Learning Travel Recommendations From User-Generated GPS Traces

YU ZHENG AND XING XIE
Microsoft Research Asia

The advance of GPS-enabled devices facilitates people to record their location histories with GPS traces, which imply human behaviors and preferences related to travel. In this paper, we perform two types of travel recommendations by mining multiple users' GPS traces. The first is a generic one that recommends a user with top interesting locations and travel sequences in a given geospatial region. Here, interesting locations mean the culturally important places, such as Tiananmen Square in Beijing, and frequented public areas, like shopping malls and restaurants. The second is a personalized recommendation that provides an individual with locations matching her travel preferences. To achieve the first recommendation, we model multiple users' location histories with a tree-based hierarchical graph (*TBHG*). Based on the *TBHG*, we propose a HITS (Hypertext Induced Topic Search)-based inference model, which regards an individual's access on a location as a directed link from the user to that location. This model infers two values, the interest level of a location and a user's travel experience, by taking into account 1) the mutual-reinforcement relation between the two values and 2) the geo-region conditions. Considering the inferred values, we mine the classical travel sequences among locations. In the personalized recommendation, we first understand the correlation among locations in terms of 1) the sequences that the locations have been visited and 2) the travel experiences of the persons accessing these locations. Beyond the geo-distance relation, this correlation represents the relation between locations in the spaces of human behavior. Later, we incorporate the location correlation into a collaborative filtering (CF)-based model that infers a user's interests in an unvisited location based on her locations histories and that of others. We evaluated our system based on a real-world GPS trace dataset collected by 107 users over a period of one year. As a result, our HITS-based inference model outperformed baseline approaches like *rank-by-count* and *rank-by-frequency*. Meanwhile, when considering the users' travel experiences and location interests, we achieved a better performance in recommending travel sequences beyond baselines including *rank-by-count* and *rank-by-interest*. Regarding the personalized recommendation, our approach is more effective than the weighted Slope One algorithm with a slightly additional computation. In addition, in contrast to the Pearson correlation-based CF model, our method is much more efficient while keeping the similar effectiveness.

H.2.8 [**Database Management**]: Database Applications - *data mining, Spatial databases and GIS*. H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *clustering, retrieval model*.

General Terms: Algorithms, Measurement, Experimentation

Additional Key Words and Phrases: Location recommendation, Location History, GPS trace, Collaborative filtering, GeoLife.

1. INTRODUCTION

GPS-enabled devices are changing the way people interact with the Web by using locations as contexts. With such a device, a user is able to acquire present locations, search for the information around them and find driving directions to a destination. Recently, many users start recording their outdoor movements with GPS traces for many

This article is an expanded version of Zheng et al. [2009], which appears in Proceedings of the 18th International Conference on the World Wide Web, 791–800.
Authors' addresses: Y. Zheng and X. Xie, Microsoft Research Asia, 4F Sigma Building, No. 49 Zhichun Road, Haidian District, Beijing 100190, China; email: {yuzheng, xingx}@microsoft.com.

reasons, such as travel experience sharing [Counts and Smith 2007; GeoLife 2007], life logging [Zheng et al. 2008c and 2008d] and sports activity analysis [SportsDo 2007; Bikely 2006]. At the same time, a branch of websites or forums that enable people to establish some geo-related Web communities have appeared on the Internet. By uploading their GPS traces to such communities, individuals can manage their travel experiences on a Web map and share travel knowledge among each other. For instance, a person is able to find some attractive places from other people's travel routes, and plan an interesting and efficient journey based on multiple users' experiences.

Although a huge amount of GPS traces have been accumulating, most of these applications still directly use raw GPS data, like coordinates and time stamps, without much understanding. Being faced with such a large dataset, it is impossible for community users to browse GPS traces one by one and identify interesting locations by themselves. That is, the travel recommendations provided by these communities are not comprehensive and significant enough.

Typically, people need two types of recommendations during a journey, generic and personalized recommendations. Regarding the generic recommendation, people usually desire to know the most popular/interesting locations in a geospatial region and the classical travel sequences among these locations. To define interesting location, we mean the culturally important places, such as Tiananmen Square in Beijing and the Statue of Liberty in New York (i.e. popular tourist destinations), and commonly frequented public areas, such as shopping malls/streets, restaurants, cinemas and bars. With the information mentioned above, an individual can understand an unfamiliar city in a very short period and plan their journeys with minimal effort. Besides the generic recommendation, an individual also wants to visit some locations matching her travel preferences (personalized). For instance, a food-lover prefers to find some restaurants providing delicious foods although these restaurants might not be the most popular places in a city.

