Hindawi
Scientific Programming
Volume 2018, Article ID 3907513, 9 pages
https://doi.org/10.1155/2018/3907513

Research Article

Deployment Strategy for Car-Sharing Depots by Clustering Urban Traffic Big Data Based on Affinity Propagation

Zhihan Liu, Yi Jia, and Xiaolu Zhu

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

Correspondence should be addressed to Zhihan Liu; zhihan@bupt.edu.cn

Received 26 October 2017; Revised 19 January 2018; Accepted 11 February 2018; Published 15 March 2018

Academic Editor: Youngjae Kim

Car sharing is a type of car rental service, by which consumers rent cars for short periods of time, often charged by hours. The analysis of urban traffic big data is full of importance and significance to determine locations of depots for car-sharing system. Taxi OD (Origin-Destination) is a typical dataset of urban traffic. The volume of the data is extremely large so that traditional data processing applications do not work well. In this paper, an optimization method to determine the depot locations by clustering taxi OD points with AP (Affinity Propagation) clustering algorithm has been presented. By analyzing the characteristics of AP clustering algorithm, AP clustering has been optimized hierarchically based on administrative region segmentation. Considering sparse similarity matrix of taxi OD points, the input parameters of AP clustering have been adapted. In the case study, we choose the OD pairs information from Beijing's taxi GPS trajectory data. The number and locations of depots are determined by clustering the OD points based on the optimization AP clustering. We describe experimental results of our approach and compare it with standard K-means method using quantitative and stationarity index. Experiments on the real datasets show that the proposed method for determining car-sharing depots has a superior performance.

1. Introduction

Big data exists everywhere and is providing with kinds of large data sets which can make people's life more convenient and realize sustainable development [1]. Big data usually requires a set of techniques and technologies with new forms of integration to reveal insights from datasets that are diverse, complex, and of a massive scale [2]. Over the last few years, urban traffic data have been exploding, and we have truly entered the age of big data for transportation [3]. This situation inspires us to make some new attempts on urban traffic big data. In the paper, we propose a new attempt of urban traffic big data to determine the locations of car-sharing depots.

Car-sharing systems intend to offer an alternative model of car rental, by which users are permitted to use vehicles charged by hours [4]. These respects are used to evaluate car-sharing systems, namely, urban traffic environment, depots' layout, and rental mode. As for the urban traffic environment, intuitively big cities are a good choice because they have high demand for public transportation. With regard to depot locations, car sharing is a most important long-term decision owing to the fact that it has a direct impact on quality, efficiency, and cost of service and affects profit and market competitiveness. Deploying car-sharing depots based on demand has been a big challenge due to the lack of realistic vehicle operational data. Therefore, a detailed study of determining depot locations precisely would be necessary.

To maximize profits by distributing the depots rationally, the following three respects should be considered. (1) Consumers aspect: the ideal distance of walking on foot is 0–3 km. If the distance is too long, the willingness of users to rent vehicles will decrease significantly. (2) Return on investment aspect: good locations of depots will significantly improve overall earnings. From existing car-sharing systems, we know that car sharing is overwhelmingly concentrated in metropolitan cores; around 95% of members are found in these settings [5]. For example, Autolib' is a full electric car-sharing service in Paris. Up to July 2016, it offers over 1000 depots that can be found within a 5-minute walk in Paris. (3) Feasibility of depots construction aspect: generally speaking, car-sharing depots should be located in hot spots,

such as shopping malls, office building parking lot, and transport hub. These sites usually have enough available parking, and cost of construction is relatively low. On the whole, consumers hope they can rent vehicles as conveniently as possible. However, car-sharing service providers aspire to earn more and spend fewer on constructing depots at the same time. Considering these factors in an integrated manner, a frequently visited area by taxis is a good choice, namely, taxi hotspots. With widespread traffic sensors, urban traffic data is easily acquired and becomes of large scale. There are many methods to discover taxi hotspots from taxi GPS trajectory data. However, not all taxi hotspots are well suited for car-sharing depots. Having available parking spots is necessary for building car-sharing depots. Origin and Destination points (OD points) of users' trips can be extracted from taxi GPS trajectory data, which reflect traffic hotspots and indicate the potential demand of car sharing. Based on the above theory, we propose a method to discover the traffic hotpots by clustering taxi OD points and determine the locations of car-sharing depots.

As there are so many clustering algorithms, different clustering algorithm gives different clusters. It is important to choose an appropriate clustering algorithm to make a balance between time cost and performance. One of the most popular clustering algorithms is K-means. However, K-means works well only when the number of clusters is known before clustering. It is exciting that another popular clustering algorithm, AP (Affinity Propagation) clustering algorithm, can determine the number of clusters spontaneously. Nevertheless, the complexity of AP is unacceptable, particularly when the dataset is of large scale. To improve the computing complexity of AP, this paper proposes an optimization method based on administrative region segmentation and sparse representation of similarity matrix. The results of this study demonstrate the benefit of large-scale data to determine the locations of car-sharing depots. The results can provide some guidance and suggestion to government and car-sharing service providers in the early stage of car-sharing system construction. Although this study only uses a specific city as a case, the proposed method and framework are also applicable to other cities.

The contributions of this paper mainly lie in the following two aspects:

(i) We propose a novel optimization approach to determine the depot locations by clustering massive OD points with AP algorithm based on administrative region segmentations. We propose a method based on density to optimize the parameter of AP and briefly introduce the principle and application range of AP clustering method for sparse similarity matrix.

