Hindawi
Mobile Information Systems
Volume 2017, Article ID 7653706, 11 pages
https://doi.org/10.1155/2017/7653706

Research Article

Recovering Individual's Commute Routes Based on Mobile Phone Data

Xin Song,[1] Yuanxin Ouyang,[1] Bowen Du,[1] Jingyuan Wang,[1] and Zhang Xiong[1,2]

[1]*School of Computer Science and Technology, Beihang University, Beijing, China*
[2]*Research Institute of Beihang University in Shenzhen, Shenzhen, China*

Correspondence should be addressed to Bowen Du; dubowen@buaa.edu.cn

Received 20 September 2016; Revised 17 November 2016; Accepted 12 December 2016; Published 9 February 2017

Academic Editor: Qingchen Zhang

Mining individuals' commute routes has been a hot spot in recent researches. Besides the significant impact on human mobility analysis, it is quite important in lots of fields, such as traffic flow analysis, urban planning, and path recommendation. Common ways to obtain these pieces of information are mostly based on the questionnaires, which have many disadvantages such as high manpower cost, low accuracy, and low sampling rate. To overcome these problems, we propose a commute routes recovering model to recover individuals' commute routes based on passively generated mobile phone data. The challenges of the model lie in the low sampling rate of signal records and low precision of location information from mobile phone data. To address these challenges, our model applies two main modules. The first is data preprocessing module, which extracts commute trajectories from raw dataset and formats the road network into a better modality. The second module combines two kinds of information together and generates the commute route with the highest possibility. To evaluate the effectiveness of our method, we evaluate the results in two ways, which are path score evaluation and evaluation based on visualization. Experimental results have shown better performance of our method than the compared method.

1. Introduction

Human mobility has been a significant research area in recent years. An improved understanding of human mobility is meaningful in lots of fields, such as predicting the spread of disease, evaluating the effect of human travel on the environment, and urban planning.

Common questionnaire based ways to obtain the information of human mobility have quite a lot of flaws. Dealing with questionnaires can cost huge amount of manpower and spend quite a lot of time. Due to the huge cost of manpower and time, the number of samples is limited. Another problem is that people are easily affected by subjective factors which can make the result of the questionnaire unstable. To address all these problems, there are plenty of researches studying human mobilities based on GPS data [1] and taxi location trajectories data [2]. But these data have one big flaw, which is that they cannot cover the people wildly enough. Differently, mobile phone users have a wild coverage not only in developed countries but also in developing countries. Besides, mobile phone data contain rich information that can be used in multiple domains, such as revealing people's travel trajectory [3, 4], mining important locations [5], finding the spatial nature of human mobility [6], assisting the demographic census [7], and studying communication network [8].

Due to the wide coverage and tremendous information embedded, mobile phone data are quite suitable for analysing human mobilities. Individual's commute route is an important part of human mobility. Specifically, knowing people's commute routes has great importance in terms of at least three conspicuous aspects: (1) traffic flow analysis: we can infer each road segment's level of congestion from multiple people's commute routes; (2) urban planning: after the new road is mended, we can observe whether there is a change of individuals' commute routes so as to judge the effectiveness of the new mended road; (3) personalized services: according to many other people's common commute routes, individuals can be recommended commute routes and way of transportation based on their home and work places.

We are facing three main challenges. (1) The first is the low precision of location information embedded in the data. We can only use the cell towers coordinates to approximately represent people's real history locations. And all the cell phones within the distance of 1000 meters from the tower can receive its signal. So this will cause the low precision of location information. (2) The second challenge is that the time interval between two adjacent signal records of one mobile device can be quite long, which sometimes can reach one hour. So the sampling rate of the location trajectory is extremely low. (3) The third challenge is from the fact that people may have multiple commute paths. Changing the transportation tools always means changing the route, so finding the most possible path from all the overlapping everyday commute paths is our last challenge.

Facing all these challenges, we propose a commute routes recovering model to recover individual's commute route from mobile phone data. The model includes two main modules: data preprocessing module and map matching module. The first module extracts commute trajectories from raw dataset and formats the road network into a better modality. The second module combines two kinds of information together and generates the commute route with highest possibility. To the best of our knowledge, this is the first work focusing on recovering commute routes through mobile phone data of multidays. On the whole, this paper offers the following contributions:

(i) We design a data preprocessing module to form the data into suitable modalities for the route recovering task. For the mobile phone data, we apply the leader clustering algorithm to cluster the nearby cell towers and adopt an important location detecting strategy to find individual's home and work place. For the road network data, we design one road segmentation algorithm and one road merging algorithm to format the road segments.

(ii) We design the map matching module which firstly fuses the trajectory information extracted from mobile phone data with real world road network data and then adopts a path generating algorithm to generate the path with highest possibility.

(iii) To evaluate the effectiveness of the proposed model, we design two evaluating methods: path score evaluation and evaluation based on visualization. The path score evaluation calculates one score for each path, which considers the number of nearby cell towers of each path. Then we design the visualizing part drawing all the related data on the map to show the whole process of path recovering. This can directly and clearly show the relevance between the raw trajectory data and the generated commute path and can prove the better performance of our model.

The rest of this paper is structured as follows. Section 2 reviews the related work. Section 3 describes the mobile phone data and the real world road network data we used in this paper. We introduce the whole framework of the proposed method for recovering individual's commute routes in Section 4. Section 5 shows the experimental results and visualization of all the commute information. And the paper is concluded in Section 6 with a brief discussion of limitations and directions of future research.