However, we will meet some challenges when conducting these two types of recommendations. First, the interest level of a location does not only depend on the number of users visiting this location but also lie in these users' travel experiences. Intrinsically, different people have different degrees of knowledge about a geospatial region. In a journey, the users, with more travel experiences about a region, would be more likely to visit some interesting locations in that region. For example, the local people of Beijing are more capable than overseas tourists of finding out high quality restaurants and famous shopping malls in Beijing. Second, an individual's travel experience and interest level of a location are relative values (i.e., it is not reasonable to judge whether or not a location is interesting), and are region-related (i.e., conditioned by the given geospatial region). A user, who has visited many places in a city like New York, might have no idea about another city, such as Beijing. Likewise, the most interesting restaurant in a district of a city might not be the most interesting one of the whole city (as other restaurants from the remaining districts might outperform it). Third, current CF models are not good enough to understand an individual's travel preferences from her location history and predict her interests in an unvisited location. The traditional item-based methods [Linden et al. 2003; Lemire et al. 2005] have a good online efficiency while cannot well model human travel behaviors, such as the visited sequence of locations and travel knowledge of different users. Some user-based CF models [Li et al. 2008; Zheng et al. 2010b] (using user similarity) can model human travel behavior while will cause a huge computation loads (as these method needs to compute the similarity between each pair of users and the number of users could increase continuously).

In this paper, we conduct both generic and personalized travel recommendations based on multiple users' GPS traces. To achieve the generic recommendation,

- We propose a tree-based hierarchical graph (*TBHG*), which can model multiple users' travel sequences on a variety of geospatial scales.
- Based on the *TBHG*, we propose a HITS-based model to infer users' travel experiences and interest of a location within a region. This model leverages the main strength of HITS to rank locations and users with the context of a geospatial region, while calculating hub and authority scores offline. Therefore, we can ensure the efficiency of our system while supporting users to specify any geo-regions as queries.
- Considering an individual's travel experiences and the interests of a location as well as people's transition probability between locations, we mine the classical travel sequences from multiple users' location histories.

To conduct the second recommendation, we first mine the correlation among locations from multiple users' GPS traces in terms of 1) the sequences that the locations have been visited and 2) the travel experiences of the users creating these sequences. Later, the location correlation is incorporated into a CF-based model that infers a user's interests in an unvisited location based on her locations histories and that of others.

The rest of this paper is organized as follows. Section 2 summarizes the related work. Section 3 introduces some basic concepts used in this paper and gives an overview of our work. Section 4 describes the process of location history modeling. Section 5 presents the methodology of mining interesting locations and classical travel sequences. Section 6 illustrates the method of mining location correlation. Section 7 details the recommenders. Section 8 reports on major experimental results followed by some discussions. Section 9 concludes this article.

2. RELATED WORK

2.1 Mining Location History

*2.1.1 Mining individual location history***.** During the past years, a branch of research has been performed based on individual location history recorded in GPS traces. These works include detecting significant locations of a user [Ashbrook et al. 2003; Hariharan and Toyama 2004], predicting the user's movement among these locations [Krumm et al. 2007; Liao et al 2005], and recognizing user-specific activities at each location [Liao et al. 2004 and Patterson et al. 2003]. As opposed to these works, we aim to model multiple users' location histories and learn patterns from numerous individuals' behaviors.

2.1.2 Mining multiple users' location histories. Gonotti et al. [2007] mined similar sequences from users' moving trajectories, and Mamoulis et al. [2004] proposed a framework for retrieving maximum periodic patterns in spatio-temporal data. MSMLS [Krumm et al. 2007] used a history of a driver's destinations, along with data about driving behavior extracted from multiple users' GPS traces, to predict where a driver may be going as a trip progresses. Eagle et al. [2006] aimed to recognize the social pattern in daily user activity from the dataset collected by 100 users with a Bluetooth-enabled mobile phone. Based on raw GPS data, Zheng et al. [2008a; 2008b; 2010a] classified people's GPS trajectories into different categories of transportation modes consisting of driving, walking, taking a bus, and riding a bike. In contrast to these techniques, we extend the paradigm of mining users' location histories from exploring users' behaviors to understanding locations and modeling the relation between users and locations.