(ii) We implement experiments on a large-scale dataset about containing ninety thousand OD points extracted from taxi GPS trajectories generated by about 12,000 taxis in Beijing. Our method produces about 50 points suited for the car-sharing depots. Then we evaluate our model with the net similarity between optimized AP and K-means. The results show that our AP has an advantage over K-means. All the experiments show that our method is feasible and effective in determining car-sharing depots by clustering.

The remainder of the paper is organized as follows. Section 2 introduces some related works about locations of depots and taxi GPS data briefly. Section 3 presents the details of our method to determine the locations of car-sharing depots. Section 4 discusses the experimental results and analyzes the results. Conclusions and future work are discussed in Section 5.

2. Related Works

The majority of researches on determining the locations of car-sharing depots are dealing with urban traffic big data. In this section, we review some of the existing works.

2.1. Determining the Location of Depots. Urban big data enables a highly granular and longitudinal system, and it can help us understand city system and service better [6–10]. It can be used in many fields such as planning and governing cities, and business. For example, [3] applies big data to traffic flow prediction. Reference [11] presents a model to evaluate train timetable from the viewpoint of passengers' data on rail transit lines. Reference [12] proposes a study about public electric vehicle charging stations using traffic big data.

Many kinds of urban traffic big data are used for the depot locations problem. Reference [13] presents an approach to optimize locations of depots in one-way car-sharing systems in which vehicle stock imbalance issues are solved by three trip selection schemes. Reference [14] presents a method to optimize the locations of bike sharing stations and the fleet dimension and measures the bicycle relocation activities required in a regular operation day. Reference [15] develops a simulation model that considers demand variability and one-vehicle relocation policy and tests the solutions provided by the previous MIP model. Reference [16] analyzes the performance of the car sharing service across all stations, estimates the key drivers of demand, and uses these drivers to identify future locations of depots. Reference [17] determines the locations of depots based on the predicted car-sharing demand. The basis and premise of determining the candidate depots have been given. However, it is still difficult to determine the candidate depots so that a detailed study is extremely necessary.

2.2. Analyzing Taxi GPS Trajectory Data. Taxi GPS trajectory data is an important and effective type of urban traffic big data for analyzing some certain problems about transportation. More and more researches begin to focus on taxi GPS data in recent years. There are a number of works on analyzing taxi GPS data. Reference [18] uses taxi GPS data to analyze traffic congestion changes around the Olympic games in Beijing. Reference [19] presents a method to construct landmarks-nodes graph. Landmarks are defined as frequently traversed road segments by taxis. They present an approach to split adaptively a day into different time segments based on the entropy and variance of the travel time between landmarks. This brings up an estimative distribution of the travel times

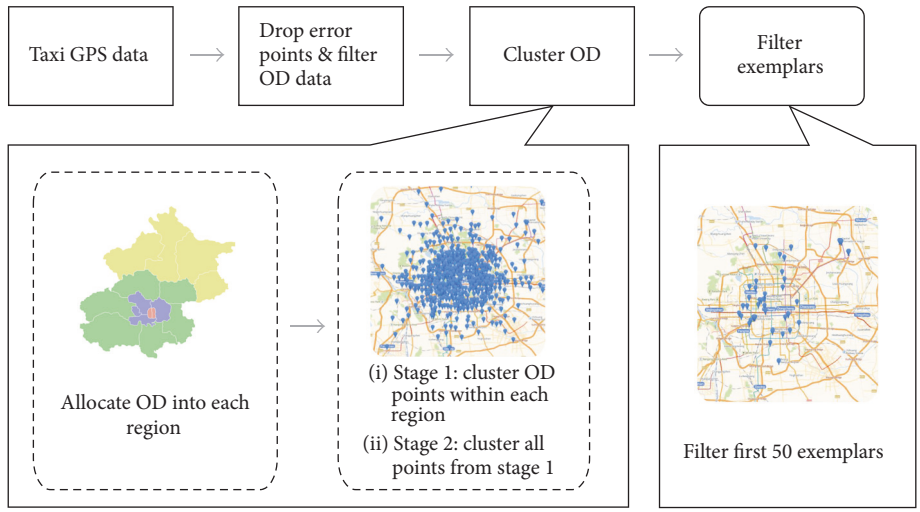

Figure 1: The framework of our proposed method.

between landmarks. Reference [20] proposes a method to construct a model of traffic density based on a large scale of taxi trips, which can be used to predict the traffic conditions and estimate the effect of emissions on the city's air quality. Reference [21] develops a method to identify traffic hotspots based on taxis GPS data, which is based on the method of clustering taxi GPS data by K-means algorithm. However, the most obvious problem of K-means is that it needs an input parameter K, which means you must know how many traffic hotspots in advance. From above all, for depot locations problem, more attention to taxi GPS data analyzing is needed.

3. Methodology

This section focuses on introducing our method in detail, which aims at finding suitable locations and the number of car-sharing depots while satisfying consumers demand and minimizing the total cost. Assuming that the total demand of car-sharing depots is unknown but positively associated with taxi flows, our approach is to cluster taxi OD points and then find hotspots from continuous taxi GPS trajectories, which could be considered as the locations of car-sharing depots. Our architecture of this paper is shown in Figure 1. The framework consists of three major components: filtering raw data, clustering OD points, and the final exemplars filter. The detailed process will be introduced in the following sections.

