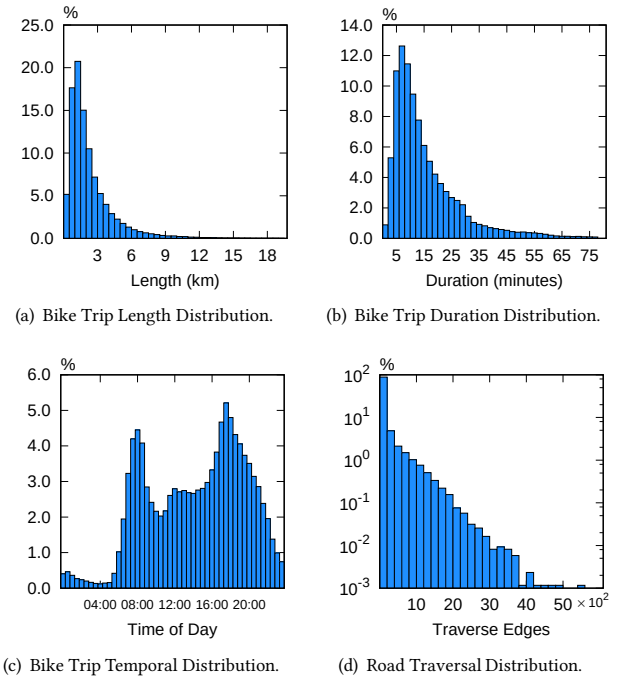
Mobike Trajectories. Each Mobike trajectory contains a bike ID, a user ID, a temporal range of the trajectory, a pair of start/end locations, and a sequence of intermediate GPS points.

The Mobike dataset is collected in one month (i.e., 09/01/ 2016 - 09/30/2016) from the city of Shanghai. (Figure 8 gives an overview of the spatial distribution of GPS locations). The dataset contains 13,063 unique users, 3,971 bikes, and 230,303 trajectories (with a total of 18,039,283 unique GPS points).

5.2 Mobility Statistics of Mobike Data

Trip Length Distribution. Figure 10(a) summarizes the trip lengths distribution of the Mobike users. From the figure, it is clear that the majority of the trajectories are relatively short, i.e., more than 70% of the trips are shorter than 2 km, as people primarily take bikes for shorter trips. The observation is consistent with the assumption that shared bike service is the solution for the "last mile problem" in public transportation systems [7].

Trip Duration Distribution. Figure 10(b) gives the trajectory duration distribution, where the majority of the trips are within



(a) Bike Trip Length Distribution. (b) Bike Trip Duration Distribution.

(c) Bike Trip Temporal Distribution. (d) Road Traversal Distribution.

Figure 10: Mobike Trip Characteristics.

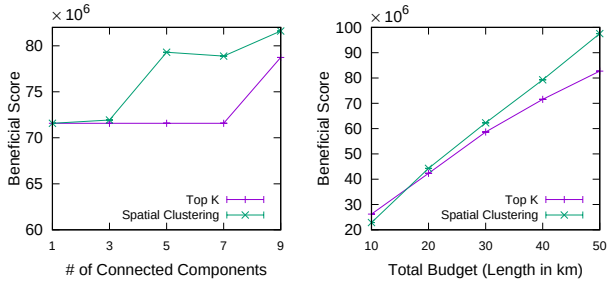
(a) 6:00AM – 8:00AM (b) 8:00AM-10:00AM

Figure 11: Temporal Imbalance Example of Mobike Trips.

30 mins. This is because: 1) most of the trips are less than 2 km, which should be completed within 15 mins, and 2) the pricing plan of Mobike charges a user one RMB per 30 mins (we also notice a sudden drop around the 30 min mark).

Trip Temporal Distribution. Figure 10(c) illustrates the distribution of the trip start time. It is obvious that there are two usage peaks, i.e., the morning/evening rush hours. It is interesting to see there is still a small amount of usage late at night, i.e., 10:00PM - 3:00AM, which is generated by the overtime workers.

