1. Introduction
2. Related Work
3. Preliminary

In this section, we will clarify some terms used in subsequent discussion.

**Definition 1** (Location point). A location point is a triple of longitude, latitude and timestamp in the form generated from raw GPS positioning data.

**Definition 2** (Trajectory). A trajectory is an ordered sequence of location points in the form where and .

**Definition 3** (Cell-temporal point). In the route pattern mining procedure, the study area is divided into grid cells with a width of , as shown in Fig.1. A cell-temporal point (CTP) is a triple of abscissa, ordinate and timestamp in the form where and are in Cartesian coordinate system. The subscript of means that this point is the point of trajectory after preprocessed. A CTP is transformed from a location point according to the distribution of cells in the grid. Assume that point with time component shown in Fig.1 is located in the cell after coordinate transformation, then the corresponding CTP can be represented as .

**Definition 4** (Cell-temporal sequence). A cell-temporal sequence (CTS) is an ordered sequence of CTPs in the form abstracted from one trajectory where and . One sample CTS in Fig.1 is .

**Definition 5** (Cell density). A cell density is the number of CTPs belong to this cell. For example, the density of cell is 1 while the cell is 2.

**Definition 6** (Region of interest). Given the density of each cell in grid, we can merge some adjacent cells into rectangular area, namely region of interest (ROI), according to the cell density using particular algorithm. There are three ROI in Fig.1 namely , and .

**Definition 7** (Region sequence). A region sequence (RS) is an ordered sequence of ROIs in the form abstracted from one trajectory.

**Definition 8** (Move vector). A move vector (MV) is in the form indicated a route pattern that coming from region and arriving at region at timestamp *t*.

**Definition 9** (). A ) is a set of move vectors passing by the region *r*. For example, means that user arrived at region at time through , orderly as shown in Fig.1.



**Fig. 1.** An example of location points, CTPs and ROIs.

1. Architecture

In this section, the architecture of proposed next place prediction system is introduced.

* 1. Data preprocessing

Due to the influence of weather and obstacle, there are outliers in the data obtained by GPS devices need to be removed before route pattern mining. Three filters have been developed to remove these outliers.

### 4.1.1 Orientation filter

After the visualization of trajectory dataset, we find that trajectories are not straight or just turning at the crossroad simply, because user tends to stay at a particular place for a while during the travel while GPS device keep recording the location data. As a side effect of this situation, redundant data are recorded. The left part of Fig.2 shows an example, in which there is one trajectory namely *T* starting at location point *pstart* and ending at *pend* , passing by the points described as solid circles. The right part of Fig.2 is the local enlarged drawing to the part of trajectory in dashed rectangle. As we can see, location points in dashed circle is dense and messy. These redundant points are too specific to abstract user’s route pattern. For next place prediction system, the valuable information is the travel pattern that user starts from *pstart*, passing by the dashed circle zone, finally ends at *pend*. The redundant data could increase computing load and cause prediction failure. So, the prediction system needs to have the ability to normalize the redundant location points in the circle to one characteristic point like the star in the circle. A sliding window based orientation filter is developed to recognize and remove this type of data.

Firstly, the filter algorithm calculates the orientation angle of each location points behind *pstart*. The definition of orientation angle is denoted in the right part of Fig.2 clearly. Take the point *p2* for example, the angle between the vector and upward direction (geographically the north direction) is the orientation angle belongs to *p2*. All the orientation angles of trajectory *T* is shown as Fig.3.



**Fig. 2.**



**Fig. 3.**

The sliding window has two parameters as its attribute, window size and window height. When filter algorithm executed on a trajectory, the window will slide along location points belong to this trajectory in the order of time component of these points. The window size parameter limits the number of points that fall within the window each time the window slides. The window height parameter limits the maximum difference between points within the window. In Fig.3, window size is set to 3 and window height is set to 40 degree. During the sliding process of window, the continuous points those always located in the window corresponds to the continuous portion of the trajectory in Fig.2, such as the potion between *pstart*and *p1*. The points those cannot be held in the window correspond to the messy points, such as points between *p1* and *p6*, those need to be normalized to one characteristic point. The normalized result can be obtained by calculating the mean of longitude and latitude of redundant points.