3.1. Filtering Raw Data. Filtering the efficient points from taxi GPS trajectory data is the necessary preparation, because not all the trips are efficient. For example, some error data caused by the breakdown of the GPS-equipment or some invalid data cannot reflect the character of traffic flows validly. The location of car-sharing depots is determined by the travel demand of travelers. We just filter the origin and destination points of passengers' trips which reflect the travel demand to some extent. And the OD points can be extracted from the continuous GPS trajectories according to trigger event.

Each taxi GPS point is described by a set of six elements: taxi id, trigger event, operation status, time, longitude, and latitude of GPS. "Taxi id" is the license of the car, which is a unique identifier for each taxi. "Trigger event" is the event that represents the taxi's trigger status. When the trigger event is equal to 0, that means the taxi turns to the "no-load" status from others. And 1 means turning to "load," 2 means "fortified," 3 means "withdraw garrison." "Operation status" is the operation status of the taxi. 0 means "no-load," 1 means "load," 2 means "Parking," and 3 means "off-the-line." "Time" is taxi's current time (SGT), the format is "mm-dd-hh-mm-ss". "Longitude" is the GPS coordinates of the taxi (East longitude and North latitude). For example, a data record (1143, 1, 1, 1106123843, 116.556101, 39.963646), it means the taxi 1143 was turning to "no-load" and the current time is 12:38:43 6th November, Beijing Time, and the taxi was located at 116.556101°E and 39.963646°N. To satisfy our demand, we need to filter points which trigger event has a sudden jump from 1 to 0 or 0 to 1, namely, OD points. It is worth mentioning that all the OD points are sorted by birth time.

3.2. Clustering the OD Points. In order to determine the locations of car-sharing depots, we make the cluster analysis on OD points based on AP clustering algorithm.

3.2.1. New Preference $\{s(k,k)\}$ in AP Clustering. Firstly, we review the standard AP model [22]. For N data points, the input is a set of pairwise similarities $\{s_{ij}\}$, where s_{ij} is the similarity of point j to point i, and a set of exemplar preferences $\{p_j\}$, where p_j is the preference for choosing point j as an exemplar. Generally, preference p_j is set as the similarity s_{jj} and influences the final number of identified exemplars. The goal is to select a subset of data points as exemplars and assign every nonexemplar points to the corresponding exemplar, so as to maximize the overall sum of similarities between points and their exemplars. There are two kinds of message exchanged between data points, namely,

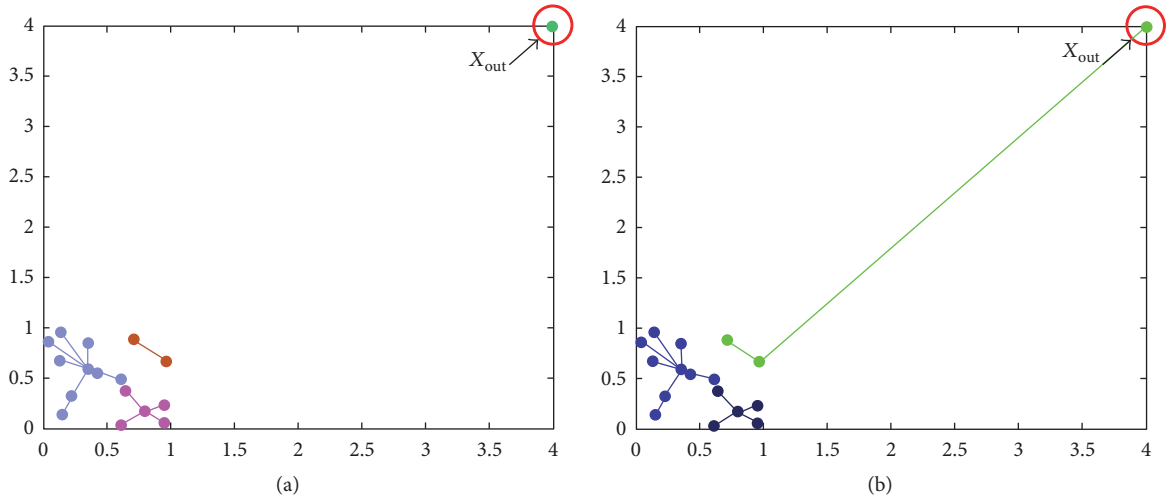

Figure 2: Outliers belonging to in different preference.

responsibility $r(i,k)$ and availability $a(i,k)$. To begin with, the availability values $a(i,k)$ are set to zero and the responsibility values $r(i,k)$ are set to the input similarity between point i and k. The message's updates of AP are computed as follows:

$$r(i,k) \longleftarrow s(i,k) - \max_{k' \text{s.t.} k' \neq k} \left\{ a\left(i,k'\right) + s\left(i,k'\right) \right\} \tag{1}$$

$$a(i,k) \longleftarrow \begin{cases} \min\left\{0, r(k,k) + \sum_{i' \text{s.t.} i' \notin \{i,k\}} \max\left\{0, r\left(i',k\right)\right\}\right\} & i \neq k \\ \sum_{i' \text{s.t.} i' \neq i} \max\left\{0, r\left(i',k\right)\right\} & i = k, \end{cases} \tag{2}$$

where $r(i,k)$ represents the evidence for how well point x_k serves as the exemplar of x_i. $a(i,k)$ reflects the evidence for how appropriate x_i choosing x_k as its exemplar. Equation (1) indicates that the update of $r(i,k)$ decreases the similarity $s(i,k)$ by removing the corresponding candidate exemplars from competition. Equation (2) represents the update process of $a(i,k)$ and gathers evidence from data points as to whether each candidate exemplar would make a good exemplar.

