1 Introduction

2 Related Work

3 Architecture

4 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 $p =< lang, lat, t >$ generated from raw GPS positioning data.

Definition 2 (Trajectory). A trajectory is an ordered sequence of location points in the form $T = (p_1, p_2, \cdots, p_n)$ where $p_{i+1}.t > p_i.t$ and $p_{i+1}.t - p_i.t < \Delta T$ $(1 \leq i < n)$.

Definition 3 (Cell-temporal point). A cell-temporal point (CTP) is a triple of abscissa, ordinate and timestamp in the form $< x, y, t >$ where x and y are in Cartesian coordinate system. A CTP is transformed from location point according to the distribution of cells in the grid.

Definition 4 (Cell-temporal sequence). A cell-temporal sequence (CTS) is an ordered sequence of CTPs in the form $CTS = (ctp_1, ctp_2, \cdots, ctp_n)$abstracted from one trajectory where $ctp_{i+1}.t > ctp_i.t$, $ctp_{i+1}.x - ctp_i.x \leq 1$, $ctp_{i+1}.y - ctp_i.y \leq 1$ and $(ctp_{i+1}.x - ctp_i.x) + (ctp_{i+1}.y - ctp_i.y) \neq 0$.

Definition 5 (Cell density). A cell density is the number of CTPs belong to this cell.

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.

Definition 7 (Region sequence). A region sequence (RS) is an ordered sequence of ROIs in the form $RS = (r_1, r_2, \cdots, r_n)$ abstracted from one trajectory.

Definition 8 (Move vector). A move vector (MV) is in the form $< r_{pre}, r_{next}, t >$ indicated a route pattern that coming from region r_{pre} and arriving at region r_{next} at timestamp *t*.

Definition 9 ($MVS(r)$). A $MVS(r)$ is a set of move vectors passing by the region *r*. For example, $MVS(r_a) = \{< r_b, r_c, t_1 >\}$ means that user arrived at region r_c at time t_1 through r_b, r_a orderly.

5 Data Preprocessing

6 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 three 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 provided in Fig.1.

Algorithm 1 ConstructROI(G, δ)

Input: A grid G with densities $G(i,j)$, a density threshold δ
Output: A set R of rectangular regions over G.

1. $R = \emptyset$; $G^* = \{(i,j) \epsilon G | G(i,j) \geq \delta\}$;
2. $foreach\ (i,j) \epsilon G\ do\ used(i,j) = false$;
3. $foreach\ (i,j) \epsilon G^*\ in\ descending\ order\ of\ G(i,j)\ do$
4. $\quad if\ \neg used(i,j)\ then$
5. $\quad\quad r = \{(i,j)\}$;
6. $\quad\quad repeat$
7. $\quad\quad\quad foreach\ dir \epsilon \{left, right, up, down\}\ do$
8. $\quad\quad\quad\quad r_{dir} = r\ extended\ on\ direction\ dir$;
9. $\quad\quad\quad\quad ext = \{dir | r_{dir} \subseteq G\ \wedge avg_{density(r_{dir})} \geq G(i,j)\}$;
10. $\quad\quad\quad if\ ext \neq \emptyset\ then$
11. $\quad\quad\quad\quad dir = \arg max_{d \epsilon ext} avg\_density(r_d)$;
12. $\quad\quad\quad\quad r = r_{dir}$;
13. $\quad\quad until\ ext = \emptyset$;
14. $\quad\quad foreach\ (i,j) \epsilon r\ do\ used(i,j) = true$;
15. $\quad\quad R = R \cup \{r\}$;
16. $return\ R$;

Fig. 1. The algorithm used to construct the ROIs.

