Self-adaptive Trajectory Prediction Method Based on Density Clustering

Hongfang ZHOU*, Jie GUO, Xinyi WANG, Guorong ZHANG, Xuehan ZHAO

School of Computer Science and Engineering, Xi’an University of Technology, Xi’an 710048, China

Abstract

To solve the problem of user trajectory prediction in the mobile communication environment, we proposed a self-adaptive trajectory prediction method (ATPDC) based on density clustering. And it consists of two stages which are trajectory modeling stage and trajectory updating stage respectively. In the first stage, it constructs the user trajectory prediction model by clustering historical trajectory. And in the second stage, it enhances the model built on the former stage. We test it on the MR records in the mobile communication environments. Experimental results show that ATPDC algorithm can achieve the incremental updating with satisfactory prediction accuracy and prediction efficiency with the growth of user trajectory data. Furthermore, it is also suggested that the mobile MR road test reports contain potential user behavior patterns and could be used to analyze and mine users’ behaviors.

Keywords: Trajectory Prediction; Density Clustering; Self-adaptive; Trajectory Modeling; Trajectory Updating

1 Introduction

With the wide use of mobile portable devices and rapid development of wireless communication and GPS, people can get large amounts of real-time user location data at low costs. And by analyzing such information, we can obtain real-time longitude/latitude coordinates and driving direction of the user. The location points relative to time are connected to form user’s moving trajectories over a period of time [1]. There exist abundant spatial structure information and user behavior rules in large amounts of user location data and moving trajectories. Via analyzing and mining them, we can obtain a variety of value-added services and tools for users, such as electronic maps and route guidance service of intelligent traffic, friend recommendations and personalized services in social networks, etc [2,3].

In recent years, LBS (Location Based Service) has been increasingly concerned by many scholars. Trajectory prediction technology is one of the most popular issues. Existing researches mainly focus on mining continuous trajectory of moving objects, e.g, GPS. However, few researchers have mined users’ behavior patterns in the mobile communication environments. Generally, user data generated in the mobile communication environments is characterized by huge amounts, complex

*Corresponding author.
Email address: zhouhf@xaut.edu.cn; 99582540@qq.com (Hongfang ZHOU).

1553–9105 / Copyright © 2015 Binary Information Press
DOI: 10.12733/jcis13693
April 1, 2015
data format, discrete distribution, dynamic changes, and so on. In order to solve the problem of user trajectory prediction in the mobile communication environments, we take mobile communication road test reports called MR (Message Report) as data source and propose a self-adaptive trajectory prediction method based on density clustering in the paper.

2 Research Status

In the study of user behavior pattern mining, the trajectory prediction technology has gained fruitful research results. Through analyzing moving trajectory data of 100,000 users, Gonzalez et al. find that users’ activity area possesses certain regularity in time and space [4]. By using mobile phone bills of each user during three months, and measuring the entropy of user trajectory, it comes to the conclusion that the prediction accuracy of human behavior patterns must be [80%, 93%] [5]. This work not only shows that user moving trajectory has certain regularity and predictability, but also lays a theoretical foundation for mining user behavior patterns. In order to build multivariate mixed model of the trajectory, Gaffne et al. estimate the probability of user trajectory belonging to the model, and use EM algorithm [6] to evaluate parameters of the mixed model. By means of this method, we can only obtain global user behavior model. Lee et al. investigate the problem of user sub-trajectory firstly and present TRACLUS algorithm to discover user sub-trajectories [7]. Lee et al. believe that sub-trajectories can reflect the users’ specific interests. TRACLUS algorithm is based on the idea of partition-and-group. Concretely, a complete user trajectory is partitioned into a set of trajectory segments firstly. And then, similar trajectory segments are merged into clusters to form user sub-trajectories. Subsequently, the ideas of sub-trajectory and partition-and-group are applied to solve the problems of trajectory classification and outlier detection. A trajectory feature generation framework called as TraClass [8] and a trajectory outlier detection algorithm [9] are proposed. Based on these, Agrawal et al. give the TCMM framework for incremental clustering of trajectory data, which contains two stages—Micro-clustering and Macro-clustering respectively [10]. During Micro-clustering stage, its main task is dividing new user trajectory data into existing trajectory clusters to form micro-clusters based on some predefined similarity measures. During Macro-clustering stage, micro-clusters generated in the previous stage are merged and clustered again. And these operations are executed only when a user wants to query current trajectory clustering result.

Above studies are limited to moving trajectory prediction, and they are limited to the problem of the mining user mobile patterns to some degree in some fields. However, realizing these methods depends on the complete, continuous and static user trajectory data to a large extent, which makes them unsuitable in the following situations.