Road Traversal Distribution. Figure 10(d) depicts the road segment distribution with respect to the number of traversed trajectories (in semi-log scale). It is obvious that most road segments are covered by less than 100 trajectories, which echoes that bikers have destinations all over the urban area. On the other hand, there are over 2,000 road segments, with more than 1,000 trajectories, which validate the necessity of planning effective bike lanes.



(a) Different k Values. (b) Different Total Budgets.

Figure 12: Effectiveness Evaluation.

Temporal Imbalance. Figure 11 gives the Mobike trajectory starting locations at different time periods, which exhibits significant temporal imbalance. For example, in the early morning, i.e., Figure 11(a), more trajectories start at the residential areas. However, around 08:00 a.m. to 10:00 a.m, more trips start at the subway station (as Figure 11(b)). After we analyze their final destinations, it is clear that in the early morning, people who live nearby ride bikes to the subway stations for work. Then, after one hour, more people arrive at the subway station and ride to nearby malls and offices.

5.3 Effectiveness Studies

In this subsection, we study the effects of different parameters in our system. Unless mentioned explicitly, the default parameters used in the experiments are: $k = 5$, total construction budget $B = 30KM$ (we use the length of the segment as the cost $e_i.c$, as the cost and the length are highly correlated), and $\alpha = 1$.

Different k Values. Figure 12(a) gives the total beneficial scores $E'.g$ of choosing different numbers of components (i.e., k values). As a result, we have the following insight: 1) in most cases, the spatial clustering-based initialization method gets a higher score; 2) the scores for Top-k method stays the same for $k < 7$, as all the top-7 segments are connected; 3) when k value is small, two methods are similar. This is because in these cases the starting segments of clustering results are the same as the top-k.

Different Total Budgets. Figure 12(b) illustrates the total scores with different total budgets, from 10 KM to 50 KM. From the figure, we make the following observations: 1) the spatial clustering-based initialization method performs better when the budget is larger. 2) when the budget is small, top-k method is better than spatial clustering based method. This is because, when the budget is small, the best strategy may be expanding the segment with the most number of trajectories (essentially the intuition of top-k method). However, when the budget is large, the segments with high scores per cost around the top-1 or top-2 ranked segments can be fully covered (as most bike trajectories is less than 2 KM). At this time, a more effective way should include the segments around other spatial hot spots, rather than still expanding around that top-1 or top-2 ranked segments.

Different α Values. Figure 13 provides the results with different α settings, with the spatial clustering based method, where the

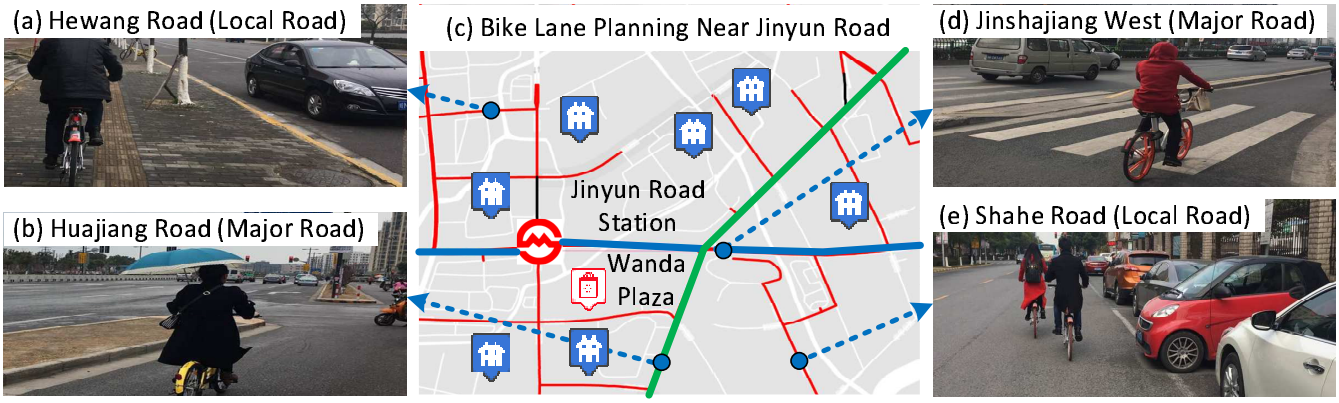
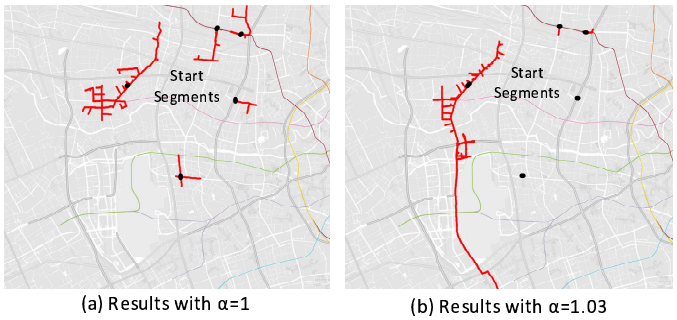