2. Related Work

2.1. Application of Mobile Phone Data. Applications of mobile phone data have been a hot spot of research areas in recent years, which is mainly due to the wild coverage of mobile devices among people. Besides, there is rich information embedded in the mobile phone records which can be used in multiple domains, such as revealing peoples travel trajectory [3, 4], mining important locations [5], finding the spatial nature of human mobility [6], assisting the demographic census [7], protecting the identity, location, and sensitive information [9], and studying communication network [8].

Researches on human mobilities mainly focus on mining peoples mobility patterns [10, 11] and identifying important locations [12]. Differently, our work focuses on the specific commute routes of people which is quite meaningful in transportation-related areas.

2.2. Multimodal Data Fusion. With the era of big data coming, multiple kinds of data have been generated in different domains. Researchers from all over the world try to solve problems based on various data. Data from different domains always have multiple modalities, each of which has a different representation, distribution, scale, and density [13].

To find the traffic regularity between city areas, Zheng et al. [14] adopted a two-stage model, which firstly partitions a city into regions by major roads using map segmentation method [15] and then maps the GPS trajectories of taxicabs onto the regions to formulate a region graph. DNN-based model can be used to learn new feature representations through data with same modality [16, 17] and data with different modalities [18, 19] while concerning data privacy [20, 21]. Xin et al. [22] present a multisource active transfer learning framework for entity resolution task. Blum and Mitchell [23] employ cotraining method using a large unlabeled sample to boost performance of a learning algorithm when only a small set of labeled examples is available. Wang et al. [24] fuse multiple features in face recognition task. A new method for multiview dimensionality reduction is proposed by Zhang et al. [25]. Zhang et al. cluster incomplete multimedia data based on tensor distance [26, 27]. Rong et al. [28] utilise association rules to add group information to personal profiles.

In our work, we fuse cell tower location trajectories with real world road network data by the transfer matrix defined in our proposed model, which is different with existing approaches.

2.3. Map Matching. Map matching problem refers to the task of matching a raw trajectory to roads on a digital map. Map matching algorithms can be categorized into local/incremental algorithms [29] and global algorithms [30] according to the range of sampling points considered when matching the trajectories [31].

Yuan et al. [32] propose an Interactive Voting-Based Map Matching algorithm to solve the problem of low sampling rate GPS trajectories. GPS signal is recorded no longer than every 2 minutes, but the time interval between two mobile phone records can be nearly half an hour. And considering the low precision of location information, traditional methods cannot be used on the mobile phone data.

Thiagarajan et al. [33] propose an energy-efficient system for trajectory mapping using raw position tracks obtained largely from cellular base station fingerprints. The mobile phone data used in their work have a much higher signal sampling rate which are different from the data recorded by the real mobile operators, so they could reconstruct the route just based on one day trajectory. Our work is based on the real world mobile phone data which means the sampling rate is extremely low, so we combine the trajectories in multidays to increase transfer information. To the best of our knowledge, our work is the first to recover people's commute routes based on multidays' mobile phone data.

3. Data Description

In this section, we introduce two datasets used in this paper, which are mobile phone dataset and road network dataset. We extract individuals commute trajectories in multiple days from mobile phone dataset. The road network dataset is used to map the cell tower trajectories to real world road paths.

3.1. Mobile Phone Dataset. The mobile phone data used in this paper were collected during the period from October 24, 2013, to March 24, 2014, in Wuxi, China, containing about six million users equally spread over space. All the users can totally generate 40 million raw records each hour everyday which include huge amount of location information recorded in form of cell-id, area-id which can singly represent one cell tower. Based on the geographical data which contain the coordinate of each cell tower, we can easily transfer the cell tower id into coordinates. Each record in the raw dataset contains four parts: user id, cell tower id, time stamp, and tag. The time stamp can record the precise time when this record was recorded. The tag shows the specific activity one record stands for. The records are generated when the users are engaged in communication via the cellular network. Specifically, the records are recorded at the beginning and the end of each voice call placed or received, when a short message is sent or received, and when Internet is connected. So the cell tower's coordinate can approximately represent the user's history locations.

To better overcome the challenge of low sampling rate in our experiment, we tend to choose the devices that can generate stable and adequate data. The rule is that the chosen devices need to be recorded at least 24 records in each day during the selected period. Besides, we remove the individual's data whose home and work place are the same.

Administrative region of Wuxi includes three main parts: two small towns and one larger city. Figure 1 shows the spatial distribution of cell towers in Wuxi; as we can see, urban area has higher density of towers than the suburb area.

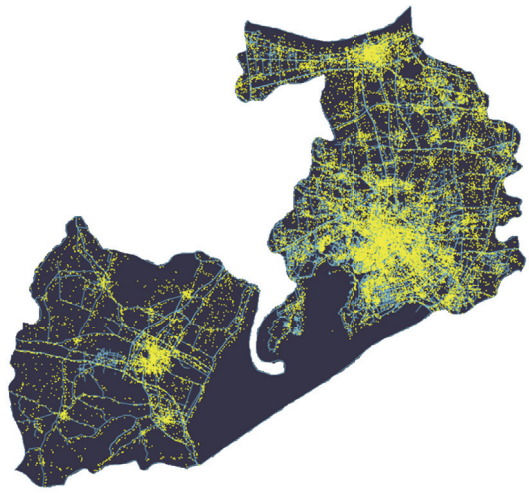


Figure 1: Spatial distribution of cell towers.