• In reality, user moving trajectory usually can be expressed as an incremental data flow over a period of time, and trajectory database is a constantly changing and accumulating data set. Particularly, the discreteness and indeterminacy of user location information make user behavior model changing with time dynamically in the mobile communication environments.

• Mobile user data is usually characterized by huge amounts, complex data format and discrete distribution, which makes these data difficult to apply existing method directly on continuous trajectory data in the mobile communication environments.

• When new moving trajectory data arrives, the trajectory prediction algorithm can dynamically update original user trajectory prediction model, and avoid to train a new one again.
In the presence of three problems above, we propose a self-adaptive trajectory prediction method (ATPDC) based on density clustering in the paper. The method contains two stages: trajectory modeling stage and trajectory updating stage. In the trajectory modeling stage, trajectory clusters are generated by clustering new moving data for user using density-based clustering algorithm. Then the similarities between new and old trajectory clusters are calculated. Next, old and new trajectory clusters with the maximum similarity and their similarity is not higher than the giving threshold are merged. The new trajectory clusters which cannot be merged will be regard as emerging trajectory patterns and join them to the set of user trajectory clusters. In the trajectory updating stage, it completes two tasks of updating user global trajectory clusters and deleting abnormal trajectory points.

3 Problem Description

3.1 User moving data

Now, we can obtain user location information by various positioning ways, such as GPS, Wi-Fi and GSM. Among them, the coverage of Wi-Fi is smaller. Therefore, such data are not suitable for the analysis of outdoor moving trajectories. Personal GPS trajectory data are not generally public, and users can not keep GPS service activating on mobile devices due to higher consuming on power of GPS devices. So, it is difficult to use GPS services to get complete user trajectory information. In the GSM standards, telephones send measurement reports regularly about wireless environment called MR (Message Report) to the switch. Using MR data generated by user’s cell phones and latitude/longitude information recorded in GSM network database, we can get the real-time location information, and then obtain the complete movement trajectories of users. Liu et al. make some achievements on user moving trajectories modeled by using base station information from user bill data [11]. But errors between user real location and geographical position indicated by the base stations are larger, which affects the accuracy of trajectory prediction. Compared with user bill data, MR not only provides the information about main service cell of phone, but also provides its neighboring cells information. Using this kind of data, we can generate more accurate location information. Positioning technology and MR data introduced in reference [13] are used to estimate users’ location, and the ultimate positioning accuracy can be up to 50m. This paper uses the positioned MR data to develop moving trajectory prediction.

3.2 Related concepts

**Definition 1** (Moving Report) Moving Report refers to the user geographic location which includes longitude and latitude information at a certain moment. A moving report is denoted as \( R(\text{Imsi}, \text{Timestamp}, \text{Lon}, \text{Lat}) \), where \( \text{Imsi} \) (International Mobile Subscriber Identification Number) is user ID which identifies user’s identity uniquely, \( \text{Timestamp} \) is the time when report is generated, \( \text{Lon} \) is the geographical longitude information of the user \( \text{Imsi} \) at time \( \text{Timestamp} \), and \( \text{Lat} \) is the geographical latitude information of the user \( \text{Imsi} \) at time \( \text{Timestamp} \).

**Definition 2** (Grid Cell) The geographical area where moving report is generated is divided into uniform, close adjacent grid arrays, and each cell of grid arrays is defined as a Grid Cell. Location
coordinate of a cell is indicated as $\langle \text{grid}_x, \text{grid}_y \rangle$, where $\text{grid}_x$ is the row index of the cell, and $\text{grid}_y$ is the column index of the cell.

**Definition 3** (Moving Point) Moving Point is defined as a point which is represented by position coordinate $D(\text{grid}_x, \text{grid}_y, m)$ of grid cell associated with original moving report $R(\text{Imsi}, \text{Time-stamp}, \text{Lon}, \text{Lat})$ for user Imsi at time Timestamp. Here, $\langle \text{grid}_x, \text{grid}_y \rangle$ is the grid coordinate of geographical point $\langle \text{Lon}, \text{Lat} \rangle$, and $m$ is the number of moving reports contained in grid cell $\langle \text{grid}_x, \text{grid}_y \rangle$ at time Timestamp.