Figure 14: A Real Case Study Near Jinyun Road Station.



(a) Results with α=1 (b) Results with α=1.03

Figure 13: Effects of α Values.

red lines are recommended paths and the black dots are their start segments. It is interesting that, when α is large, most of the network expansions happened in one connected component. Moreover, with a higher α, the result of the expansion goes further on some major roads. The reason behind these two phenomena is that, when α is large, higher beneficial scores are given for covering more portion of the bike trajectories.

5.4 Case Study

To better understand the effectiveness of our bike lane recommendations, we conduct a field case study. We choose to visit the area near Jinyun Road subway station, as this area appears in all of our recommendations, regardless of the parameters.

Figure 14(c) gives an overview of the overall POI distribution of the area: 1) Jinyun Road is the terminal station of subway line 13, 2) there is a very large shopping mall (Shanghai Jiangqiao Wanda Plaza) next to the subway station; and 3) around the subway station, there are many populated residential areas within a 2 km radius, marked as the blue icons on the figure. As a result, cycling is the most convenient way for the residents in this area to go to the subway station or the shopping mall, which explains this area having the highest bike usage density in our dataset.

When we arrive at the Jinyun Road station, we discover that the government has built a few designated bike lanes. Based on our observation, the government plans these bike lanes with a simple strategy: building designated bike lanes for all major roads, and painting bike lanes for the most of the local roads.

For example, the major roads in the figure have designated bike lanes, which are the Jinshajiang West Road (i.e., highlighted in blue) and Huajiang Road (i.e., highlighted in green), as shown in our photos: (Figure 14(b) for Huajiang Road and Figure 14(d) for Jinshajiang West Road). These observations demonstrate the effectiveness of our system, as all of these major roads are included in our bike lane recommendation results.

On the other hand, there are no designated bike lane on local roads, e.g., Hewang Road (Figure 14(a)) and Shahe Road (Figure 14(e)). However, we observe that there is also extensive bike usage on these roads, as they are the paths to highly populated residential areas. Although there are painted bike lanes on the road, the cycling conditions are pretty bad. In Figure 14(a), the bike users have to ride on the sidewalk, as the original bike lane is taken by a parked car. As a consequence, it not only makes the cycling experience much worse, but also is potentially dangerous for people walking or running on the sidewalk. In the other example, i.e., Figure 14(e), at Shahe Road, the bike users are forced to ride on the main lane of the road, as all the space of the biking path is taken by cars, which may lead to traffic accidents.

As a result, based on our analysis, we conclude that the government's current strategy, i.e., building bike lanes only on major roads, is insufficient. With the real bike trajectories and data-driven analysis, we propose that the cycling conditions in these local road segments in our recommendation should be improved. For example, the government should build designated bike lanes, replace off-street parking spaces with (underground) parking garages, and enforce better management of illegal parking.