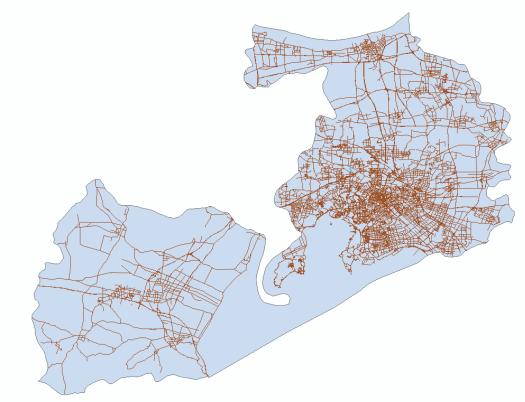


Figure 2: Real world road network in Wuxi.

3.2. Road Network Dataset. The road network dataset used in our work contains real world road information in Wuxi. Figure 2 shows the spatial distribution of all the road segments. Similarly, there are much more road segments in urban areas than the suburb. Each road segment in the dataset includes multiple points which are sampled from the corresponding real world road. The length of each raw road segment varies a lot. And there are 12158 road segments in the dataset. Based on the dataset we can take the whole information of real world roads into consideration when generating the commute routes of individuals.

Due to the information privacy concern, all the data are anonymous; we note that no private data are used in the experiment.

4. Model

As is shown in Figure 3, the framework of commute route recovering model contains two main modules: data preprocessing module and map matching module.

Data preprocessing is the first step of the proposed method for recovering individuals commute route. Data

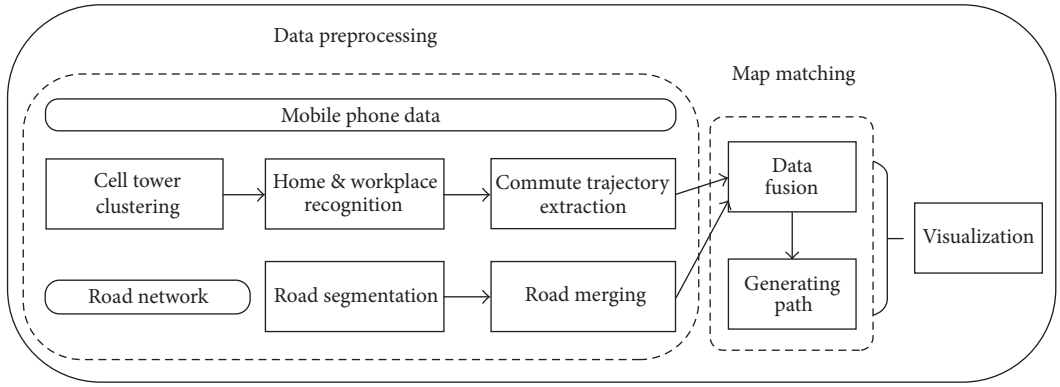


Figure 3: Framework of commute route recovering model.

preprocessing module contains two main parts: trajectory extraction and road network formation. Trajectory extraction submodule includes three steps: raw data clustering, important location detection, and commute trajectory extraction, aiming to extract individuals' everyday paths from home to work place in form of cell group trajectories. Then the road network formation submodule contains two steps: road segmentation and road merging. This module aims to reconstruct the raw road network data into suitable formation which can reduce the calculating time and increase the accuracy of the result.

Map matching module includes two steps. The first step is mapping commute trajectories to the real world road segments so as to generate the road segments transfer matrix, which is the key point for multidays data fusion task. And the second step is generating the commute paths in form of continuous road segments trajectories based on the transfer matrix.

4.1. Trajectory Extraction. To recover individual's commute route which specifically refers to the path between home and work place, we firstly extract individual's history location trajectories from raw mobile phone dataset. As is mentioned above, each location point in the trajectories represents one cell tower id which can be transferred to its corresponding coordinate (longitude, latitude). Let seq_i denote a sequence of records of user i in one day such as $seq_i = \{l_1^i, l_2^i, \ldots, l_n^i\}$, where l_k^i is the kth location of user i. This is the raw trajectory that we can easily obtain from the raw dataset. Then the module aims to transfer $seq_i = \{l_1^i, l_2^i, \ldots, l_n^i\}$ *to* $seq_i = \{g_1^i, g_2^i, \ldots, g_n^i\}$, where g_k^i represents the kth group of several nearby locations which is generated by the clustering step and g_1^i represents the location of home and g_n^i represents the location of work place which are detected by the important location detection module.

4.1.1. Leader Clustering. Figure 4 shows one person's history locations in multiple days. Each yellow point represents one cell tower and the size of the point is proportional to the number of records recorded by the corresponding cell tower, which means the bigger the point is, the more possible one person will appear at that area. In the real world, a motionless mobile phone device may contact with different cell towers at different time and the cell tower reselection often happens where cell towers' coverage overlaps with others, so two nearby points in the map may be generated by users in the one single place and should be clustered in one single group.

Input: all the points of one person, denote as P
Output: all the groups of one person, denote as G;
(1) **while** P.size! = 0 **do**
(2) P_{leader} = SelectLeader(P)
(3) P.remove(P_{leader})
(4) *nearbyPoints* = selectNearbyPoints(P)
(5) $group_i$.addAll(*nearbyPoints*)
(6) P.removeAll(*nearbyPoints*)
(7) **end while**

Algorithm 1: Leader clustering algorithm.

We apply leader clustering algorithm [34] to cluster nearby cell towers into corresponding groups to handle the cell tower reselection problem. The reason we choose leader clustering algorithm is that it does not require the clusters' number before clustering but needs a weight for each point so as to pay more attention to the leader points, which is exactly suitable for this problem. As is mentioned above, each point has one value which represents the number of records recorded by the corresponding cell tower and we use this value as the weight of each point. The time complexity of leader clustering algorithm is $O(N_p \times k)$, where N_p represents the number of all points and k refers to the total number of clusters and the space complexity is $O(N_p)$.