**Definition 4** (Trajectory Point and Influence Area) Moving points within $[t_i, t_j]$ are clustered to form trajectory clusters, and these clusters are denoted as Eq. (1).

$$C_{mov} = \{ < t_1, \text{grid}_x_1, \text{grid}_y_1, m_1 >, ..., < t_n, \text{grid}_x_n, \text{grid}_y_n, m_n > \} (t_i \leq t_1 ... t_n \leq t_j)$$ (1)

Trajectory Point is the weighted average of all moving points from the trajectory cluster, and there is one-to-one relationship between trajectory point and trajectory cluster. Influence area refers to the round area of trajectory point as the center trajectory point and its influence area is denoted as $O(\langle \text{grid}_x, \text{grid}_y, m, k \rangle)$, where $\langle \text{grid}_x, \text{grid}_y \rangle$ is the grid cell coordinate of the trajectory point being in, calculated as Eq. (2), $m$ is the number of moving reports contained in the corresponding trajectory point, and its value is the sum of moving reports which belong to all moving points in $C_{mov}$, $k$ is the effective radius of influence area of the trajectory point, calculated as Eq. (3).

$$\text{grid}_x = \sum_{k=1}^{n} \text{grid}_x_k \times \frac{m_k}{\sum_{r=1}^{n} m_r}, \text{grid}_y = \sum_{k=1}^{n} \text{grid}_y_k \times \frac{m_k}{\sum_{r=1}^{n} m_r}$$ (2)

Here, $\langle \text{grid}_x_k, \text{grid}_y_k, m_k \rangle (1 \leq k \leq n) \in C_{mov}$.

$$k = \begin{cases} k_{\max}, & \sum_{r=1}^{n} m_r \geq m_{\max} \\ \sum_{r=1}^{n} m_r \times \text{Rate}_k, & m_{\min} < \sum_{r=1}^{n} m_r < m_{\max}; 0 < \text{Rate}_k < 1 \\ k_{\min}, & \sum_{r=1}^{n} m_r \leq m_{\min} \end{cases}$$ (3)

Here, $m_{\max}, m_{\min}$ are upper threshold and lower threshold of the amount of moving reports respectively, and $k_{\max}, k_{\min}$ are upper threshold and lower threshold of influence area of trajectory point respectively.

**Definition 5** (Similarity between Trajectory Points) Similarity between Trajectory Points is used for measuring the similarity between trajectory clusters which these trajectory points belong to. The similarity between $O_1(\langle \text{grid}_x_1, \text{grid}_y_1, m_1, k_1 \rangle)$ and $O_2(\langle \text{grid}_x_2, \text{grid}_y_2, m_2, k_2 \rangle)$ is defined as Eq. (4). And here, $\alpha$ is the side of the grid cell.

$$\text{Sim}(O_1, O_2) = \frac{k_1 + k_2 - \sqrt{((x_1 - x_2) \times \alpha)^2 + ((y_1 - y_2) \times \alpha)^2}}{\min(k_1, k_2) \times 2}$$ (4)
Definition 6 (Prediction Probability of Trajectory Point) Prediction Probability of Trajectory Point refers to the possibility which a user occurs in the trajectory point or its influence area at any time or period. Suppose the set of historical trajectory points for user is denoted as $S_o$ during the period $[t_i,t_j]$. 

$$S_o = \{O_1(grid_{x_1}, grid_{y_1}, m_1, k_1), ..., O_n(grid_{x_n}, grid_{y_n}, m_n, k_n)\} \quad (5)$$

Prediction probability of trajectory point is calculated according to Eq. (6) during this period. Here, $p_{O_i}$ is the prediction probability of the trajectory point $O_i$.

$$p_{O_i} = \frac{m_i}{\sum_{i=j}^{n} m_j} \quad (1 \leq i \leq n) \quad (6)$$

Definition 7 (Moving Trajectory) Moving Trajectory refers to the trajectory point sequences in which trajectory points are sorted in the chronological order. One day of user Imisi are divided into several time segments $\{T_{[t_i,t_j]}, T_{[t_j,t_k]}, ..., T_{[t_n,t_i]}\}$ $(0 \leq t_i < t_j < t_n < 24)$. And the set of trajectory points generated in each time segment is denoted as Eq. (7).

$$S_{Traj} = \{\{O_1^{T_{[t_i,t_j]}}, O_2^{T_{[t_i,t_j]}}, ..., O_p^{T_{[t_i,t_j]}}\}, ..., \{O_1^{T_{[t_n,t_i]}}, O_2^{T_{[t_n,t_i]}}, ..., O_p^{T_{[t_n,t_i]}}\}\} \quad (7)$$

And moving trajectory of the user is denoted as Eq. (8).

$$\{O_1^{T_{[t_i,t_j]}}, O_2^{T_{[t_i,t_j]}}, ..., O_p^{T_{[t_i,t_j]}}\} \rightarrow \{O_1^{T_{[t_n,t_i]}}, O_2^{T_{[t_n,t_i]}}, ..., O_p^{T_{[t_n,t_i]}}\} \quad (8)$$

4 Trajectory Prediction Algorithm

4.1 Trajectory modeling

Different mobile phone users have various habits of using mobile phone, which makes the number of user MR reports collected by the specific devices have bigger differences and the regional shapes covered by MR are various. Density-based clustering method is suitable for finding clusters with arbitrary shapes and different sizes. And furthermore, it can eliminate interference caused by outliers effectively. Therefore, density-based clustering method is used to analyze user MR data in the paper. According to people’s daily routine, MR data in a full day are divided into six periods. They are 0 am to 6 am, 6 am to 9 am, 9 am to 12 am, 12 am to 14 pm, 14 pm to 18 pm and 18 pm to 24 pm. And they represent rest, going to work, working, getting off work, working and entertainment respectively. In this paper, users’ moving points in each time period are modeled separately to get the trajectory points, which can be arranged in some a time order.