6 SYSTEM DEPLOYMENT & EXPERT REVIEW

In this section, we first describe the details of our deployed system on Microsoft Azure. After that, expert feedback from government officials are presented and summarized.

6.1 System Implementation

Our bike lane planning system is publicly available online [2], where the website user interface is implemented using bootstrap,

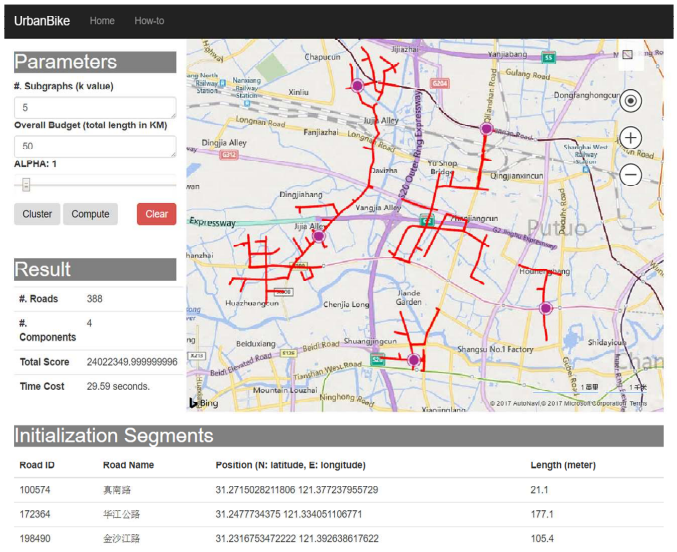


Figure 15: System Interface.

C#, Asp.NET and Bing Map V8 API, and the system is deployed on Microsoft Azure. Figure 15 is an example of the system interface. The system allows users to interact with it using different parameters, and get bike lane construction recommendations within a short time. The interface contains following components:

Parameters. This section of the interface allows users to input the parameters, such as the maximum number of connected components (or k value), total budget (we consider the length as the cost), and the α value (with a slider). There are two main buttons on this area: 1) *Cluster*, which shows the results of spatial clustering, after a user inputs the k value; and 2) *Compute*, which generates the bike lane recommendation, with all parameters.

Result. In this section, a table is used to show the algorithm results, including: 1) the number of road segments in the recommendation, 2) the total score based on the α and the recommended paths, and 3) total execution time of the task.

Initialization Segments. In this section, we show the list of the road segments, which are used to initialize the *greedy network expansion*. For each road segment, we present information, which includes: road segment ID, road name, center position (with latitude and longitude) and the length in km.

Main Map View. In the upper right section, there is our main map view. In this view, it displays the spatial clusters, distinguished using different colors, when the user inputs a value k and presses the *Cluster* button. It will also visualize bike lane recommendation results, when the user inputs all the parameters and clicks *Compute*. The starting road segments are illustrated as the purple dots and the recommended road segments are in red poly-lines.

6.2 Expert Review

We presented our system to the government officials from Xuhui District, Shanghai, and collected their feedback.

Overall, they highly appreciated our data-driven bike path planning approach and found the system is very useful to help their planning. One of the officials commended: " The idea of using the real sharing bike trajectories for planning the bike lanes is very reasonable. The data mining results from the system will serve as a very solid foundation for our urban planners to build more effective bike lanes in Shanghai".

7 RELATED WORK

In this section, we summarize the related works in three main areas: 1) data-driven urban planning, 2) trajectory data mining, and 3) traditional bike lane planning methods.