As is shown in Algorithm 1, we firstly select one leader among all the points that have not been clustered based on the weight of each point; then we put all the nearby points which are within the distance of 300 meters from the leader into one group. We keep doing this procedure until all the points have been grouped. Figure 4 shows all the groups of one person's history locations, each member of the group is connected by lines, and the leader of the group is covered by a blue circle.

Figure 4: Cell groups after leader clustering.

Choosing a suitable radius in the clustering process is necessary. But the difficulty is that different areas have different number of cell towers; for example, urban area contains more towers than the suburban area. The average distance between each pair of cell towers is less than 1 km. We tried a range of radius to do the clustering and found that 300 m performs well in our experiment.

4.1.2. Important Location Detection. To extract the cell group trajectory from home to work, we firstly need to find out where home and workplace is. Intuitively we believe that people will stay at home and workplace much longer than other places, so there should be more records recorded around these important places. After the clustering, we treat the group of cell towers as the smallest unit representing individual's history locations. To find out people's home and workplace, we adopt a simple but useful strategy. As mentioned by Isaacman et al. [12], most of people spend the leisure time between 7 p.m. and 6 a.m. at home and spend the work time between 1 p.m. and 5 p.m. at workplaces. So we calculate R_i for each group which represents the total number of records recorded by all the cell towers in the group during specific period. Home is detected as follows:

$$home = \{group_i \mid \max(R_i) \cap t \subset HomeTime\}. \quad (1)$$

Similarly, workplace is selected as follows:

$$work = \{group_i \mid \max(R_i) \cap t \subset WorkTime\}. \quad (2)$$

4.1.3. Commute Trajectory Extraction. Through clustering step, we can transfer individual's everyday sequence $seq_i = \{l_1^i, l_2^i, \ldots, l_n^i\}$ *to* $seq_i = \{g_1^i, g_2^i, \ldots, g_n^i\}$, where l_i refers to locations of cell towers and g_i refers to groups of cell towers. Based on the important location detection step, we can obtain the id of groups which are around home or workplace; then we capture the trajectory between home and workplace.

We used four weeks' data in our experiment, and most of the people have 20 trajectories from home to workplace during the four weeks' period. As is shown in Figure 5, each colored line represents one trajectory from home to workplace.

After extracting all the trajectories, we form the group transfer matrix $M_{G\times G}$ based on these trajectories. Each element M_{ij} in the matrix records the frequency for the ith group transfer to the jth group in all the trajectories. Matrix $M_{G\times G}$ is the key point for the multiday data fusion task, aiming to increase the sampling rate of the commute path.

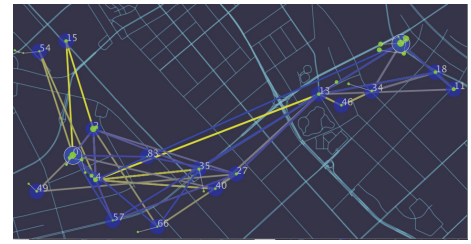


Figure 5: Commute trajectories in multiple days.

4.2. Road Network Formation. Road network dataset is used to transfer cell groups trajectories to the real world road segment trajectories. Road network formation module reconstructs raw road network data with a better modality. One road segment in the raw dataset is stored as a series of points. For example, road segment $R_i = \{p_1, p_2, \ldots, p_n\}$, where p_i refers to the ith point of the road segment.

In this subsection, we introduce two operations for the road network dataset, road segmentation, and road merging, which can increase the precision of the generated path and reduce the time complexity of the algorithm.

4.2.1. Road Segmentation. As is shown in Figure 6, there are some extremely long road segments that have multiple common points with other road segments. This will decrease the precision of map matching procedure because we treat each road segment as the smallest unit. Road segmentation step aims to divide the long road segment into several short segments which contain no common points with other road segments inside the road. For example, we divide road segment $R_i = \{p_1, p_2, \ldots, p_k, \ldots, p_n\}$ into two parts: $R_i^1 = \{p_1, p_2, \ldots, p_k\}$ and $R_i^2 = \{p_k, \ldots, p_n\}$, where p_k is an intersection point between R_i and other segments.

Segmentation procedure contains two steps. The first step is finding out all the common points between each pair of road segments. The second step is dividing long road segments into several short segments based on the common points. After this step, each road segment contains no exit point between the start and end points.

4.2.2. Road Merging. Figure 7 shows another flaw of the raw dataset. As we can see, each road segment has at least one common point with other segments. The problem is that there are some really short road segments that can totally be attached to their adjacent longer road segments. This can reduce the total number of road segments and the time complexity of this step. The problem actually lies in the raw dataset; road segmentation step cannot cause this problem.

To deal with this, we apply an algorithm which firstly finds the all the common points that exactly belong to two road segments. Then it merges the corresponding road segments

Figure 6: Long road segment.

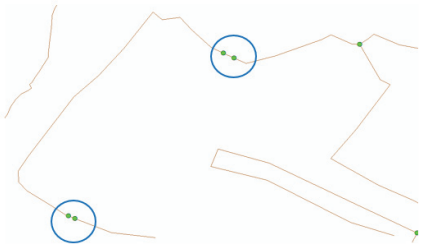


Figure 7: Redundant road segment.