In the trajectory modeling stage, moving points can be gotten by rasterizing new generated MR data. Then we cluster moving points in different periods to generate trajectory clusters by means of the method proposed in reference [12]. Each trajectory cluster is considered as a hot-spot area of user in corresponding period, and the center point of each cluster is considered as user’s trajectory point. According to the similarity between new and old trajectory points in the same period, new and old trajectory clusters are merged. And then trajectory points and their influence areas of the merged trajectory cluster should be updated. For the new trajectory clusters which
are not be merged, the corresponding trajectory points are considered as new trajectory points generated in corresponding periods and are added to the set of original moving trajectories.

Trajectory modeling algorithm is described as follows. For convenience, \( TM \) is indicated as moving trajectory model for user, \( Traj_{old} \) is the set of original trajectory clusters, \( Set_{mov} \) is the set of moving points, \( S_{mod} \) is the similarity threshold of trajectory points in trajectory modeling stage, \( t_{del} \) is the largest retention times of the trajectory points which are not removed from the set of trajectory points, \( P_{min} \) is the lower threshold of prediction probability for trajectory point, and \( n_{keep} \) is the retention times of trajectory point.

Algorithm: trajectory modeling algorithm

Input: \( Set_{mov}, Traj_{old} \)

Output: \( Traj_{old}, TM \)

1) Corresponding to six periods described earlier in this section, the set \( Set_{mov} \) of all moving points in a full day is divided into six subsets. Using the density-based clustering algorithm DCURS (Density-based Clustering Using Representative Set) [12], it clusters moving points on each subset to generate new set \( Traj_{new} \) of trajectory clusters, and calculates the corresponding trajectory point and the corresponded influence area.

2) If \( Traj_{old} \) is empty, it calculates prediction probability of trajectory point in each period and builds the moving trajectory prediction model \( TM \). If not, then turn to the Step 3.

3) Merging the corresponding trajectory clusters in \( Traj_{old} \) and \( Traj_{new} \), and updating trajectory points and its influence area. Based on the definition of similarity between trajectory points, it determine the most similar old trajectory cluster with the new trajectory cluster from \( Traj_{old} \). If the similarity is greater than the given threshold \( S_{mod} \), new and old trajectory clusters will be merged. And the new trajectory cluster will be deleted from \( Traj_{new} \).

4) Deleting invalid trajectory clusters and trajectory points from \( Traj_{old} \) in each period. It calculates prediction probability of each trajectory point in \( Traj_{old} \), and then sorts trajectory points with the prediction probability in descending order. Finally, if the number of trajectory points in the period is less than or equal to \( t_{del} \), trajectory points and trajectory clusters ought to be not deleted. And if the number of trajectory points in the period is greater than \( t_{del} \), trajectory points and trajectory clusters which prediction probability is larger should be retained. And trajectory points and trajectory clusters that the prediction probability is less than \( p_{min} \) and retention times is greater than \( n_{keep} \) should be deleted.

5) Adding new trajectory clusters not merged in \( Traj_{new} \) to \( Traj_{old} \), updating corresponding trajectory point and its influence area, and then calculating prediction probability of trajectory point and building the moving prediction model \( TM \).

4.2 Trajectory updating

In the trajectory modeling stage, merging new and old trajectory clusters and adding new trajectory points are executed in original moving trajectory. And these operations change the structures between clusters, which details are shown in Fig. 1. In Fig. 1(a), it shows that the original moving trajectory has two independent trajectory clusters \( C_1 \) and \( C_2 \), and includes two trajectory points \( O_1 \) and \( O_2 \). In Fig. 1(b), it shows that the new trajectory cluster \( C_3 \) and trajectory point \( O_3 \) are generated when new MR data are produced. In Fig. 1(c), it shows that the new trajectory cluster
$C_3$ and old trajectory cluster $C_1$ are merged to generate the new cluster $C_1$. And the clusters $C_1$ and $C_2$ are no longer independent. The distance between moving points in these clusters is enough close on geographical distribution so that the space structure between cluster $C_1$ and $C_2$ is changed. In Fig. 1(d), it shows that the cluster $C_1$ and $C_2$ should be merged to generate a whole trajectory cluster $C_4$, and the new trajectory point $O_4$ and its influence area are calculated again.