2.2 Location Recommenders

2.2.1 Recommenders based on real-time location. Mobile tourist guide systems typically recommend locations and sometimes provide navigation information based on a user's real-time location. Previously, such kinds of systems were somehow naïve as they always returned the information close to an individual without understanding the individual and the nearby locations. Recently, some researchers aim to filter away from the returned results the invisible entities occluded by the nearby building [Beeharee and Steed 2007; Simon and Fröhlich 2007]. Meanwhile, another branch of work [Abowd et al. 1997; Park et al. 2007] started involving a user's location history in these systems to provide the user with a more personalized recommendation. In contrast to these techniques, we aim to integrate the social environment of an individual into travel recommenders by helping the individual deeply understand the locations around them with the knowledge mined from not only their own but also other users' location histories.

2.2.2 Recommenders based on location history. Using multiple users' real-world location histories, some recommender systems, such as *Geowhiz* [Horozov et al. 2006], *CityVoyager* [Takeuchi and Sugimoto 2006], and GeoLife [Zheng et al. 2009a; Li et al. 2008; Zheng et al. 2010b], have been designed to recommend geographic locations like shops or restaurants to users. Horozov et al. [2006] proposed an enhanced collaborative filtering solution to generate the recommendation of a restaurant. Takeuchi et al. [2006] attempted to recommend shops to users based on their individual preferences estimated by analyzing their past location histories. Li and Zheng et al [2008] first mined a user similarity from human location history by considering the sequence property of travel behaviors and the hierarchical property of geographical spaces. Further, Zheng et al. [2010b] incorporate this user similarity into a user-centric CF model to conduct a personalized friend & location recommendation. Zheng et al. [2010c] use a collaborative learning approach to enable an activity-location recommendation based on GPS traces associated with user-generated comments. That is, given an activity like shopping, the system will recommend the best k locations where this activity can be performed. Meanwhile, by selecting a location, e.g., the Olympic Park of Beijing, a user will be recommended with the best k activities that should be conducted in the location.

The major difference between these works and ours lies in three aspects. First, we differ the travel experiences and knowledge of different users. Therefore, their location histories should be treated in different weights. Second, we consider the relation between locations and users' travel experiences, e.g., the mutual reinforcement relation and the region-related constraints. Third, although the related recommenders also use CF model to predict a user's interests in a location, they either have a poor online efficiency [Zheng et al. 2010b; Zheng et al. 2010c] or cannot well model human travel behaviors [Horozov et al. 2006; Takeuchi and Sugimoto 2006], such as the travel sequence and geo-hierarchy.

3. OVERVIEW

3.1 Application Scenarios

The work reported in this paper is an important component of our project GeoLife [Zheng et al. 2008a, 2008b, 2008c, 2008d, 2009a, 2009b; Chen et al. 2010; Zheng et al. 2010a, 2010b, 1020c], whose prototype has been internally accessible within Microsoft since Oct. 2007. So far, we have had 106 individuals using this system.

Figure 1 shows the user interface of our applications running on desktop computers. In the right part of this figure, we can view the top five interesting locations and the most five experienced users in the region specified by the present view of the Web map. The top five classical travel sequences within this region are also displayed on the map. By zooming in/out and panning this map, an individual can retrieve such results within any regions. In addition, the photos taken at an interesting location will be presented on the bottom of the window after a user clicks the icon representing the location on the map. Once a user has a certain number of GPS traces accumulated in GeoLife, she can view the personalized recommendation that offers locations matching her travel preferences.