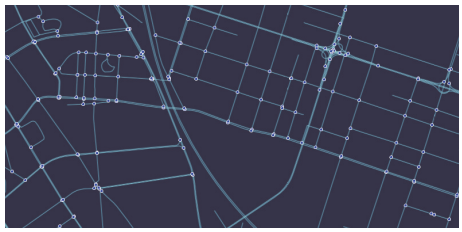


Figure 8: Road network after formation.

based on the common points. Algorithm 2 shows the detailed procedure of road merging algorithm. And Figure 8 shows the road network after the formatting step. The time complexity of the algorithm is $O(N_r)$, where N_r represents the number of road segments and the space complexity is $O(N_p)$, where N_p refers to the number of sampling points consisting of all the road segments.

4.3. Map Matching. In this subsection, we will introduce the procedure of map matching module which includes two parts: data fusion and path generation. The map matching module aims to fuse extracted cell group trajectories with real world road network so as to recover individual's commute route. Data fusion step calculates the road segment transfer matrix $M_{G\times G}$; then path generating step recovers the most likely commute route based on the matrix $M_{G\times G}$.

4.3.1. Data Fusion. This step aims to map each group in the commute cell group trajectories to the corresponding road segments and then form the road segments transfer matrix $M_{R\times R}$ based on the frequency recorded in matrix $M_{G\times G}$. Given matrix $M_{G\times G}$, Algorithm 3 shows the detailed procedure of the data fusion algorithm.

Input: raw road network R_{raw}
Output: format road network R_{format};
(1) cp = findCommonPoints();
(2) $pTwo$ = CalculatiPointsOnlyBelongToTwoRoads(cp);
(3) **for** *point* in $pTwo$ **do**
(4) mergeRoad(*point*);
(5) **end for**

Algorithm 2: Road merging algorithm.

Input: cell group transfer matrix: $M_{G\times G}$
Output: road segment transfer matrix: $M_{R\times R}$;
(1) **for** m_{ij} in $M_{G\times G}$ **do**
(2) $pointSet_i$,$pointSet_j$ = GetAllMembers(i, j);
(3) **for** p_m in $pointSet_i$ **do**
(4) **for** p_n in $pointSet_j$ **do**
(5) $weight = m_{ij}\times$ Weight(p_m) $\times$ Weight(p_n);
(6) $road_a$,$road_b$ = GetNearestRoad(p_m, p_n);
(7) $M_{R\times R}[road_a][road_b]$ += $weight$;
(8) **end for**
(9) **end for**
(10) **end for**
(11) $M_{R\times R}$ = MakingContinuousByDijkstra($M_{R\times R}$);

Algorithm 3: Map matching algorithm.

The transfer frequencies of all pairs of cell groups are recorded in the matrix $M_{G\times G}$. One cell group contains multiple location points; we select the road segment set for each group which includes all the nearest road segments for each point in the group. Then the transfer frequency of two groups is distributed to all pairs of road segments from one set to another according to the weight of the corresponding points in the group.

Through fusing multiday data, we hugely increased the number of location records around individual's commute route. But there still exit some adjacent recorded points whose corresponding road segments are not contiguous with each other. This can cause the discontinuity of the path. To solve this problem, we adopt Dijkstra algorithm to complete each pair of road segments with the shortest path between them. Because as the number of records is increasing, the average distance between two adjacent recorded points will become much smaller. And in most of the cases, people will choose the shortest path when passing two enough close road segments.

The complexity of map matching algorithm is acceptable. Let N_g represent the number of cell groups for one person. Most of the people passed less than 100 cell groups. Let n_p represent the average size of all groups. And the value of n_p is less than 10. Then the time complexity of the algorithm is $O(N_g \times n_p^2)$, and the space complexity is $O(N_r^2)$, where N_r refers to the number of all the road segments contained in the road segment set mentioned above which is quite a little part of the whole road network.

Input: road to road transfer matrix: $M_{R\times R}$
Output: the most possible path: $P = \{r_1^i, r_2^i, \ldots, r_n^i\}$;
(1) $maxFrequence$ = findMaxFrequence($M_{R\times R}$);
(2) **for** m_{ij} in $M_{R\times R}$ **do**
(3) if(m_{ij}==0)
(4) m_{ij} = Double.Max_Value;
(5) else
(6) $m_{ij} = (maxFrequence - m_{ij})/m_{ij}$;
(7) **end for**
(8) P = Dijkstra($M_{R\times R}$, R_{begin}, R_{end})

Algorithm 4: Path generating algorithm.

4.3.2. Path Generation. The path generating module is the final part of the whole model, which generates the most likely commute route for each person. The final commute route is a trajectory of continuous road segments. Given road segments transfer matrix $M_{R\times R}$, Algorithm 4 shows the detailed procedure of route recovering algorithm.

A path's frequency is equal to the sum of all its contained road segments' frequency. And commonly we believe that the path with the highest frequency may be the most possible commute route, but the problem is that the longest path will definitely have the highest frequency among all, and the most possible path should be a relatively shorter one. To balance the length and the total frequency of the path, we recalculate the value of each element m_{ij} in matrix $M_{R\times R}$ according to the following formula:

$$m_{i,j} = \frac{maxValue - m_{i,j}}{m_{i,j}}, \tag{3}$$

where m_{ij} represents the element of the road transfer matrix and $maxValue$ represents the max value in the matrix. So, for example, the road segment with the highest frequency will get zero for the new value and contrarily the road segment with lower frequency will get a higher value. Plenty of formulas have been tried and we found that the formula above performs well in our experiment. Finally, we adopt Dijkstra algorithm to generate the commute path. The time complexity of this algorithm is $O(N_r^2)$.