![Fig. 1: Structure change between clusters](image)

In order to solve the problem of structure change between clusters, it is required to update again moving trajectory model generated in trajectory modeling stage. Firstly, the algorithm adjusts the structure of trajectory clusters generated in trajectory modeling stage, and also merges and updates the trajectory clusters and trajectory points which the similarity is greater than similarity threshold. To avoid the situation that merging trajectory clusters makes the scale of trajectory clusters unreasonably extended and the trajectory model distort, it is also required to appropriately reduce the sizes of trajectory clusters. In addition, the size of trajectory cluster is determined by the number of moving reports contained in the cluster, so the reduction of the size of trajectory cluster is equivalent to the reduction of the number of moving reports. Suppose the trajectory point $O(grid_x, grid_y, m, k)$ associates with the trajectory cluster $C$, and the reduction factor is $\beta$, the size of trajectory cluster reduced is calculated as Eq. (9).

$$m' = m \ast \beta \tag{9}$$

Trajectory updating algorithm is shown as follows. For convenience, $TM$ is indicated as moving trajectory model which is the output in trajectory modeling stage, $Tra_{old}$ is the set of trajectory clusters which is the output in trajectory modeling stage, $s_{upd}$ is the similarity threshold of trajectory points in trajectory updating stage, and $\beta$ is the reduction factor of trajectory clusters.

Algorithm: Trajectory updating algorithm

Input: $TM$, $Tra_{old}$

Output: $TM$

1) It calculates the similarity between any two trajectory points during each period in $TM$. If the similarity is greater than or equal to $S_{upd}$, trajectory clusters which trajectory points associate with should be merged, and the retention times $n_{keep}$ of each cluster is increased by 1.

2) It reduces the sizes of all trajectory clusters in $Tra_{old}$ by using the reduction factor $\beta$, and updates the coordinate and influence area of corresponding trajectory point.

3) It updates prediction probability of trajectory point during each period in $TM$, and output $TM$. 
4.3 Trajectory predicting

After handling with user’s MR data in trajectory modeling and updating stage, ATPDC algorithm trains moving trajectory model which is used to predict user’s trajectory. The model is composed of six sequences including trajectory points. And it can predict future moving trajectory by prediction probability of these trajectory points. One day of user are divided into several time segments \( \{T_{[t_i,t_j]}, T_{[t_j,t_k]}, \ldots, T_{[t_n,t_i]}\} (0 \leq t_i < t_j < t_n < 24) \). The trajectory model of user \( \text{Imsi} \) is trained as follows:

\[
TM_{\text{Imsi}} = \{ \{O_{1}^{T_{[t_i,t_j]}}, O_{2}^{T_{[t_i,t_j]}}, \ldots, O_{p}^{T_{[t_i,t_j]}}\}, \ldots, \{O_{1}^{T_{[t_n,t_i]}}, O_{2}^{T_{[t_n,t_i]}}, \ldots, O_{r}^{T_{[t_n,t_i]}}\} \}
\]

Where the trajectory point \( O_T^{T} \) is expressed as \( \langle \text{grid}_x^T, \text{grid}_y^T, \text{m}_x^T, \text{k}_x^T, \text{p}_x^T \rangle \), \( p_x^T \) is the prediction probability of trajectory point \( O_T^{T} \). In each period \( T \), the sum of prediction probability is 1, and the trajectory point with biggest prediction probability is selected as prediction result of user trajectory.

5 Experiments

In order to ascertain that the moving trajectory prediction can be well implemented by analyzing the positioned MR data, we verify the validity of ATPDC algorithm from two aspects about the trajectory prediction accuracy and prediction time. It is noteworthy that no one research about trajectory prediction has used MR data characterized by discrete distribution, so the experimental results cannot be compared with other algorithms. We only analyze and verify the feasibility of ATPDC algorithm. All experiments are performed on a PC with Operation System of Windows 7, an i5 CPU, 6G memories and the platform of JDK1.7.

5.1 Experimental data

In this paper, we use MR data which is 100 users’ call records within 14 days in Jan.2013 from a city in China. The data format is shown in Table 1.

<table>
<thead>
<tr>
<th>Imsi</th>
<th>Timestamp</th>
<th>Lon</th>
<th>Lat</th>
</tr>
</thead>
<tbody>
<tr>
<td>4906001734560437</td>
<td>2013/01/14 11:05:44.024</td>
<td>106.7084884644355</td>
<td>26.5759388702175</td>
</tr>
<tr>
<td>4906202204218178</td>
<td>2013/01/15 14:36:45.015</td>
<td>106.703338623047</td>
<td>26.5740965672545</td>
</tr>
<tr>
<td>4906002221232006</td>
<td>2013/01/20 20:53:25.080</td>
<td>106.704711914063</td>
<td>26.5768600105869</td>
</tr>
</tbody>
</table>

There are two problems required to solve in the procedure of modeling user trajectory by using moving reports.