The Algorithm 1 shows the process of sliding window based orientation filter.

|  |
| --- |
| **Algorithm 1** |
| Input: A grid with densities , a density threshold  Output: A set of rectangular regions over .  ⒈  ⒉ |

### 4.1.2 Duplication filter

When the speed of user moving is low or GPS device positioning frequency is relatively high, the collected location points are more intensive or even repeat. In next place prediction system, there is no necessary to use such high degree of intensive data, so the adjacent points and repeat points can be removed. If the distance between two consecutive points is smaller than a threshold , the duplication filter removes the second one. The number of location points and the calculation load are reduced while the route pattern is retained.

### 4.1.3 Outlier filter

Outlier filter is used to remove outlier and smooth the trajectory. After reading in three consecutive location points, such as *p1*, *p2* and *p3*, the outlier filter calculates the angle , if the angle is smaller than threshold then the point *p2* will be removed.

* 1. 4.2 System architecture

1. The Mining of Route Pattern

In this section, we describe how to mine route patterns using preprocessed trajectory data. The proposed mining procedures are consisted in four stages: construction of Cell-Temporal Sequences (CTSs), construction of Regional Sequences (RSs), construction of Move Vectors (MVs) and the building of prediction graph.

Our mining approach starts by discretizing the working space using a regular grid with cells of small size. Each cell in the grid has a corresponding coordinate. The location points are abstracted using ordinary Cartesian coordinates so that each point corresponds to a particular cell in the grid. For the reason of the short time interval of series location points or the stay of user in one place, one cell may contain consecutive location points belong to one trajectory. We remove duplicate points to reduce computational complexity. Linear interpolation is used to ensure that all cells that users have passed can be extracted. At last, we can obtain a Cell-Temporal Sequence (CTS) represent a single trajectory. Similarly, all CTSs can be constructed using all trajectories.

During the construction of CTSs, another product is the density of each cell. Regions-Of-Interests (ROIs) can be constructed based on the cell density using Algorithm 1 which is similar to the work of Giannotti *et al*. [].

|  |
| --- |
| **Algorithm 1** |
| Input: A grid with densities , a density threshold  Output: A set of rectangular regions over .  ⒈  ⒉  ⒊  ⒋  ⒌  ⒍  ⒎  ⒏  ⒐  ⒑  ⒒  ⒓  ⒔  ⒕  ⒖  ⒗ |

**Fig. 2.** The algorithm used to construct the ROIs.

Fig.3 shows an example of route pattern mining. The coordinate of cell in lower left corner of grid is , and in upper right corner is . A triple in the form is a CTP belongs to the cell . We can find a CTS in the grid, for example, (). Finally, we obtain 7 ROIs according to the Algorithm 1.

According to the ROIs and time component of location point in trajectory, it is simple to obtain the RSs and MVS(r) for all trajectories of user. Take Fig.3 for example, there are 3 RSs is the grid. They are , and . Considering the region , there are two MVs namely and . Then, the prediction graph will be generated based on RSs and MVs, as shown in Fig.4. Every node in the prediction graph represents a ROI extracted in previous stage. Each node in the graph contains two members. The letter in one circle is the index of this ROI. The triples in the rectangle beside the circle are the MVS of this node. Take the node *f* in the graph, corresponding to region , for example. According to the Fig.3, there are two trajectories go through region from region to region , and the timestamps arriving at are and respectively. For node *f* in Fig.4, we have .



**Fig. 3.** An example of the route patterns of a user.



**Fig. 4.** The prediction tree for the example in Fig.3.

1. The Prediction of Next Position

We first analyze the probabilistic model used in next-position prediction and the basic prediction algorithm, then propose the time-decay based prediction algorithm.