The entire process of path recovering for one individual takes 1083 milliseconds and uses nearly 1 G of memory. One thing that needs to be noted is that finding the shortest path between two road segments usually costs quite a lot of time. To decrease the time consumption, we calculated all the shortest paths between each pair of road segments based on the Dijkstra algorithm and recorded the data into files. And this preprocessing step can hugely decrease the time consumption of the whole method.

5. Experiment and Result

In this section, we introduce the ways we used to evaluate the proposed method and results of the experiments.

5.1. Experimental Setup. Because the mobile phone data used in our experiment are all anomalous, we cannot directly obtain individuals real commute route. Another issue is that people may not always pass the same route every day. So to testify to the effectiveness of our method in recovering individual's commute route, we adopt two ways of evaluation: path score evaluation and visual evaluation. Path score evaluation grades each path and compares paths based on their score. Visual evaluation draws all the commute information on the map and compares paths visually.

To the best of our knowledge, there exists no model directly generating individual's commute route based on mobile phone data of multiple days. Then we choose the method which is always used in calculating the commute distance [35] as the baseline compared method. The compared method treats the shortest path from home to work based on the real world road network as the approximate commute path. And we compare the paths generated by the two methods through path score evaluation and visual evaluation. Due to the particularity of the experiment, we randomly choose 10 cases for the study.

5.2. Path Score Evaluation. To evaluate the quality of the path, we use number of nearby cell towers to measure the path's authenticity. Intuitively we believe that the most regular path will pass most of cell towers. For each road segment in the path, we calculate the number of nearby cell towers which are within 300 meters from each road segment. Then we can obtain the total number of nearby cell towers for the whole path and we treat the number as the score of the path. For each path $P = \{r_1, r_2, \ldots, r_n\}$, where r_i represents the ith road segment of the path, we calculate the score of the path based on the following formula:

$$score = \sum_{i=1}^{n} sizeOf\left(C_i\right), \quad C_i \in C, \tag{4}$$

where C_i represents the set of all the cell towers within 300 meters from the road segment r_i and C refers to all the cell towers contained in one person's history locations.

We randomly extract 10 people's data and generate the paths by the proposed method and compared method separately. Then we calculate the score of each path. The result is shown in Figure 9; from that we can see that all the paths generated by our method perform better than the compared paths except two samples which have the same score for the two paths.

5.3. Temporality Analysis. We fuse the data generated in multiple days to increase the sampling rate of location points around the commute route. To test the effectiveness of data fusion module, we use the data generated in different number of days to recover the commute route by the proposed model; then we evaluate each path by the path score evaluation. The selected periods include one day, one week, two weeks, three weeks, and four weeks. For the one day's data, we actually choose the day that performs best among all the days. As is shown in Figure 11, the score of recovered path increases as the period getting longer. And we can see that the score based

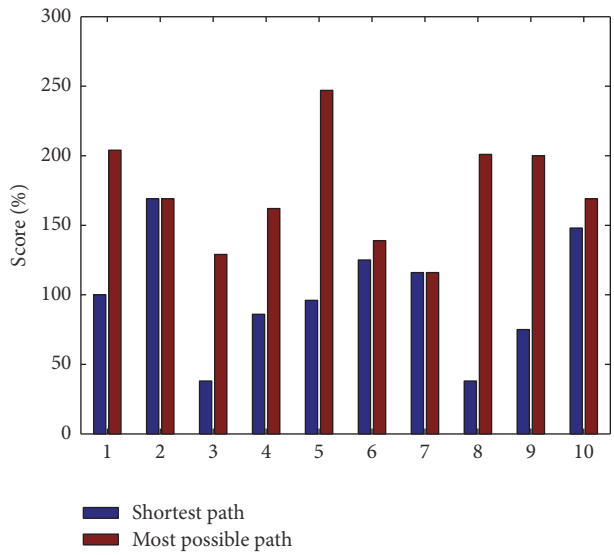
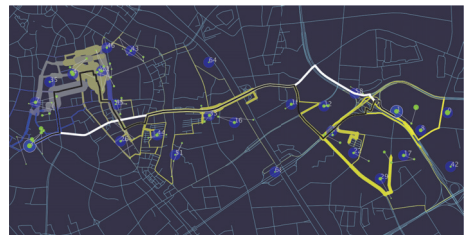


Figure 9: Performance of path score for the two methods.

(a) Commute cell group trajectories of Person A in multiple days



(b) Frequency of each road segment of Person A and his/her shortest path (white line) and most possible path (black line)

(c) Commute cell group trajectories of Person B in multiple days

(d) Frequency of each road segment of Person B and his/her shortest path (white line) and most possible path (black line)

Figure 10: All the commute information of two individuals.

on one week's data has huge improvement than the one based on one day's data and the score tends to stable after the period increases to two weeks.

5.4. Visual Evaluation

5.4.1. Visualization. For better understanding individual's commute route, we draw all the commute information on the map. As mentioned above, people are not always passing the same route every day. For example, people in Beijing are not allowed to drive their own cars during some special days, so they have to take the bus or subway to get to work instead and this can cause the different commute routes for one person. We randomly choose two people's commute information and draw them on the map. We design two kinds of figures for the visualization: one contains the basic information and another contains all the generated routes.