Data-Driven Urban Planning. With the availability of massive amounts mobility data from users, vehicles and public transportation systems, urban computing techniques have become more and more popular in many urban planning tasks, as the massive mobility data reflects real travel demands in the physical world [38]. For example, [39] mines patterns in taxi trajectories to suggest road constructions and public transportation projects. [35] infers different function zones in a city based on traffic patterns and POI distribution. [5, 17] identify potential traffic patterns and anomalies in the city based on multiple mobility datasets. In this paper, we focus on providing a data-driven approach to find a more effective and economic way for bike lane planning.

Trajectory Data Mining. The bike lane planning problem is related to the trajectory data mining [6, 16, 21, 22, 24, 26, 27, 34]. Many systems have been proposed to discover frequently used routes based on massive trajectory data, e.g., [6, 15, 16, 21, 26, 27]. There are also some projects on clustering/summarizing trajectories on the road network [13, 20], which help urban planners to know the popular routes and improve public transportation system. The closest projects on bike trajectory mining are [10, 11, 19], which focus on summarizing the trajectory commonality and find out the K-Primary Corridors for bike lanes. However, all of these works can not be directly used for bike lane planning, as they fail to consider the realistic budget and connectivity constraints.

Traditional Bike Lane Planning. Traditional bike lane planning in a city is mainly studied in the transportation domain, and relies heavily on the empirical experience, e.g., [8, 12]. To evaluate the necessity of building bike lanes, [29, 32] provide some high level suggestions based on public surveys and the geographical statistics, such as the road network and POI distributions. There have been some attempts [18] to systematically discover factors for actual bike route choices based on survey data. Recently, there have also been some works on traffic predication and route suggestion based on the station-based bike-sharing systems, e.g., [23, 25].

8 CONCLUSION

In this paper, we propose a data driven approach to plan bike lanes based on the real bike trajectories collected from Mobike (a major station-less bike sharing system) in the City of Shanghai. Our system can address the bike lanes planning problem in a more realistic way, considering the constraints and requirements from urban planners' perspective: 1) budget limitations, 2) construction convenience, and 3) bike lane utilization. We also propose a flexible

beneficial score function to adjust preferences between the number of covered users and the length of covered trips. The formulated problem is proven to be NP-hard, thus we propose a *greedy network expansion* algorithm with two different initialization methods: top-k based and spatial clustering.

We perform extensive experiments on a large scale Mobike data and demonstrate the effectiveness of our proposed bike lane planning framework, where interesting trade-off phenomena are observed namely the top-k based (resp. spatial clustering based) initialization approach works well with low (resp. high) construction budgets. We also conduct an on-field case study based on our path recommendation results, and present many important insights to improve cycling convenience in a given area. A demonstration system is deployed on Microsoft Azure for public use, and the expert feedback from the government officials from Xuhui District, Shanghai, confirms the effectiveness and usability of our system.

Finally, in future work, we plan to use the parallel computing framework in Microsoft Azure to improve system response time to work more efficiently with larger trajectory datasets. Also, we would like to further explore the interactive planning process to incorporate more human intelligence.

ACKNOWLEDGMENT

We thank Beijing Mobike Technology Co., Ltd. for providing the Mobike trajectories, Prof. Yingcai Wu from Zhejiang University for visualization suggestions, and Huichu Zhang from Shanghai Jiaotong University for conducting the on-field case study.

Yu Zheng was supported by the National Natural Science Foundation of China Grant No. 61672399, No. U1609217, and 973 Program, No. 2015CB352400.

Yanhua Li was supported in part by NSF CRII grant CNS-1657350 and a research grant from Pitney Bowes Inc.