* 1. The Simple Probability Model

We build the next-position prediction model based on a probability analysis. The ROIs constructed in the mining procedure are used to describe user route patterns. Given the current ROI *r* or a series of ROIs of a user, the predicted next ROI that the user will visit is determined by the following equation:

or

where is the probability that and *r* simultaneously occur, which means a user will visit region right after the current region *r*. is the conditional probability that region will be visited given *r*. Because the user is currently at the region *r*, that is *P*(*r*)=1,. The probability matrix for region *r* is defined by following equation:

where the conditional probability is the probability of going to right after the current region *r*. The conditional probability in the matrix can be calculated using the number of user obtained from the route pattern mining procedure. So, the matrix *M*(*r*) can be expressed as:

where is the total number of MVs that contain a move from current ROI *r* to ROI *k* according to the classical probability model which is similar to the work of Chen *et al*. []

* 1. The Basic Prediction Algorithm

According to the basic prediction algorithm, the critical part of the probability matrix calculation is to calculate the number and the total routes number those go through the region *r*. Take the region in Fig.3 for example, we have ,,,. So, and the region is the best possibility next position.

* 1. The Route Prediction Algorithm Based on Time Decay

One obvious weakness in the basic prediction algorithm is that it does not consider the support attenuation characteristics of history data as a function of time. It is not hard to understand that recently collected data are more credible than old data, not least the users’ trajectory data. We call this feature effectiveness. Another feature of user travel is the periodicity. For example, one employee used to go to work at 8 a.m. and go home at 5 p.m. from Monday to Friday. If we predict his next location at 8:30 a.m., it will be very likely that he is on the way to work instead of on the way home.

To take serious notice of support attenuation characteristics of trajectory data over time, we proposed the prediction algorithm based on time decay used in prediction procedure.

Using the prediction graph constructed in the mining stage, we can obtain for ROI . Similar to formula (), we define

where is the valid total number of MVs that contain a move from current ROI r and ,which can be calculated by next formula, is the valid number of MVs that contain a move from current ROI *r* to ROI *k*.

where is the weight of effectiveness, and is the weight of periodicity satisfying . is the effectiveness coefficient of the move vector in satisfying . There are two method to calculate appropriately as shown in formula () and formula ().

where is the time component of move vector in satisfying , and is the current date of prediction algorithm executed. Q, N and T are the parameters.

is the periodicity coefficient of the *­­* move vector in satisfying . can be calculated using following equations:

where is the minimum time interval (in hours) between and current time .

The next-position region *k* which is the output of the algorithm can be expressed as:

Considering about the example in 7.2, take features of effectiveness and periodicity into account, assume that , ，, , and , then we have . So, the next-position we predict is region

1. Performance Evaluation and Discuss

实验数据来源介绍，实验用所有数据点的叠加图

数据属性展示（轨迹条数，轨迹点数等）

数据过滤效果分析（数据，地图截图）

实验设置的参数解释（不同参数值设置会导致的结果展示图表）

预测过程展示（网页截图），给定某点后，1给出网页截图，2后台给出得到的Region Map，给出最终打分

预测结果分析（怎么选择训练集测试集；多种算法的对比；已知一点，已知多点情况下的结果对比）

In this section, we introduce the evaluation of the next place prediction system. In consideration of the repeatability of our evaluation and ensuring the accuracy of trajectory dataset, we use the dataset published by Microsoft Research Asia in the year of 2012[]. This GPS trajectory dataset was collected in Geolife [] project by 182 users in a period of over 5 years (from April 2007 to August 2012). A GPS trajectory of this dataset is represented by a sequence of time-stamped points, each of those contains the information of latitude, longitude and altitude. We choosed ? users trajectory data to carry out experiments. Figure ? plots the distribution (heat map) of data we used. All GPS data are located in the city of Beijing. Table ? shows the total numbers of location points we used in performance evaluation.

1. Conclusions

Acknowledgements.

Reference