Figures 10(a) and 10(c) show all the cell towers one person passed, all the groups generated by the leader cluster algorithm and commute paths in each day. Each yellow dot represents one specific cell tower and the larger dot means that the corresponding tower recorded much more records for the person. The lines in different color represent the commute route in different days. Finally, the blue circle

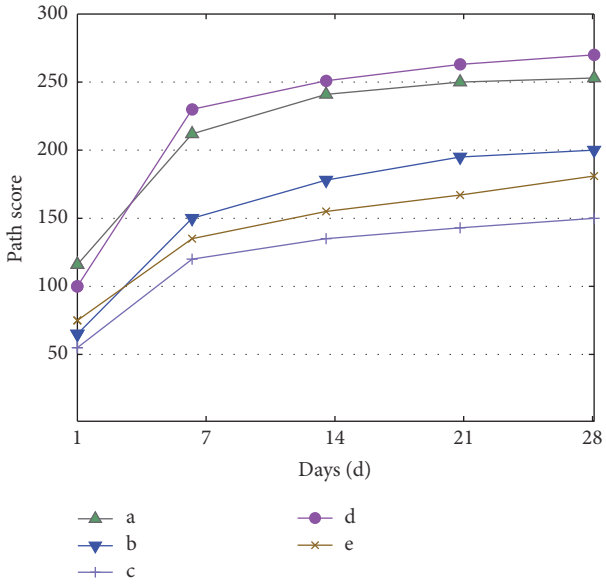


Figure 11: Choose shortest path as commute path.

represents the center of the group and all the members who belong to this group are linked together by the fine yellow lines.

Figures 10(b) and 10(d) include all the road segments one person passed every day, the shortest path from home to work and the path with highest frequency which is generated by our method. The width of each road segment is proportional to its frequency, and the more times one person passes the road segment, the higher frequency the road segment obtains. Besides, we calculate the average time when people passed the road segment and use different colors to represent different period of time. As we can see in Figure 10, the brighter color the road segment has, the earlier time the person will pass the road at. Besides, there are two long paths from home to work: the white one is generated by the baseline method which is exactly the shortest path and the black one is the path generated by our method.

5.4.2. Visual Analysing. Figures 10(a) and 10(c) show all the commute paths of the two people in different days. From these lines we can basically see the main commute path. Besides the main path, there are some paths that are quite different from the main path, which verify supposing that people are not always passing the same route every day. Figures 10(b) and 10(d) draw the road segments with different widths and colors which can better reveal the whole commute information of the people. As we can see from the figure, the black path is surrounded by much more cell towers than the white path and the road segments included in the black path have much higher frequencies than the white path. Then we can conclude that the black path which is generated by our method is much closer to the real commute path than the compared path.

As is shown in Figure 12, there are few situations that people did actually choose the shortest path as the common commute path. That is not common because the shortest path may not be the fastest path when considering the status of the roads.

Figure 12: Choose shortest path as commute path.

6. Conclusion and Discussions

In this paper, we propose a commute route recovering model to recover individual's commute route based on passively generated mobile phone data. The proposed model contains two main modules to deal with different tasks. The data preprocessing module applies leader clustering algorithm to deal with the challenge of low precision of location information. The map matching module calculates the transfer frequency of all related road segments by fusing multidays' commute paths with road network to deal with the challenge of low sampling rate of signal records and multiple overlapping paths. The model generates the path with the highest possibility as the commute path. We adopt two ways to evaluate the result. Experiments show better performance of our model than the compared method.

To the best of our knowledge, our work is the first to explore recovering people's commute route based on the mobile phone data in multiple days. So inevitably the proposed model may have several limitations. For example, the model is sensitive to the quality of the real world road network and that will determine the precision of the generated path to a certain extent. Besides, how to generate the more authentic paths from the transfer matrix in a better way is the aspect we will keep exploring.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 61572059), the National Natural Science Foundation of China (no. 51408018), the State Key Program of National Natural Science of China

(Grant no. 71531001), the State's Key Project of Research and Development Plan (2016YFC1000307), and the Program of Shenzhen (JCYJ20150624154400509).

References

[1] D. Soper, "Is human mobility tracking a good idea?" *Communications of the ACM*, vol. 55, no. 4, pp. 35–37, 2012.

[2] J. Tang, F. Liu, Y. Wang, and H. Wang, "Uncovering urban human mobility from large scale taxi GPS data," *Physica A: Statistical Mechanics and Its Applications*, vol. 438, pp. 140–153, 2015.

[3] M. Zilske and N. Kai, "A simulation-based approach for constructing all-day travel chains from mobile phone data," *Procedia Computer Science*, vol. 52, no. 1, pp. 468–475, 2015.

[4] C. Chen, L. Bian, and J. Ma, "From traces to trajectories: how well can we guess activity locations from mobile phone traces?" *Transportation Research Part C: Emerging Technologies*, vol. 46, pp. 326–337, 2014.

[5] R. Ahas, S. Silm, O. Järv, E. Saluveer, and M. Tiru, "Using mobile positioning data to model locations meaningful to users of mobile phones," *Journal of Urban Technology*, vol. 17, no. 1, pp. 3–27, 2010.

[6] N. E. Williams, T. A. Thomas, M. Dunbar, N. Eagle, and A. Dobra, "Measures of human mobility using mobile phone records enhanced with GIS data," *PLOS ONE*, vol. 10, no. 7, Article ID e0133630, 2015.

[7] J. Blumenstock, G. Cadamuro, and R. On, "Predicting poverty and wealth from mobile phone metadata," *Science*, vol. 350, no. 6264, pp. 1073–1076, 2015.

[8] J.-P. Onnela and A.-L. Barabsi, "Structure and tie strenghs in mobile communication networks," *Proceedings of the National Academy of Sciences*, vol. 104, no. 18, pp. 7332–7336, 2007.

[9] X. Pan, W. Chen, L. Wu, C. Piao, and Z. Hu, "Protecting personalized privacy against sensitivity homogeneity attacks over road networks in mobile services," *Frontiers of Computer Science*, vol. 10, no. 2, pp. 370–386, 2016.