REFERENCES

[1] 2015. Transport minister encourages people to get on their bike for Cycle to Work Day. https://www.gov.uk/government/news/transport-minister-encourages-people-to-get-on-their-bike-for-cycle-to-work-day. (2015).
[2] 2017. UrbanBike System. http://urbanbike.chinacloudsites.cn/. (2017).
[3] Jie Bao, Chi-Yin Chow, Mohamed F Mokbel, and Wei-Shinn Ku. 2010. Efficient evaluation of k-range nearest neighbor queries in road networks. In *MDM*. IEEE, 115–124.
[4] Jie Bao, Ruiyuan Li, Xiuwen Yi, and Yu Zheng. 2016. Managing massive trajectories on the cloud. In *SIGSPATIAL GIS*. ACM, 41.
[5] Sanjay Chawla, Yu Zheng, and Jiafeng Hu. 2012. Inferring the root cause in road traffic anomalies. In *ICDM*. IEEE, 141–150.
[6] Zaiben Chen, Heng Tao Shen, and Xiaofang Zhou. 2011. Discovering popular routes from trajectories. In *ICDE*. IEEE, 900–911.
[7] Paul DeMaio. 2009. Bike-sharing: History, impacts, models of provision, and future. *Journal of Public Transportation* 12, 4 (2009), 3.
[8] Jennifer Dill and Kim Voros. 2007. Factors affecting bicycling demand: initial survey findings from the Portland, Oregon, region. *Transportation Research Record: Journal of the Transportation Research Board* 2031 (2007), 9–17.
[9] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, and others. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise.. In *KDD*, Vol. 96. 226–231.
[10] Michael R Evans, Dev Oliver, Shashi Shekhar, and Francis Harvey. 2012. Summarizing trajectories into k-primary corridors: a summary of results. In *SIGSPATIAL GIS*. ACM, 454–457.
[11] Michael R Evans, Dev Oliver, Shashi Shekhar, and Francis Harvey. 2013. Fast and exact network trajectory similarity computation: a case-study on bicycle corridor planning. In *UrbComp*. ACM, 9.
[12] Geoff French, Jim Steer, and Nick Richardson. 2014. Handbook for cycle-friendly design. https://goo.gl/m3DwoY. (2014).
[13] Binh Han, Ling Liu, and Edward Omiecinski. 2012. Neat: Road network aware trajectory clustering. In *ICDCS*. IEEE, 142–151.
[14] John A Hartigan and Manchek A Wong. 1979. Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28, 1 (1979), 100–108.
[15] Abdeltawab M Hendawi, Jie Bao, and Mohamed F Mokbel. 2013. iRoad: a framework for scalable predictive query processing on road networks. *Proceedings of the VLDB Endowment* 6, 12 (2013), 1262–1265.
[16] Abdeltawab M Hendawi, Jie Bao, Mohamed F Mokbel, and Mohamed Ali. 2015. Predictive tree: An efficient index for predictive queries on road networks. In *ICDE*. IEEE, 1215–1226.
[17] Liang Hong, Yu Zheng, Duncan Yung, Jingbo Shang, and Lei Zou. 2015. Detecting urban black holes based on human mobility data. In *SIGSPATIAL GIS*. ACM, 35.
[18] Tetsuro Hyodo, Norikazu Suzuki, and Katsumi Takahashi. 2000. Modeling of bicycle route and destination choice behavior for bicycle road network plan. *Transportation Research Record: Journal of the Transportation Research Board* 1705 (2000), 70–76.
[19] Zhe Jiang, Michael Evans, Dev Oliver, and Shashi Shekhar. 2016. Identifying K Primary Corridors from urban bicycle GPS trajectories on a road network. *Information Systems* 57 (2016), 142–159.
[20] Ahmed Kharrat, Iulian Sandu Popa, Karine Zeitouni, and Sami Faiz. 2008. Clustering algorithm for network constraint trajectories. In *Headway in Spatial Data Handling*. Springer, 631–647.