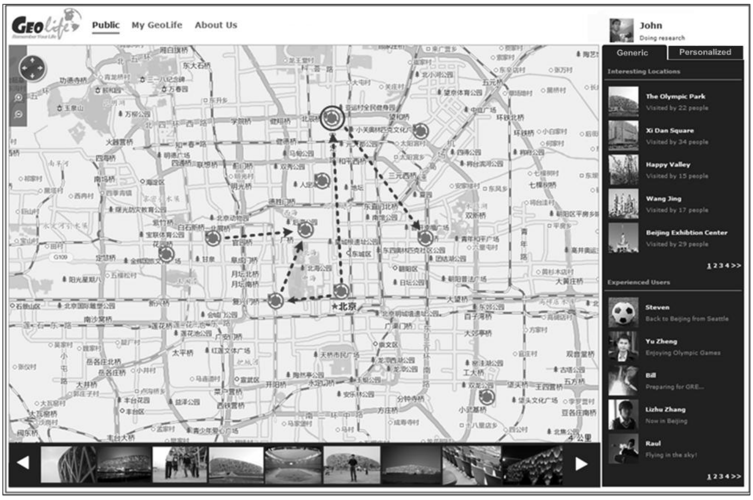


Figure 1. The user interface regarding location recommendation

As shown in Figure 2, a user with a GPS-phone can find out the top five interesting locations as well as the most five classical sequences nearby their present geographic position (the red star). In addition, when the user reaches a location, our system would provide them with a further suggestion by presenting the top three classical sequences start from this location. Of course, users can also view the personalized recommendation on a mobile phone as long as they have GPS traces stored in GeoLife.

Figure 2. Location recommendations on a GPS-phone

3.2 Preliminary

Definition 1. GPS Trace. A GPS trace Tra is a sequence of time-stamped points, $Tra = p_0 \rightarrow p_1 \rightarrow, \dots, \rightarrow p_k$, where $p_i = (x, y, t)$ $(i = 0,1, \dots, k)$; (x, y) are latitude and longitude respectively, and t is a timestamp. $\forall 0 \leq i \leq k,\ p_{i+1}.t > p_i.t$.

Definition 2. $Dist(p_i, p_j)$ denotes the geospatial distance between two points p_i and p_j, and $Int(p_i, p_j)= |p_i.t - p_j.t|$ is the time interval between two points.

***Definition 3*: *Stay Point*.** A stay point s is a geographical region where a user stayed over a time threshold T_r within a distance threshold D_r. In a trace, s is characterized by a set of consecutive points $P = \langle p_m, p_{m+1}, \dots, p_n \rangle$, where $\forall m < i \leq n,\ Dist(p_m, p_i) \leq D_r$, $Dist(p_m, p_{n+1}) > D_r$ and $Int(p_m, p_n) \geq T_r$. Therefore, $s = (x, y, t_a, t_l)$, where

$$s.x = \textstyle\sum_{i=m}^{n} p_i.x/|P|, \qquad (1)$$
$$s.y = \textstyle\sum_{i=m}^{n} p_i.y/|P|, \qquad (2)$$

respectively stands for the average *x* and *y* coordinates of the collection *P*; $s.t_a = p_m.t$ is the user's arriving time on *s* and $s.t_l = p_n.t$ represents the user's leaving time.

As shown in Figure 3, $\{p_1, p_2, \dots, p_8\}$ formulate a trace, and a stay point would be detected from $\{p_3, p_4, p_5, p_6\}$ if $d \leq D_r$ and $Int(p_3, p_6) \geq T_r$. In contrast to a raw point p_i, a stay point carries a particular semantic meaning, such as the shopping mall the user accessed and the restaurant they visited.

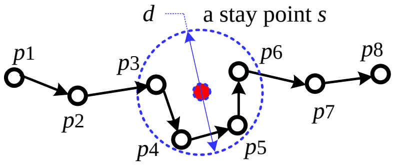


Figure 3. A GPS trace and a stay point

***Definition 4: Location History*.** An individual's location history *h* is represented as a sequence of stay points they have visited with corresponding transition times,

$$h = \langle s_0 \xrightarrow{\Delta t_1} s_1 \xrightarrow{\Delta t_2}, \dots, \xrightarrow{\Delta t_{n-1}} s_n \rangle, \qquad (3)$$

where $\forall 0 \leq i < n, s_i$ is a stay point and $\Delta t_i = s_{i+1}.t_a - s_i.t_l$ is the time interval between two stay points.

Intrinsically, people generate many trips in their lives. For instance, an individual would visit some shopping malls in a trip and start a new trip two days later to go hiking. Thus, we need to partition an individual's location history *h* into some trips if the travel time spent between two consecutive locations exceeds a certain threshold T_p.

Definition 5: Trip: A trip is a sequence of stay points consecutively visited by a user, $Trip = \langle s_0 \xrightarrow{\Delta t_1} s_1 \xrightarrow{\Delta t_2}, \dots, \xrightarrow{\Delta t_{n-1}} s_n \rangle$, where $\forall 1 \leq k < n, \Delta t_k < T_p$ (a threshold).