The above update rules require only simple, local computations that are easily implemented, and messages need only be exchanged between pairs of points with known similarities. At any point during affinity propagation, availabilities and responsibilities can be combined to identify exemplars. For point i, the value of k that maximizes $a(i,k)+r(i,k)$ either identifies point i as an exemplar if $k = i$, or identifies the data point that is the exemplar for point i.

AP considers all data points as potential exemplars. It takes as input a set of similarity $s(i,k)$, while $s(k,k)$ is set by input preferences. In this paper, similarity is set to be negative Euclidean distance: for points x_i and x_k, $s(i,k) = -\|x_i - x_k\|^2$. Note that the preference can be used to control the number of final exemplars, with low preferences leading to small number of exemplars and high preferences leading to large number of exemplars. Generally, the preferences of all data points are set to be the median of the input similarities so that all data points are equally suitable as exemplars. That declares no prior inclination toward particular data points as exemplars.

However, it would lead some outliers to generate corresponding clusters which consist only relatively small data points in traditional AP algorithm. For example, Figure 2(a) shows the outliers in the AP clustering procedure, and the data points in the upper right corner would form a single cluster far from with the other clusters. However considering the scenario of this paper, outliers are the points where taxis seldom pass by. From the economic point of view, the outliers are not suitable to be individual candidate car-sharing depots. Therefore, we prefer to merge outliers into nearest high-density cluster. Based on the purpose, we propose a new formulation of the input preference as follows:

$$s(k,k) = \frac{1}{N-1} \sum_{i=0, i \neq k}^{N} s(k,i), \quad \forall k \in N. \tag{3}$$

The new preference $\{s(k,k)\}$ is set to be the average of similarities between point x_k and others. This value is related to density around the point. The higher the density, the bigger the value; meanwhile the point is more like to be chosen as the exemplar. Figure 2(a) presents the cluster result which preferences are set to be the median of all the input similarities. We can see outlier X_{out} becomes a cluster consisting only itself. Figure 2(b) presents the cluster result in which preferences are set followed by (3). Obviously, X_{out} belongs to the nearest relative high-density cluster.

3.2.2. AP Clustering Based on Administrative Region Segmentation. It is intuitively plausible that AP's runtime is $O(N^3)$ per iteration. However, as [23] presents, sharing computations allow us to compute messages efficiently which can reduce runtime within $O(N^2)$ per iteration. Figure 3 presents the curve of runtime with the total number N ranging from 5000 to 30000 by 5000 per step. We can find that the runtime increases dramatically with N rising. Results of experiments confirm the conclusion. While the N is thirty

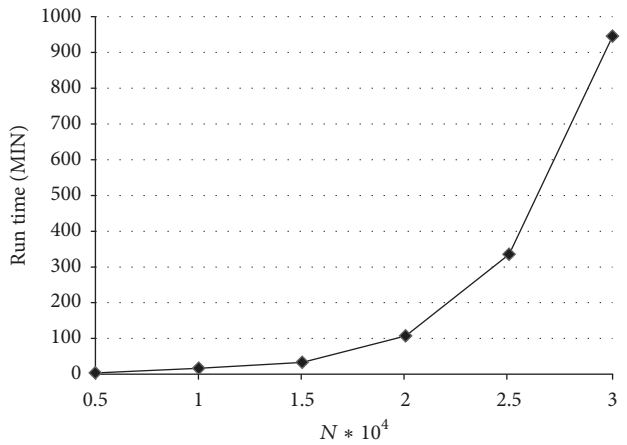

Figure 3: Runtime curve over the number of data points.

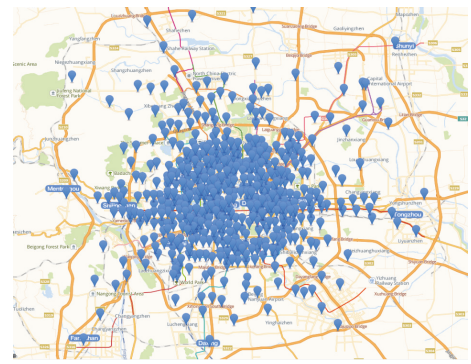

Figure 4: All exemplars in map.

thousand, runtime is about 15 hours. If the N becomes larger, it is unacceptable. For example, the number of OD points within twenty-four hours is about one hundred thousand, and it may cost hundreds of hours.

In order to solve the above problem, we propose an optimized AP clustering method based on administrative region segmentation. Suppose that N OD points are distributed uniformly in R administrative regions. There are four following major steps of our method:

(1) Allocate all OD points into different sets by administrative regions. The set of OD points in each region is named as AdminRegion$[x]$, $x = 1, 2, \ldots, R$. Each region has nearly N/R OD points.

(2) Implement standard AP in each AdminRegion$[x]$, and then we get a set of exemplars named AdminRegionCenter$[x]$ for each AdminRegion$[x]$. The time complexity of this step is $O(RT(N/R)^2) = O((1/R)TN^2)$, where T is the number of iterations.

(3) Repeat the two steps above for every day's OD points. Then we have each region's exemplars in each day: AdminRegionCenter[day]$[x]$, day $= 1, 2, \ldots, D$, $x = 1, 2, \ldots, R$.

(4) For each region, put all the exemplars of AdminRegionCenter[day]$[x]$ to a set, day $= 1, 2, \ldots, D$. Implement AP on these sets separately. Then we have the final exemplars of each region.

For simplicity, we mark step (1) to (3) as stage 1 and step (4) as stage 2.