Fig.2 shows an example of route pattern mining. The coordinate of cell in lower left corner of grid is (0,0) , and in upper right corner is (5,3). A triple in the form $< x, y, t_{ij} >$ is a CTP belongs to the cell (x, y). We can find a CTS in the grid, for example, $(< 3,0, t_{11} >, < 3,1, t_{12} >, < 4,1, t_{13} >, < 4,2, t_{14} >)$. 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.2 for example, there are 3 RSs is the grid. They are $S_1 =< r_a, r_b >$,$S_2 =< r_a, r_c, r_d, r_e, r_g >$ and $S_3 =< r_a, r_f, r_g >$. Considering the region r_f, there are two MVs namely $< r_a, r_g, t_{55} >$ and $< r_a, r_g, t_{65} >$. 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 r_f, for example. According to the Fig.2, there are two trajectories go through region r_f from region r_a to region r_g, and the timestamps arriving at r_g are t_{55} and t_{65} respectively. For node *f* in Fig.3, we have $MVS(r_f) = \{< r_a, r_g, t_{55} >, < r_a, r_g, t_{65} >\}$.

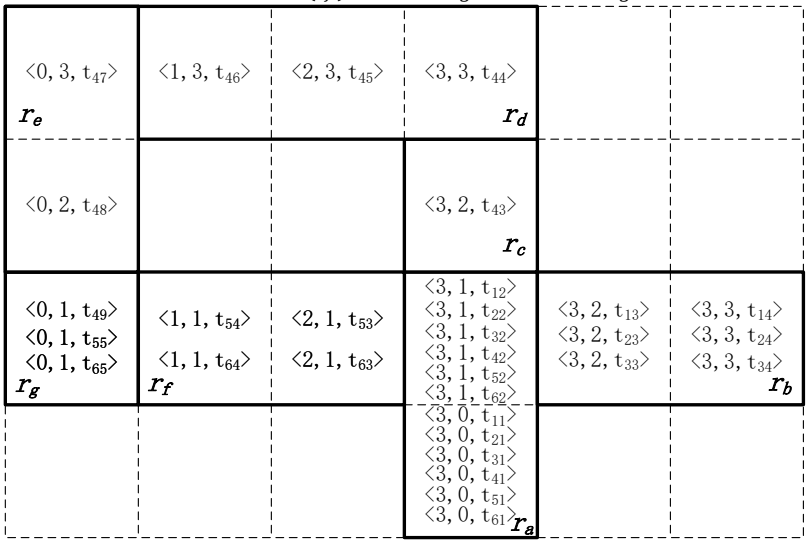

Fig. 2. An example of the route patterns of a user.

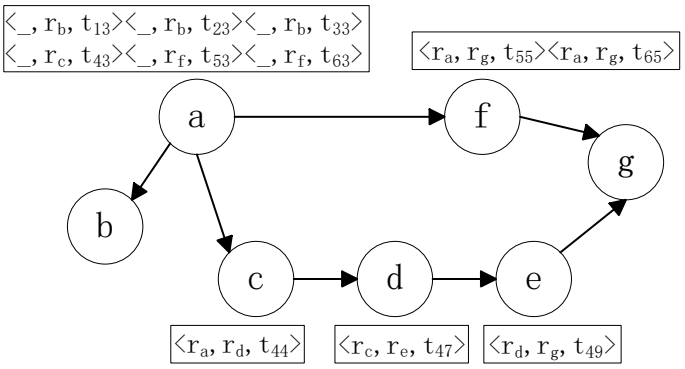

Fig. 3. The prediction tree for the example in Fig.2.

7 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.

7.1 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 $(r_1, r_2, \cdots, r_k)$ of a user, the predicted next ROI r_{next} that the user will visit is determined by the following equation:

$$P(r_{next}, r) = P(r_{next}|r) * P(r)$$

or

$$P(r_{next}, r_1, r_2, \ldots, r_k) = P(r_{next}|r_1, r_2, \ldots, r_k) * P(r_1, r_2, \ldots, r_k)$$