However, so far, people's location histories are still inconsistent as the stay points detected from various individuals' traces are not identical. To address this issue, we propose the *TBHG* to model multiple users' location histories. Generally speaking, a *TBHG* is the integration of two structures, a tree-based hierarchy *H* and a graph *G* on each level of this tree. The tree expresses the parent-children (or ascendant-descendant) relation of the nodes pertaining to different levels, and the graphs denote the peer relation among the nodes on the same level.

As demonstrated in Figure 4, in our system two steps need to be performed when building a *TBHG*. 1) *Formulate a tree-based Hierarchy H*: We put together the stay points detected from users' GPS logs into a dataset. Using a density-based clustering algorithm, we hierarchically cluster this dataset into some geospatial regions (a set of clusters *C*) in a divisive manner. Thus, the similar stay points from various users would be assigned to the same clusters on different levels. 2) *Build graphs on each level*: Based on the tree-based hierarchy *H* and users' location histories, we can connect the clusters of the same level with directed edges. If consecutive stay points from one trip are individually contained in two clusters, a link would be generated between the two clusters in a chronological direction according to the time serial of the two stay points.

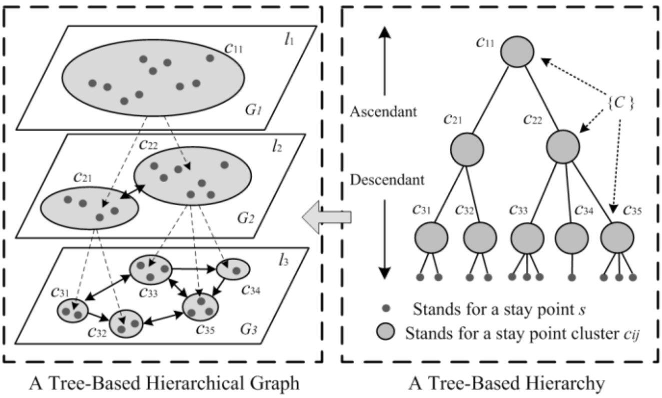


Figure 4. Building a tree-based hierarchical graph

Definition 6. Tree-Based Hierarchy H: *H* is a collection of stay point-based clusters *C* with a hierarchy structure *L*. $H = (C, L)$, $L = \{l_1, l_2, \ldots, l_n\}$ denotes the collection of levels of the hierarchy, and $C = \{c_{ij} \,|\, 1 \le i \le |L|, 1 \le j \le |C_i|\}$ means the collection of clusters on different levels. Here, c_{ij} represents the *j*th cluster on level $l_i \in L$, and C_i is the collection of clusters on level l_i.

Definition 7. Tree-Based Hierarchical Graph (***TBHG***): Formally, a *TBHG* is the integration of *H* and *G*, *TBHG*=(*H*, *G*). *H* is defined in *Definition 6*, and *G*=$\{ g_i = (C_i, E_i), 1 \le i \le |L|\}$. On each layer $l_i \in L$, $g_i \in G$ includes a set of vertexes C_i and the edges E_i connecting $c_{ij} \in C_i$.

Based on the *TBHG*, we can substitute a stay point in a user's location history h with the cluster ID the stay point pertains to. Supposing $s_0 \in c_{31}, s_1 \in c_{32}, s_n \in c_{3n}$, Equation (3) can be replaced with

$$h = \langle c_{31} \xrightarrow{\Delta t_1} c_{32} \xrightarrow{\Delta t_2}, \ldots, \xrightarrow{\Delta t_{n-1}} c_{3n} \rangle. \qquad (4)$$

Therefore, a trip can be represented as a set of consecutive stay-point-clusters sequence ($\forall 0 \le k \le n, \Delta t_k < T_p$), and h can be partitioned into a set of trips on different levels of the hierarchy, $h = \{Trip\}$.

Notations: In the rest of this paper, we use the following notations to simplify the descriptions. $U = \{u_1, u_2, \ldots, u_n\}$ represents the collection of users in a community, $u_k \in U, 1 \leq k \leq |U|$ denotes the kth user, and Tra^k, S^k, h^k and TP^k respectively stand for the u_k's GPS traces, stay points, location history and trips.

3.2 Architecture

Figure 5 shows the architecture of our system, which is comprised of three parts: location history modeling, knowledge mining, and recommendation. The first two operations can be performed off-line, while the last process should be conducted online based on the geo-region specified by a user.