While AP based on administrative region segmentation works well, it still takes too much time due to the mass data. For example, the iterations of Haidian District in Beijing need 12 hours, which is obviously unacceptable. One improvement is that sparse similarity matrix to AP is applied. AP exchanges message between each point. If the similarity between two points is too low, the message between them is so few that we can set it to zero. In other words, we can set a minimal and acceptable similarity threshold ε, then the computational formula of similarity becomes as follows:

$$s(i,j) = \begin{cases} s(i,j) & \text{if, } (s(i,j) > \varepsilon) \\ -\infty & \text{if } (s(i,j) < \varepsilon). \end{cases} \quad (4)$$

Then, we transform similarity matrix $\{s(i,j)\}$ to sparse matrix. That means we just need to compute the nonempty elements. We can use triples to store the sparse matrix and calculate based on some techniques to reduce the complexity. For N OD points, $\{s(i,j)\}$ have N^2 elements. Through above method, it becomes M elements. Obviously, $M < N^2$. The time complexity of step (2) can be decreased to $O((1/R)TM)$ from $O((1/R)TN^2)$. When the scale of dataset is quite large and the dataset is widely distributed, the method is obviously efficient.

Despite the fact that the sparse method can shorten the time, it also brings some problems. The similarity between two points involved in the calculation is limited in sparse threshold ε. The procedure of finding exemplars limited in a certain distance will increase the number of exemplars.

After above optimization procedure, AP clustering method based on administrative region segmentation now can be implemented to get a set of exemplars. These exemplars are selected from actual data points, informally called "centers." It can be considered as a subset of representative points of all the points. So we take these exemplars as traffic hotspots. These traffic hotpots can be regarded as potential candidate depots.

Figure 4 shows clustered exemplars from thirty thousand OD points of Beijing. We find that 82 percent exemplars are located within the Fifth Ring Road. As we all know, the Fifth Ring Road is the boundary of densely populated area. At the macroscopic level, these cluster exemplars results conform to the reality very well.

3.3. Exemplars Filter. Based on the previous steps, we have a set of exemplars. However, we cannot simply regard all the exemplars as car-sharing depots. In some cases, some exemplars are so close that they almost overlap (Figure 4). What is more, the number of points in each cluster is different. Some exemplars can represent many points while others represent only a few. Therefore, we count the point number

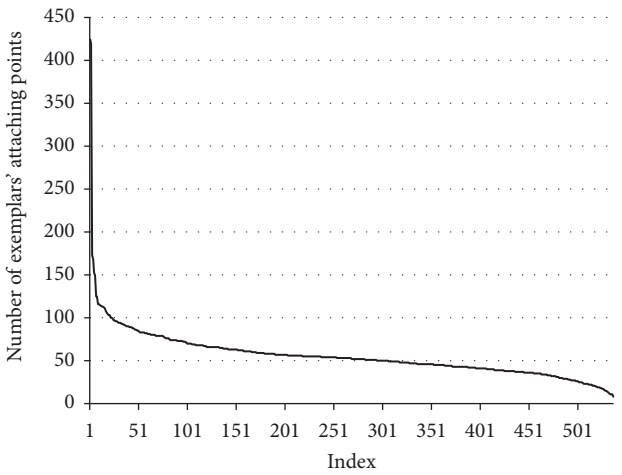

Figure 5: Count of number of exemplars' attaching points.

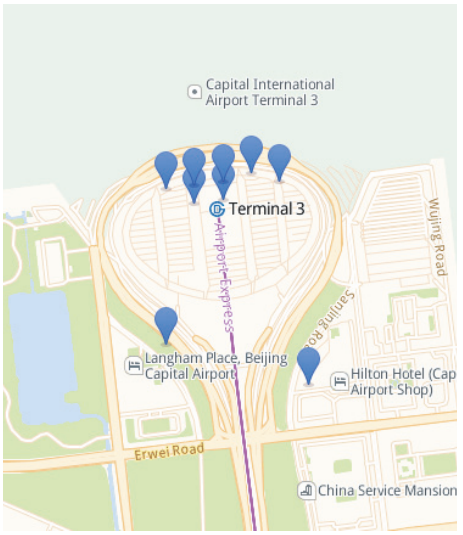

Figure 6: Exemplars overlap.

of each cluster and rank it from the largest to the smallest. As shown in Figure 5, the x-axis is the indexes of exemplars, and the y-axis is the number of points, which have be clustered to the exemplar. We find a turning point in Figure 5, and it is about (24, 100). It means that only 24 exemplars have 100 or more attaching points. This interesting result is caused by the principle of AP. AP aspires to find an exemplar for each point, and it cannot remove outliers spontaneously. Considering the goal of this paper is to find high-density areas, we can only take the first 24 exemplars to make a brief analysis and evaluation.

Figure 6 shows part of the points in Terminal 3 Building of Beijing Capital International Airport before filtering. Figure 7 shows the points after filtering. Compared with Figure 6, points in Figure 7 are no longer overlapping. Figure 8 shows the positions of the first 24 exemplars. We can find out that the most of exemplars are hotspots in our life. It is especially sensitive to train stations, large-scale business, and residence districts.

Figure 7: One of first 24.