- The location information which the moving report provides tends to reflect the geographic location when user calls, and the moving trajectory of users can not be described completely and really. Therefore, the data format has some limitation to model trajectory.
The mobile phone reports MR data to the special equipment regularly when the user calls and surfs on the Internet, or the base station which the phone associates with makes switch. If all MR reports are used as the input of the algorithm, the efficiency of the algorithm will be influenced seriously. And it is extremely difficult to update and maintain such large MR database. Considering above factors, we compress the geographical location of original MR to grid cell, and the process is called as the rasterization of MR data. Before rasterizing MR data, we divide the geographical area covered by MR data into grid cells with size of 33m*33m. The size of grid cell is confirmed in positioning stage, which is beyond our research scope in the paper. The details can be referred in reference [13]. The detailed procedure of rasterizing MR data is as follows.

Suppose a point $P$(Lon, Lat) coming from the surface of earth which is regarded as a sphere with radius $R$ moves northward $d$ meters along longitude line, then latitude $Lat$ increases $\Delta Lat$. A point $p$ moves eastward $d$ meters along latitude line, longitude $Lon$ increases $\Delta Lon$.

$$\Delta Lat = \frac{180d}{\pi R}, \Delta Lon = \frac{180d}{\pi R \cos \frac{Lon}{180}}$$

In conclusion, the latitude and longitude scopes which each grid cell covers can be gotten by calculation. Then, the location point which each MR report reflects is assigned to the corresponding grid cell, and the grid coordinates are used to replace latitude/longitude coordinates of the point. Finally, we obtain these moving points expressed as grid coordinates. The data format of moving point is shown in Table 2.

**Table 2: Moving points rasterized**

<table>
<thead>
<tr>
<th>Imsi</th>
<th>Timestamp</th>
<th>Grid_X</th>
<th>Grid_Y</th>
<th>MR_Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>4906001734560437</td>
<td>2013011411</td>
<td>874133</td>
<td>431643</td>
<td>74</td>
</tr>
<tr>
<td>4906202204218178</td>
<td>2013011514</td>
<td>874109</td>
<td>431652</td>
<td>10</td>
</tr>
<tr>
<td>4906022221232306</td>
<td>2013012020</td>
<td>874095</td>
<td>431664</td>
<td>152</td>
</tr>
</tbody>
</table>

5.2 Experimental parameters

The experimental parameters in ATPDC algorithm are listed in Table 3. Parameters 1-4 are set by analyzing the attributes of moving point and user’s activity experience. Generally, the coverage of base station is 200m-300m in one city, and it can generate about 30 MR reports in a grid cell covering 100m*100m [13]. According to the grid size (33m*33m) defined earlier in the paper, $\epsilon$ is set to 1.5 and $MinPts$ is set to 9. Due to the reasonable influence area of trajectory point can not be more than the coverage of base station, $k_{max}$ is set to 250m. In addition, the error of location in MR data is less than 50m, which makes $k_{min}$ as 25m. The number of MR reports contained in trajectory cluster closely associates with people’s daily activities. Generally, the average walking speed of an ordinary people is 1.5m/s, and people may be walking, resting, moving or mixed state when he or she calls, therefore we estimate the average speed of the people with 5m/s. According to the threshold interval [25m, 250m] of the influence area of trajectory point and the frequency which mobile phone sends MR reports in the process of calls, the counts of MR reports within trajectory cluster should be not less than 100 and not more than 1000.
Parameters 5-9 need be adjusted according to the actual amount of user data and the demand for prediction results. In the paper, we use MR reports of 100 users during 14 days to predict moving trajectory, and these parameters are set by referring to the user activity habits during one week. It is supposed that the user has no access to the same area again during a week after a user access an area, then the area is considered as the non-hotspot area of the user, and corresponding trajectory point is considered as the abnormal trajectory point which should be deleted in trajectory updating stage. In the stage which algorithm efficiency is verified, \( t_{del} \) is set to 7, \( p_{min} \) is set to 0.1, \( s_{mod} \) is set to be 0.3 and \( s_{upd} \) is set to be 0.5.