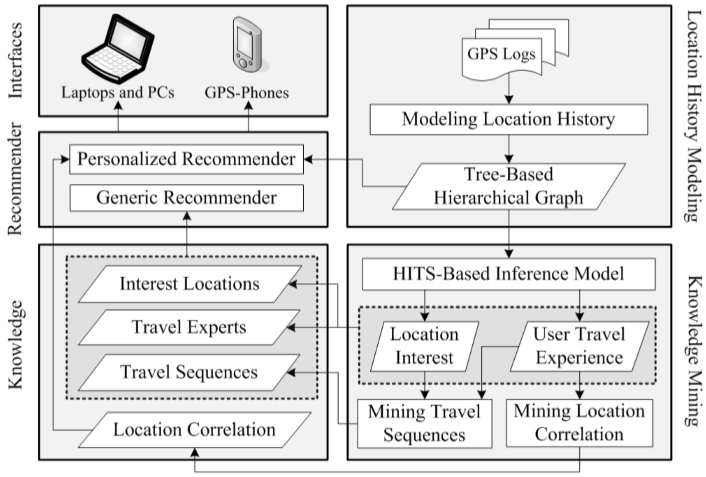


Figure 5. The architecture of our recommendation system

3.2.1 Location history modeling. Given multiple users' GPS traces, we build a *TBHG* off-line. In this structure, a graph node stands for a cluster of stay points, and a graph edge represents a directed transition between two locations (clusters). In contrast to raw GPS points, these clusters denote the locations visited by multiple users, hence would carry more semantic meanings, such as culturally important places and commonly frequented public areas. In addition, the hierarchy of the *TBHG* denotes different geospatial scales (alternatively, the zoom level of a Web map), like a city, a district and a community. In short, the tree-based hierarchical graph can effectively model multiple users' travel sequences on a variety of geospatial scales (Refer to Section 4 for details).

3.2.2 HITS-based inference model. With the *TBHG*, we propose a HITS-based inference model to estimate users' travel experiences and location interests in a given region. In this model, an individual's visit to a location (cluster) is regarded as a directed link from the individual to that location. Thus, a user is a hub if they have visited many locations, and a location is an authority if it has been accessed by many users. Further, a user's travel experience (hub score) and the interest of a location (authority score) have a mutual reinforcement relation. Using a power iteration method, we can generate the final scores for each user and location, and find out the top *n* interesting locations and the top *k* experience users in a given region. As a user's travel experience is region-related, we need to specify a geospatial region as the context for the inference model. Actually, each cluster of the *TBHG* specifies an implied region for its descendant clusters (locations). Therefore, we are able to mine in advance each individual's travel experience and interests of locations conditioned by the regions of clusters on different levels. In other words, a user would have multiple hub scores based on different regions, and a location would have multiple authority scores specified by their ascendant clusters on different levels. This strategy takes the advantage of a HITS model in ranking locations and users based on a region context (query topic), while making the calculations of authority and hub scores offline. (Refer to Section 5.2 for details).

3.2.3 Mining classical travel sequence. We calculate a classical score for each location sequence within a given region considering two factors: the travel experiences of the users taking this sequence and the interests of the locations contained in the sequence. Since there would be multiple paths starting from a location, the interest of this location should be shared among all the paths, with which it points to other locations. The location interest distributed to a path is based on the probability of users' taking the path. Later, the sequences with a relatively high classical score will be retrieved as classical travel sequences. As people would not travel to too many places in a journey, classical sequences containing two or three locations would be more useful than longer ones.

3.2.4 Mining location correlation. Based on the *TBHG*, we compute the correlation between locations by integrating the travel experiences of the users who have visited these locations in a trip in a weighted manner. Beyond the geo-distance relation and the business category similarity between locations, the location correlation describes the relation between locations in the space of human behavior.

3.2.5 Recommendation. This component provides a user with both generic and personalized recommendations according to a user-specified geo-region on web maps. About the generic recommendation, the inferred location interest, user experience and classical score of a sequences are employed to rank locations, users and travel sequences fallen in the given region. Later, the top *n* interesting locations, top *m* classical travel sequences and top *k* experienced users are recommended. With regard to the personalized recommendation, we incorporate the mined location correlation into an item-based CF model to infer a user's interests in a location that has not been visited by the user. Then, we rank these locations in terms of the predicted ratings, and recommend the user the top *n* locations with relatively high ratings.