[10] B. C. Csáji, A. Browet, V. A. Traag et al., "Exploring the mobility of mobile phone users," *Physica A: Statistical Mechanics and Its Applications*, vol. 392, no. 6, pp. 1459–1473, 2013.

[11] F. Calabrese, F. C. Pereira, G. D. Lorenzo et al., "The geography of taste: analyzing cell-phone mobility and social events," in *Proceedings of the International Conference on Pervasive Computing*, pp. 22–37, Springer, 2010.

[12] S. Isaacman, R. Becker, R. Cáceres et al., "Identifying important places in peoples lives from cellular network data," in *Pervasive Computing: 9th International Conference, Pervasive 2011, San Francisco, USA, June 12–15, 2011. Proceedings*, vol. 6696 of *Lecture Notes in Computer Science*, pp. 133–151, Springer, Berlin, Germany, 2011.

[13] Y. Zheng, "Methodologies for cross-domain data fusion: an overview," *IEEE Transactions on Big Data*, vol. 1, no. 1, pp. 16–34, 2015.

[14] Y. Zheng, Y. Liu, J. Yuan, and X. Xie, "Urban computing with taxicabs," in *Proceedings of the 13th International Conference on Ubiquitous Computing (UbiComp '11)*, pp. 89–98, ACM, Beijing, China, September 2011.

[15] N. J. Yuan, Y. Zheng, and X. Xie, "Segmentation of urban areas using road networks," Microsoft Technical Report, 2012.

[16] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[17] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, "Exploiting multi-channels deep convolutional neural networks for multivariate time series classification," *Frontiers of Computer Science*, vol. 10, no. 1, pp. 96–112, 2016.

[18] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning (ICML '11)*, pp. 689–696, Bellevue, Wash, USA, July 2011.

[19] Q. Zhang, L. T. Yang, and Z. Chen, "Deep computation model for unsupervised feature learning on big data," *IEEE Transactions on Services Computing*, vol. 9, no. 1, pp. 161–171, 2016.

[20] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351–1362, 2016.

[21] Q. Zhang, H. Zhong, L. T. Yang, Z. Chen, and F. Bu, "PPHOCFS: privacy preserving high-order CFS algorithm on the cloud for clustering multimedia data," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 12, no. 4, article 66, 2016.

[22] J. Xin, Z. Cui, P. Zhao, and T. He, "Active transfer learning of matching query results across multiple sources," *Frontiers of Computer Science*, vol. 9, no. 4, pp. 595–607, 2015.

[23] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with cotraining," in *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT '98)*, pp. 92–100, ACM, Madison, Wis, USA, 2000.

[24] Q. Wang, B. Wang, X. Hao et al., "Face recognition by decision fusion of two-dimensional linear discriminant analysis and local binary pattern," *Frontiers of Computer Science*, vol. 10, no. 6, pp. 1118–1129, 2016.

[25] Y. Zhang, J. Zhang, Z. Pan, and D. Zhang, "Multi-view dimensionality reduction via canonical random correlation analysis," *Frontiers of Computer Science*, vol. 10, no. 5, pp. 856–869, 2016.

[26] Q. Zhang, L. T. Yang, Z. Chen, and F. Xia, "A high-order possibilistic-means algorithm for clustering incomplete multimedia data," *IEEE Systems Journal*, pp. 1–10, 2015.

[27] Q. Zhang and Z. Chen, "A weighted kernel possibilistic c-means algorithm based on cloud computing for clustering big data," *International Journal of Communication Systems*, vol. 27, no. 9, pp. 1378–1391, 2014.

[28] W. Rong, B. Peng, Y. Ouyang, K. Liu, and Z. Xiong, "Collaborative personal profiling for web service ranking and recommendation," *Information Systems Frontiers*, vol. 17, no. 6, pp. 1265–1282, 2015.

[29] J. S. Greenfeld, "Matching GPS observations to locations on a digital map," in *Proceedings of the Transportation Research Board Meeting*, National Research Council (US), Washington, DC, USA, 2002.

[30] S. Brakatsoulas, D. Pfoser, R. Salas et al., "On map-matching vehicle tracking data," in *Proceedings of the 31st International Conference on Very Large Data Bases*, Trondheim, Norway, September 2005, https://www.researchgate.net/publication/221310236_On_Map-Matching_Vehicle_Tracking_Data.

[31] Y. Zheng, "Trajectory data mining: an overview," *ACM Transactions on Intelligent Systems and Technology*, vol. 6, no. 3, article no. 29, 2015.

[32] J. Yuan, Y. Zheng, C. Zhang, X. Xie, and G.-Z. Sun, "An Interactive-Voting based Map Matching algorithm," in *Proceedings of the 11th IEEE International Conference on Mobile Data*

Management (MDM '10), pp. 43–52, IEEE, Kansas City, Mo, USA, May 2010.

[33] A. Thiagarajan, L. Ravindranath, H. Balakrishnan et al., *Accurate, Lowenergy Trajectory Mapping for Mobile Devices*, Networked Systems Design and Implementation, 2011.

[34] Q. Wu, X. Qi, E. Fuller, and C.-Q. Zhang, ""Follow the leader": a centrality guided clustering and its application to social network analysis," *The Scientific World Journal*, vol. 2013, Article ID 368568, 9 pages, 2013.

[35] P. Yang, T. Zhu, X. Wan, and X. Wang, "Identifying significant places using multi-day call detail records," in *Proceedings of the 26th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '14)*, pp. 360–366, IEEE, Limassol, Cyprus, November 2014.