[21] Xiaolei Li, Jiawei Han, Jae-Gil Lee, and Hector Gonzalez. 2007. Traffic density-based discovery of hot routes in road networks. In *International Symposium on Spatial and Temporal Databases*. Springer, 441–459.
[22] Yuhong Li, Jie Bao, Yanhua Li, Yingcai Wu, Zhiguo Gong, and Yu Zheng. 2016. Mining the most influential k-location set from massive trajectories. In *SIGSPATIAL GIS*. ACM, 51.
[23] Yexin Li, Yu Zheng, Huichu Zhang, and Lei Chen. 2015. Traffic prediction in a bike-sharing system. In *SIGSPATIAL GIS*. ACM, 33.
[24] Dongyu Liu, Di Weng, Yuhong Li, Jie Bao, Yu Zheng, Huamin Qu, and Yingcai Wu. 2017. SmartAdP: Visual Analytics of Large-scale Taxi Trajectories for Selecting Billboard Locations. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 1–10.
[25] Junming Liu, Leilei Sun, Weiwei Chen, and Hui Xiong. 2016. Rebalancing Bike Sharing Systems: A Multi-source Data Smart Optimization. In *SIGKDD*. ACM, 1005–1014.
[26] Wuman Luo, Haoyu Tan, Lei Chen, and Lionel M Ni. 2013. Finding time period-based most frequent path in big trajectory data. In *SIGMOD*. ACM, 713–724.
[27] Dev Oliver, Shashi Shekhar, Xun Zhou, Emre Eftelioglu, Michael R Evans, Qiaodi Zhuang, James M Kang, Renee Laubscher, and Christopher Farah. 2014. Significant route discovery: A summary of results. In *International Conference on Geographic Information Science*. Springer, 284–300.
[28] Dimitris Papadias, Jun Zhang, Nikos Mamoulis, and Yufei Tao. 2003. Query processing in spatial network databases. In *VLDB*. VLDB Endowment, 802–813.
[29] Kathryn M Parker, Janet Rice, Jeanette Gustat, Jennifer Ruley, Aubrey Spriggs, and Carolyn Johnson. 2013. Effect of bike lane infrastructure improvements on ridership in one New Orleans neighborhood. *Annals of behavioral medicine* 45, 1 (2013), 101–107.
[30] J Pucker. 2001. Cycling safety on bikeways vs. roads. *Transportation Quarterly* 55, 4 (2001), 9–11.
[31] David Rojas-Rueda, A De Nazelle, O Teixidó, and MJ Nieuwenhuijsen. 2012. Replacing car trips by increasing bike and public transport in the greater Barcelona metropolitan area: a health impact assessment study. *Environment international* 49 (2012), 100–109.
[32] Greg Rybarczyk and Changshan Wu. 2010. Bicycle facility planning using GIS and multi-criteria decision analysis. *Applied Geography* 30, 2 (2010), 282–293.
[33] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association* 58, 301 (1963), 236–244.
[34] Guojun Wu, Yichen Ding, Yanhua Li, Jie Bao, Yu Zheng, and Jun Luo. 2017. Mining Spatio-Temporal Reachable Regions over Massive Trajectory Data. In *ICDE*. 1–12.
[35] Jing Yuan, Yu Zheng, and Xing Xie. 2012. Discovering regions of different functions in a city using human mobility and POIs. In *SIGKDD*. ACM, 186–194.
[36] Jing Yuan, Yu Zheng, Chengyang Zhang, Xing Xie, and Guang-Zhong Sun. 2010. An interactive-voting based map matching algorithm. In *MDM*. IEEE Computer Society, 43–52.
[37] Yu Zheng. 2015. Trajectory data mining: an overview. *TIST* 6, 3 (2015), 29.
[38] Yu Zheng, Licia Capra, Ouri Wolfson, and Hai Yang. 2014. Urban computing: concepts, methodologies, and applications. *TIST* 5, 3 (2014), 38.
[39] Yu Zheng, Yanchi Liu, Jing Yuan, and Xing Xie. 2011. Urban computing with taxicabs. In *Proceedings of the 13th international conference on Ubiquitous computing*. ACM, 89–98.
[40] Xingxin Zhu. 2016. Bike sharing schemes promote green transport. http://www.telegraph.co.uk/news/world/china-watch/technology/sharing-bikes-to-promote-green-transport/. (2016).