4. MODELING LOCATION HISTORY

In this Section, we detail the process of modeling multiple users' location histories with a tree-based hierarchical graph. Figure 6 gives a formal description of this operation.

First, we extract stay points (S^k) from each user u_k's traces by seeking the spatial regions where the u_k stayed within a distance threshold D_r over a time spans T_r (refer to [Li et al. 2008] for details). Then, we formulate a location history (h^k) for u_k with these stay points and partition h^k into some trips TP^k.

Second, we put these stay points together into a dataset $SP = \{S^k, 1 \leq k \leq |U|\}$, and hierarchically cluster SP into some geospatial regions C in a divisive manner by using a density-based clustering algorithm. Thus, the similar stay points from various users will be assigned to the same clusters on different levels of the hierarchy *H*. In addition, we would filter away the clusters, which might represent users' homes. If an individual has visited to a cluster with a frequency exceeding a threshold, we believe this cluster may be the individual's home or working place.

Third, based on the tree-based hierarchy *H*, we replace the stay points in each user's trips TP^k with the cluster IDs these stay points belong to. Later, we build connections among the clusters on the same level in terms of the TP^k, and construct a graph g for each level of the *H*. Finally, the hierarchy *H* and graph collection *G* formulate the *TBHG*.

Algorithm LocHisModeling($\boldsymbol{\varphi}, \boldsymbol{D_r}, \boldsymbol{T_r}, \boldsymbol{T_p}$)

Input: The collection of all users' GPS traces: $\varphi = \{Tra^k, 1 \leq k \leq |U|\}$, a distance threshold D_r and time threshold T_r for stay point detection, and a time threshold T_p for trip partition.
Output: a tree-based hierarchical graph: *TBHG*.

```
Foreach u_k ∈ U do
    S^k ← StayPointDetection(Tra^k, D_r, T_r);
    h^k ← PersonalLocHist(S^k, Tra^k);      //build individual location history
    TP^k ← TripPartition(h^k, T_p);         //divide h^k into some trips
    SP.Add(S^k);                            //get the collection of stay points
H← HierarchicalClustering (SP);             //build the hierarchy based on stay points
Foreach l_i ∈ H.L do                        //build a graph on each level
    g_i.C_i ← H.C_i;                        //each cluster represents a node in the graph
    Foreach u_k ∈ U do
        TP^k ← LocHistRepresentation(TP^k, C_i);  //replace stay points with the clusters
        g_i ← GraphBuilding(g_i, TP^k);     //connect nodes based on the trips
    G.Add(g_i);
TBHG ← (H, G);
Return TBHG;
```

Figure 6. The algorithm for location history modeling

5. MINING INTERESTING LOCATIONS AND TRAVEL SEQUENCES

In this Section, we first briefly introduce the key idea of HITS and then describe *our* HITS-based inference model. Later, using such inference results, we mine the classical travel sequences from each graph of the *TBHG.*

5.1 Basic Concepts of HITS

HITS stands for hypertext induced topic search, which is a search-query-dependent ranking algorithm for Web information retrieval. When the user enters a search query, HITS first expands the list of relevant pages returned by a search engine and then produces two rankings for the expanded set of pages, authority ranking and hub ranking. For every page in the expanded set, HITS assigns them an authority score and a hub score.

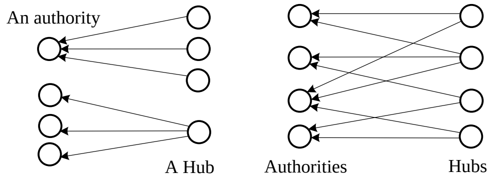


Figure 7. The basic concept of HITS model

As shown in Figure 7, an authority is a Web page with many in-links, and a hub is a page with many out-links. The key idea of HITS is that a good hub points to many good authorities, and a good authority is pointed to by many good hubs. Thus, authorities and hubs have a mutual reinforcement relation. More specifically, a page's authority score is the sum of the hub scores of the pages it points to, and its hub score is the integration of authority scores of the pages pointed to by it. Using a power iteration method, the authority and hub scores of each page can be calculated. The main strength of HITS is ranking pages according to the query topic, which may provide more relevant authority and hub pages. However, HITS needs some time consuming operations, such as on-line expanding page sets and calculating the hub and authority scores.