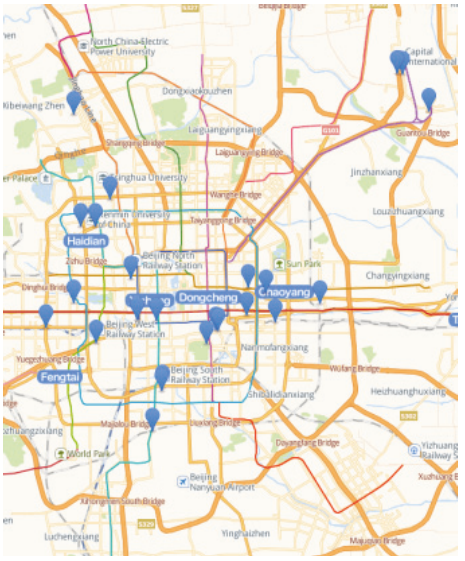

Figure 8: Exemplars first 24.

4. Experiment and Results

In order to prevent noise interference and determine appropriate locations of car-sharing depots, we take a week's taxi GPS data consisting of seven hundred thousand points in Beijing. Thus, the target of this section is to apply "administrative region segmentation" model to the large-scale dataset.

In the following sections, we describe the datasets used for the experiment in Section 4.1. Section 4.2 presents the details of the experiment based on the large-scale dataset, and finally we analyze the results in several respects.

4.1. Case Study Datasets. To determine the locations of car-sharing depots, we need to obtain two types of data, namely, OD points obtained from taxi GPS trajectory data and the boundaries of administrative regions.

Table 1: The results of our method.

Region	N	1st stage K (mean preference)	1st stage K (new preference)	2nd stage K (sparse AP)	2nd stage K (standard AP)
Haidian	237820	21094	9125	4075	304
Chaoyang	206602	14457	9009	3355	268
Fengtai	176047	11534	6845	2625	219
Xicheng	100602	4890	3492	1190	159
Dongcheng	93261	4313	3134	1042	146
Shijingshan	40137	1848	1816	582	84
Daxing	15901	1612	1577	605	73
Shunyi	15385	1321	825	325	32
Tongzhou	4936	715	664	281	43
Mentougou	3185	304	271	97	25
Fangshan	1263	267	236	104	26
Sum	895139	62355	36994	14281	1379

(1) Taxi GPS Trajectory Data. The GPS trajectory data acquired in GPS-equipped vehicles represents the mobility patterns of the citywide human, from which we can get the origin and destination points of each taxi trip. The OD points can be extracted from continuous GPS trajectory data by trigger event.

In this paper, we utilized a GPS trajectory dataset generated by 12,000 taxis in Beijing from 5 to 11 November 2012. The GPS trajectory dataset consists of about seven hundred thousand GPS points. And we extracted 895,139 OD points from GPS trajectory dataset. We define the following notations for simplicity:

(i) $Q = \{1, 2, \ldots, q, \ldots, Q\}$: set of taxi GPS trajectory data

(ii) $N = \{1, 2, \ldots, n, \ldots, N\}$: set of OD points which are filtered out from Q

(iii) $K = \{1, 2, \ldots, k, \ldots, K\}$: set of locations of car-sharing depots

(2) The Boundaries of Administrative Regions. The boundaries of administrative regions are used to divide the OD points, which consist of polygon vertexes. We obtain it from Baidu map API (https://api.map.baidu.com/library/CityList/1.4/docs/symbols/BMapLib.CityList.html).

4.2. The Results. In this paper, OD points are of large scale and dense, and the span of space is large. It satisfies the specification and requirement of sparse AP completely. Therefore, we use the sparse AP in stage 1 of Section 3.2. Following the steps in Section 3, the results of different input preference and divided two stages are summarized in Table 1. Firstly, we focus on the comparison between different input preferences. The "mean preference" means that the input preference is the mean of all similarities in stage 1. The "new preference" means that the input preference of stage 1 is the expected value of similarities calculated by (4). We found that the number of clustering exemplars based on mean preference is larger than that based on new preference, because the new preference results in allocating the outliers into corresponding high-density areas and decreasing the total number K. Secondly, compared with the results of sparse AP and standard AP of stage 2, we find that the total number of sparse AP is larger than standard AP's. The average elements in clusters of the sparse AP are 2.59, while the average elements in clusters of standard AP are 26.83. The former looks so bad because of incorrect use of sparse AP. We have mentioned that we use the sparse AP in stage 1, which causes the similarity between any two points of the input data in stage 2 and is limited over the sparse threshold. Therefore, in stage 2, it is difficult to find exemplars over the sparse threshold validly once more. When we adopt the standard AP in stage 2, the cluster results become better. Therefore, it is important to verify whether the dataset is suitable to use sparse AP. In summary, stage 1 with new preference and stage 2 with standard AP are the best choice.

For more information, we compare the results between AP and K-means in net similarity, which is defined as follows:

$$netSimilarity = \sum_{k} \left(s(k,k) + \sum_{i} s(i,k) \right). \quad (5)$$

Net similarity measures the appropriateness degree of how exemplars explain the data. It is the objective function that AP and K-means try to maximize. We can use the net similarity to evaluate the performance of the clustering methods. As we can see in Figure 9, the net similarity of AP based on administrative region is a little larger than K-means in each region, especially in big regions such as Haidian District. This is due to sparse AP used in stage 1 is more suitable to large-area regions. Meanwhile in smaller regions, such as Xicheng District, our method does not show an outstanding advantage.

The results present that there are 1379 exemplars alternative to car-sharing depots. However, as we know, not all the exemplars are suitable, and we need to filter these exemplars manually. Following the method in Section 3.3, we analyze the number of points which should be clustered to the 1379

Table 2: Comparison with AP and K-means in time cost (min).