<table>
<thead>
<tr>
<th>No</th>
<th>parameter</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \varepsilon )</td>
<td>The radius of the neighborhood of an clustering object</td>
</tr>
<tr>
<td>2</td>
<td>MinPts</td>
<td>The minimum object counts of neighborhood within an core object</td>
</tr>
<tr>
<td>3</td>
<td>( m_{max}/m_{min} )</td>
<td>The upper/lower threshold of MR counts within an trajectory cluster</td>
</tr>
<tr>
<td>4</td>
<td>( k_{max}/k_{min} )</td>
<td>The upper/lower threshold of influence area of an trajectory point</td>
</tr>
<tr>
<td>5</td>
<td>( s_{mod} )</td>
<td>The similarity threshold of trajectory points in trajectory modeling stage</td>
</tr>
<tr>
<td>6</td>
<td>( s_{upd} )</td>
<td>The similarity threshold of trajectory points in trajectory updating stage</td>
</tr>
<tr>
<td>7</td>
<td>( t_{del} )</td>
<td>The largest retention times of the trajectory points which are not removed from the set of trajectory points</td>
</tr>
<tr>
<td>8</td>
<td>( P_{min} )</td>
<td>The lower threshold of prediction probability of an trajectory point</td>
</tr>
<tr>
<td>9</td>
<td>( \beta )</td>
<td>The reduction factor of trajectory clusters</td>
</tr>
</tbody>
</table>

### 5.3 Trajectory prediction accuracy

In the moving trajectory model, the trajectory point which prediction probability is the biggest of all trajectory points in each period is considered as the prediction result of the period. It is supposed that the moving trajectory model predicts the trajectory of the user Imsi as follows.

\[
TM_{IMSI} = \{O_1^{T[t_i,t_j]}, O_2^{T[t_j,t_k]}, \ldots, O_6^{T[t_{n-1},t_i]}\} (0 \leq t_i < t_j < 24) \tag{12}
\]

Known the time \( t \in [t_j,t_k] \) which need to be predicted, and the grid coordinate of actual geographic position for user at time \( t \) is \( W^t < grid_x^t, grid_y^t > \). If the trajectory point \( O_2^{T[t_j,t_k]} (grid_{x_2}^{T[t_j,t_k]}, grid_{y_2}^{T[t_j,t_k]}, m_2^{T[t_j,t_k]}, k_2^{T[t_j,t_k]}, p_2^{T[t_j,t_k]}) \) and the actual position \( W^t \) meet as Eq. (13), the moving trajectory of user Imsi which are predicted by using \( TM \) at time \( t \) is regarded as true. Where \( D_{IMSI}(W^t, O_2^{T[t_j,t_k]}) \) is the distance between the trajectory point \( O_2^{T[t_j,t_k]} \) and the position \( W^t \), \( a \) is the side of grid cell.

\[
D_{IMSI}(W^t, O_2^{T[t_j,t_k]}) = \sqrt{(grid_{x_2}^{T[t_j,t_k]} - grid_x^t)^2 + (grid_{y_2}^{T[t_j,t_k]} - grid_y^t)^2} \tag{13}
\]

Then the prediction accuracy of trajectory model \( TM \) is defined as Eq. (14).
\[
accuracy_{TM}^T = \frac{R_T}{\sum_{i=1}^{n} W_i^T}, \quad accuracy_{TM} = \frac{\sum_{i=1}^{6} R_{T_i}}{\sum_{i=1}^{n} \sum_{i=1}^{6} W_i^T} \tag{14}
\]

In the formula, \(accuracy_{TM}^T\) and \(accuracy_{TM}\) respectively are expressed as the prediction accuracy and the average prediction accuracy of moving trajectory model \(TM\) at time \(T\). \(R_T\) is expressed as the number of trajectory points which are exactly predicted at time \(T\), \(W_i^T\) is expressed as any moving point at time \(T\).

In the experiments, three experiments including non-self-adaptive, half-self-adaptive and self-adaptive are made. In non-self-adaptive experiment, all MR data of an user in the earlier 13 days are considered as the input of ATPDC algorithm to train the trajectory model \(TM_1\). In half-self-adaptive experiment, MR data of an user in the earlier 7 days are used to train trajectory model \(TM_2\), and MR data in the days of 8-th to 13-th are used to update \(TM_2\). In self-adaptive experiment, MR data of a user every day are regarded as the input of the algorithm, which trains and updates trajectory model. That is, the data in 1-th day is used to train the trajectory model \(TM_3\), and then model \(TM_3\) is incrementally updated by using MR data of the next 12 days. The actual position of the user in 14-th day which is recorded in MR data is used to calculate the prediction accuracy of each trajectory model when moving trajectory model \(TM_1\), \(TM_2\) and \(TM_3\) are trained. Then we implement above three experiments for 100 users respectively, and calculate the average prediction accuracy for above three models during six periods and the total average prediction accuracy of each model. Experimental results are shown in Table 4 and Fig. 2.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>0am-6am</th>
<th>6am-9am</th>
<th>9am-12am</th>
<th>12am-14pm</th>
<th>14pm-18pm</th>
<th>18pm-24am</th>
</tr>
</thead>
<tbody>
<tr>
<td>non-self-adaptive</td>
<td>0.95</td>
<td>0.78</td>
<td>0.84</td>
<td>0.71</td>
<td>0.77</td>
<td>0.83</td>
</tr>
<tr>
<td>half-self-adaptive</td>
<td>0.95</td>
<td>0.81</td>
<td>0.87</td>
<td>0.76</td>
<td>0.80</td>
<td>0.82</td>
</tr>
<tr>
<td>self-adaptive</td>
<td>1.00</td>
<td>0.83</td>
<td>0.92</td>
<td>0.82</td>
<td>0.88</td>
<td>0.85</td>
</tr>
</tbody>
</table>