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

$$M(r) = [\, P(r_1|r) \quad P(r_2|r) \quad \cdots \quad P(r_k|r)]$$

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

$$M(r) = [P(r_1|r) \quad P(r_2|r) \quad \cdots \quad P(r_k|r)] = \left[\frac{N_{r,r_1}}{N_r} \quad \frac{N_{r,r_2}}{N_r} \quad \ldots \quad \frac{N_{r,r_k}}{N_r}\right]$$

$$N_r = \sum_{i=1}^{k} N_{r,r_i}$$

where N_{r,r_k} 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 Ling *et al*. []

7.2 The Basic Prediction Algorithm

According to the basic prediction algorithm, the critical part of the probability matrix calculation is to calculate the number N_{r,r_k} and the total routes number N_r those go through the region *r*. Take the region r_a in Fig.2 for example, we have $N_{r_a} =$

$6, N_{r_a,r_b} = 3, N_{r_a,r_c} = 1, N_{r_a,r_f} = 2$. So, $M(r_a) = \left[\frac{3}{6} \quad \frac{1}{6} \quad \frac{2}{6}\right]$ and the region r_b is the best possibility next position.

7.3 The 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 $MVS(r_i)$ for ROI r_i. Similar to formula (), we define

$$M(r) = [P(r_1|r) \quad P(r_2|r) \quad \cdots \quad P(r_k|r)] = \left[\frac{E_{r,r_1}}{E_r} \quad \frac{E_{r,r_2}}{E_r} \quad \cdots \quad \frac{E_{r,r_k}}{E_r}\right]$$

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

$$E_{r,r_k} = \sum_{i=1}^{N_{r,r_k}} (\omega_1 * \varphi_i + \omega_2 * \eta_i) = \omega_1 * \sum_{i=1}^{N_{r,r_k}} \varphi_i + \omega_2 * \sum_{i=1}^{N_{r,r_k}} \eta_i$$

where ω_1 is the weight of effectiveness, and ω_2 is the weight of periodicity satisfying $\omega_2 = 1 - \omega_1$. φ_i is the effectiveness coefficient of the *i*-th move vector in $MVS(r)$ satisfying $MVS(r).r_{next} = r_k$. There are two method to calculate φ_i appropriately as shown in formula () and formula ().

$$\varphi_i = M * (\frac{1}{2})^{\frac{t_i - t_D}{T}}$$

$$\varphi_i = Q + e^{-N|t_i - t_d|}$$

where t_i is the time component of *i*-th move vector in $MVS(r)$ satisfying $MVS(r).r_{next} = r_k$, and t_D is the current date of prediction algorithm executed. Q, N and T are the parameters.
η_i is the periodicity coefficient of the *i*-th move vector in $MVS(r)$ satisfying $MVS(r).r_{next} = r_k$. η_i can be calculated using following equations:

$$\eta_i = \frac{m_i}{\sum m}$$

$$m_i = 24 - \alpha(t_i, t_T)$$

where $\alpha(t_i, t_T)$ is the minimum time interval (in hours) between t_i and current time t_T. The next-position region *k* which is the output of the algorithm can be expressed as:

$$k = \arg\ max\, E_{r,r_k}$$

Considering about the example in 7.2, take features of effectiveness and periodicity into account, assume that $\varphi_{t_{13}} = \varphi_{t_{23}} = \varphi_{t_{1=33}} = \varphi_{t_{43}} = \frac{1}{3}$, $\varphi_{t_{53}} = \varphi_{t_{63}} = \frac{2}{3}$, $\eta_{t_{13}} = \eta_{t_{23}} = \eta_{t_{1=33}} = \eta_{t_{43}} = \frac{2}{3}$, $\eta_{t_{53}} = \eta_{t_{63}} = \frac{1}{3}$, and $\omega_1 = \omega_2 = \frac{1}{2}$, then we have $E_{r,r_f} = \frac{4}{3} > E_{r,r_b} = 1 > E_{r,r_c} = \frac{1}{3}$. So, the next-position we predict is region r_f.

8 Performance Evaluation and Discuss

9 Conclusions

Acknowledgements.

10 Reference