N	K	K-means	AP
5000	160	0.006457	3.503131
10000	262	0.016079	16.63446
15000	339	0.019477	33.17552
20000	402	0.027256	107.2591
25000	463	0.047055	335.4124
30000	535	0.059033	945.0208

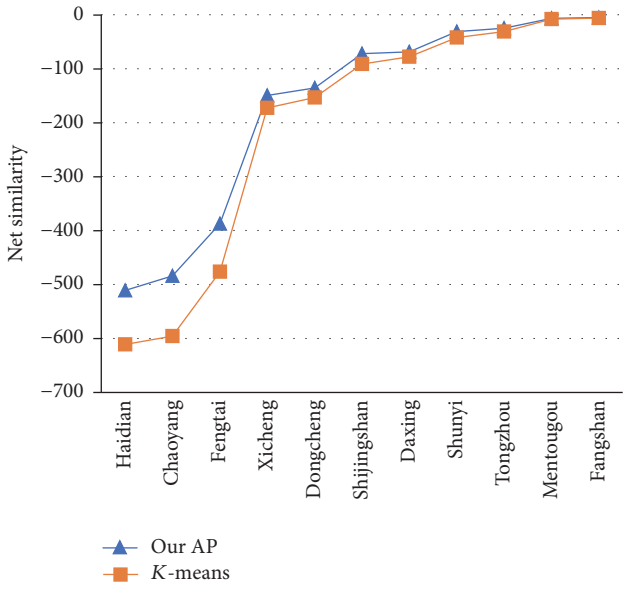

Figure 9: Net similarity.

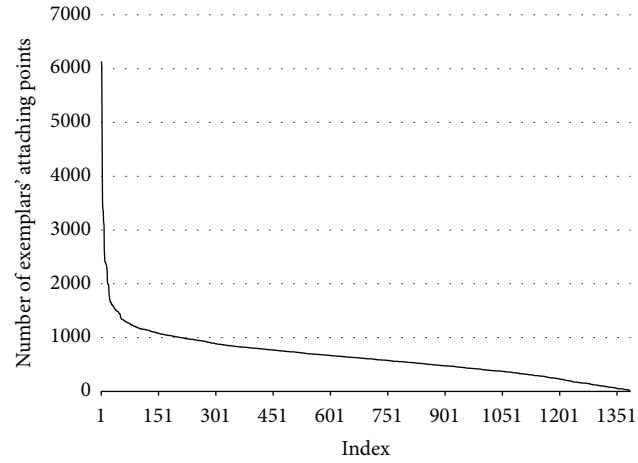

Figure 10: Count of number of exemplars' attaching points.

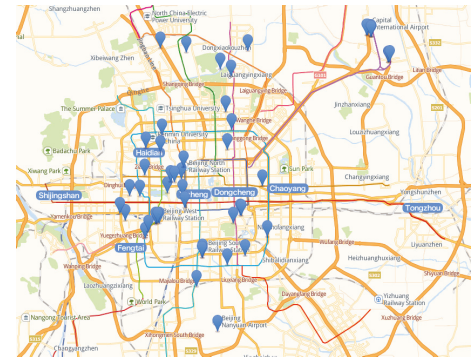

Figure 11: Exemplars first 50.

exemplars, respectively, as shown in Figure 10. Similarly, we find a turning point and its coordinate is about (43, 1500). Therefore, we take first 50 exemplars to make a brief analysis and evaluate. As Figure 11 presents, we can find that our method is sensitive to most of the traffic hotpots, especially to stations, airports, shopping centers, and hospitals. From the most direct sense, this result is as expected.

To evaluate the time cost of our AP and K-means method, we make another experiment with 5000 to 30000 OD points based on the steps above. The results of time cost are summarized in Table 2. Apparently, the time cost of AP far outweighs the K-means. It is noteworthy that the results showed in Line 3 time cost of K-means, which just executes for just once, where we set K as a result of AP. That is to say, we set the final cluster number of AP as the input K of K-means. In fact if there is no result of AP, we must determine the approximate K value through different attempts and experiments iteratively. It is hard to predict the range of K on the premise of massive data. However, the results do not convince that the time complexity of K-means performs better than AP. If we perform K-means iteratively to determine the approximate K, in the worst case scenario, the time complexity of K-means would be

$$\sum_{k=1}^{N} O(T * N * K) = O\left(T * N * \frac{N(N+1)}{2}\right) = O\left(N^3\right), \tag{6}$$

and meanwhile the time complexity of AP is $O(N^2)$, which is less than K-means. Therefore, the combination of AP and K-means is the future direction of this paper to reduce the time complexity of K-means.

5. Conclusions

In this paper, we do a new trial on urban traffic big data about the determination of car-sharing depots. Experiments are implemented on the taxi GPS trajectory data in Beijing consisting of large-scale taxi OD points to study deployment strategy for car-sharing depots. To solve the highly complex problem caused by the large-scale data set, we present an optimization AP clustering method based on administration region segmentation. We define a new preference formula

to solve the outliers problem in the process of clustering. In addition, we apply sparse AP on our method to decrease time cost. Combining theories and practices, we present the scope of application of our method. Meanwhile we have compared the objective function of AP and K-means in common, namely, net similarity. AP can not only overcome the biggest weakness of K-means, which is that K-means cannot determine K (the number of clusters) by itself, but also perform better in the net similarity. In spite of some measures taken to minimize the runtime of AP, it still takes too long. Thus, in practical applications, we can combine AP with K-means to achieve better performance. Although this study only takes Beijing as a case, the result of this paper indicates that this method has good versatility, because the method is not restricted to the data set itself. The methodological framework is applicable to any city only if the data set is available.