From the Table 4, it can be discovered that the prediction accuracy of different trajectory models during 0am-6am, 9am-12am and 14pm-18pm are higher than that during the rest of periods. The period 0am-6am is the people’s bedtime, which generally makes people’s position to tend changelessly. So this period has higher prediction accuracy. The period 9am-12am and 14pm-18pm are people’s work time, which make people’s activity pattern become unification and activity area have greater regularity. Hence these periods also have higher prediction accuracy. The periods 6am-9am, 12am-14pm and 18pm-24am usually are people’s entertainment-time, which make people’s activity behaviors greater randomness and the predictability of moving trajectory be lower. As a result, these periods have lower prediction accuracy. Comparing three groups of experimental results, ATPDC algorithm can timely update trajectory model with continual incremental user data, and the trajectory model which is trained by ATPDC algorithm has higher prediction accuracy. The reason is that the algorithm is able to use new user trajectory data to rectify and update original trajectory model. In trajectory modeling stage, merging new and old trajectory clusters and adding new trajectory points make the hotspot access area within moving trajectory to change adaptively with the change of user activity behaviors. And non-hotspot access area which is not visited frequently by the user can be timely deleted in trajectory updating.
stage. Finally, trajectory points which are output by the algorithm from moving trajectory can describe the typical activity trajectory of the user and effectively predict user moving behaviors.

As shown in Fig. 2, the total average prediction accuracy of trajectory model is generated in the case that three different data sets are used as the input of experiments. It is obvious that the model generated by ATPDC algorithm has the highest total average prediction accuracy. The accuracy is higher about 6% than non-self-adaptive experiment, and higher about 5% than half-self-adaptive experiment. It is shown that the ATPDC algorithm is an effective incremental trajectory prediction algorithm.

Fig. 2: Total average prediction accuracy of trajectory model

Fig. 3: Total average prediction time of trajectory model

5.4 Trajectory prediction time

As shown in Fig. 3, the comparison of mean execution time of three trajectory models are respectively obtained by training MR data of 100 users during 13 days. It can be seen that ATPDC algorithm takes least time in self-adaptive environment and most time in non-self-adaptive environment. It is the main reason that it avoids to use all new and old data to rebuild moving trajectory model, and only update existing model based on analyzing new user data in self-adaptive situation. As a result, it can improve the execution efficiency of the algorithm. With the growth of user data, all data are used to rebuild model, which is a very time-consuming operation. And it makes the execution efficiency of the algorithm be lower in non-self-adaptive situation.

6 Conclusion

Existing research mainly focuses on mining continuous trajectory of moving objects. But they can not be well applied to user behavior patterns discovery in mobile communication environments. Generally, user data in mobile communication environments is characterized by huge amounts, complex data format, discrete distribution, dynamic changes, and so on. And it makes it difficult to apply the existing research findings in the mobile communication environments. In order to solve the problem of user behavior analysis, we take MR data as research data and propose a self-adaptive trajectory prediction method based on density clustering in the paper. In the trajectory modeling stage, trajectory clusters are firstly formed by clustering new moving data for user using density-based clustering algorithm, and then original trajectory model is generated
by merging new and old trajectory clusters. In the trajectory updating stage, the algorithm updates again original trajectory model in order to solve the problem of structure change between trajectory clusters. Through three experiments we verify that ATPDC algorithm is able to auto-update trajectory model with the growth of user data, and it has higher prediction accuracy and execution efficiency in the self-adaptive situation. Additionally, experimental results show that user behavior patterns are implied in user MR data and the mobile communication data is also able to be used to analyze user behaviors.

Acknowledgement

This work is supported by the National Science Foundation of China under the Grant of 61402363, 61272284, and Natural Science Foundation Project of Shaanxi Province under the Grant of 2014JQ8361, and Xi’an Science Program Project under the Grant of CXY1339(5), and Beilin district of Xi’an Science and Technology Project under the Grant of GX1405, and Program Project in the Xi’an University of Technology under the Grant of 116-211302.

References