Owing to the fact that our method is simply based on clustering using taxi OD points, we have not considered much about whether the hotpots have the capacity to be car-sharing depots. Meanwhile, simply dividing dataset by regions is not rigorous. It maybe breaks some relevance especially in high-density areas. In this case, AP clustering based on grid segmentation is more reasonable. The grid segmentation means that taxi OD points are partitioned into multiple nonoverlapping grids to simplify representation of huge data points into smaller subsets. We plan to make further studies in these aspects.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the National Science and Technology Major Project (2016ZX03001025-003) and Special Fund for Beijing Common Construction Project.

References

[1] M. Batty, "Big data and the city," *Built Environment*, vol. 42, no. 3, pp. 321–337, 2016.

[2] I. A. T. Hashem, I. Yaqoob, N. B. Anuar, S. Mokhtar, A. Gani, and S. Ullah Khan, "The rise of 'big data' on cloud computing: review and open research issues," *Information Systems*, vol. 47, pp. 98–115, 2015.

[3] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.

[4] T. D. Schuster, J. Byrne, J. Corbett, and Y. Schreuder, "Assessing the potential extent of carsharing a new method and its implications," *Transportation Research Record*, no. 1927, pp. 174–181, 2005.

[5] A. Millard-Ball, G. Murray, J. Schure T et al., "Car-Sharing: Where and How It Succeeds," *Tcrp Report Transportation Research Board of the National Academies*, 2005.

[6] R. Kitchin, "Urban Big Data," *The Planner*, 2016.

[7] D. Tian, Y. Yuan, H. Qi et al., "A dynamic travel time estimation model based on connected vehicles," *Mathematical Problems in Engineering*, vol. 2015, Article ID 903962, 11 pages, 2015.

[8] D. Tian, J. Hu, Z. Sheng, Y. Wang, J. Ma, and J. Wang, "Swarm Intelligence Algorithm Inspired by Route Choice Behavior," *Journal of Bionic Engineering*, vol. 13, no. 4, pp. 669–678, 2016.

[9] X. Cao and X. Zhang, "Anomaly digging approach based on massive RFID data in transportation logistics," *International Journal of Big Data Intelligence*, vol. 1, no. 3, p. 166, 2014.

[10] F. Yang, S. Wang, J. Li, Z. Liu, and Q. Sun, "An overview of internet of vehicles," *China Communications*, vol. 11, no. 10, pp. 1–15, 2014.

[11] Z. Jiang, C.-H. Hsu, D. Zhang, and X. Zou, "Evaluating rail transit timetable using big passengers' data," *Journal of Computer and System Sciences*, vol. 82, no. 1, part B, pp. 144–155, 2016.

[12] H. Cai, X. Jia, A. S. F. Chiu, X. Hu, and M. Xu, "Siting public electric vehicle charging stations in Beijing using big-data informed travel patterns of the taxi fleet," *Transportation Research Part D: Transport and Environment*, vol. 33, pp. 39–46, 2014.

[13] G. H. D. A. Correia and A. P. Antunes, "Optimization approach to depot location and trip selection in one-way carsharing systems," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 1, pp. 233–247, 2012.

[14] L. M. Martinez, L. Caetano, T. Eiró, and F. Cruz, "An Optimisation Algorithm to Establish the Location of Stations of a Mixed Fleet Biking System: An Application to the City of Lisbon," *Procedia - Social and Behavioral Sciences*, vol. 54, pp. 513–524, 2012.

[15] D. Jorge, G. Correia, and C. Barnhart, "Testing the Validity of the MIP Approach for Locating Carsharing Stations in One-way Systems," *Procedia - Social and Behavioral Sciences*, vol. 54, pp. 138–148, 2012.

[16] V. P. Kumar and M. Bierlaire, "Optimizing locations for a vehicle sharing system," in *Proceedings of the in Swiss Transport Research Conference*, 2012.

[17] X. Zhu, J. Li, Z. Liu, and F. Yang, "Optimization Approach to Depot Location in Car Sharing Systems with Big Data," in *Proceedings of the 4th IEEE International Congress on Big Data, BigData Congress 2015*, pp. 335–342, USA, July 2015.

[18] H. Wen, Z. Hu, J. Guo, L. Zhu, and J. Sun, "Operational Analysis on Beijing Road Network during the Olympic Games," *Journal of Transportation Systems Engineering and Information Technology*, vol. 8, no. 6, pp. 32–37, 2008.

[19] J. Yuan, Y. Zheng, C. Zhang et al., "T-drive: driving directions based on taxi trajectories," in *Proceedings of the 18th International Conference on Advances in Geographic Information Systems ACM SIGSPATIAL (GIS '10)*, pp. 99–108, November 2010.

[20] P. S. Castro, D. Zhang, and S. Li, "Urban traffic modelling and prediction using large scale taxi GPS traces," in *Pervasive Computing*, vol. 7319 of *Lecture Notes in Computer Science*, pp. 57–72, Springer, Berlin, Germany, 2012.

[21] Z. Yunpeng, Z. Gang, and L. Jian, "A novel method for traffic hotspots recognition based on taxi GPS data," *Journal of Beijing Information Science & Technology University*, vol. 31, no. 1, pp. 43–47, 2016.

[22] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *American Association for the Advancement of Science: Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[23] I. Givoni, C. Chung, and B. J. Frey, *Hierarchical affinity propagation*, 2012, https://arxiv.org/abs/1202.3722.