5.2 Our HITS-Based Inference Model

5.2.1 Model Description Using the third level of the *TBHG* shown in Figure 4 as a case, Figure 8 illustrates the main idea of our HITS-based inference model. Here, a location is a cluster of stay points, like c_{31} and c_{32}. We regard an individual's visit to a location as an implicitly directed link from the individual to that location. For instance, cluster c_{31} contains two stay points respectively detected from u_1 and u_2's GPS traces, i.e., both u_1 and u_2 have visited this location. Thus, two directed links are generated respectively to point to c_{31} from u_1 and u_2. Similar to HITS, in our model, a hub is a user who has accessed many places, and an authority is a location which has been visited by many users. Therefore, users' travel experiences (hub scores) and the interests of locations (authority scores) have a mutual reinforcement relation.

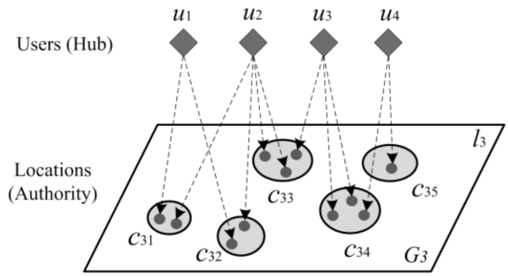


Figure 8. Our HITS-based inference model

5.2.2 Strategy for Data Selection Intrinsically, a user's travel experience is region-related, i.e., a user who has much travel knowledge in a city might have no idea about another city. Also, an individual, who has visited many places in a part of a city, might know little about another part of the city (if the city is very large, like New York). This feature is aligned with the query-dependent property of the HITS. Thus, before conducting the HITS-based inference, we need to specify a geospatial region (a topic query) for the inference model and formulate a dataset that contains the locations fallen in this region. However, using an online data selection strategy, (i.e., specify a region based on an individual's input), we need to perform lots of time consuming operations, which may reduce the feasibility of our system. Actually, on a level of the *TBHG*, the shape of a graph node (cluster of stay points) provides an implicit region for its descendent nodes. These regions covered by the clusters on different levels of the hierarchy might stand for various semantic meanings, such as a city, a district and a community. Therefore, we are able to calculate in advance the interest of every location using the regions specified by their ascendant clusters. In other words, a location might have multiple authority scores based on the different region scales it falls in. Also, a user might have multiple hub scores conditioned by the regions of different clusters.

Definition 8. Location Interest: In our system, the interest of a location (c_{ij}) is represented by a collection of authority scores $I_{ij} = \{I_{ij}^1, I_{ij}^2, \dots, I_{ij}^l\}$. Here, I_{ij}^l denotes the authority score of cluster c_{ij} conditioned by its ascendant nodes on level l, where $1 \leq l < i$.

Definition 9. User Travel Experience: In our system, a user's (e.g., u_k) travel experience is represented by a set of hub scores $e^k = \{e_{ij}^k | 1 \leq i < |L|, 1 \leq j \leq |C_i|\}$ (refer to definition 6), where e_{ij}^k denotes u_k's hub score conditioned by the region of c_{ij}.

Figure 9 demonstrates these definitions. In the region specified by cluster c_{11}, we can respectively calculate an authority score (I_{21}^1 and I_{22}^1) for cluster c_{21} and c_{22}. Meanwhile, within this region, we are able to infer authority scores (I_{31}^1, I_{32}^1, I_{33}^1, I_{34}^1 and I_{35}^1) for cluster $c_{31}, c_{32}, c_{33}, c_{34}$ and c_{35}. Further, using the region specified by cluster c_{21}, we can also calculate another authority score (I_{31}^2 and I_{32}^2) for c_{31} and c_{32}. Likewise, the authority scores (I_{33}^2, I_{34}^2 and I_{35}^2) of c_{33}, c_{34} and c_{35} can be re-inferred with the region of c_{22}. Therefore, each cluster on the third level has two authority scores, which would be used in various occasions based on users' inputs. For instance, as depicted in the Figure 9 A), when a user selects a region only covering location c_{31} and c_{32}, the authority score I_{31}^2 and I_{32}^2 can be used to rank these two locations. However, as illustrated in Figure 9 B), if the region selected by a user covers the locations from two different parent clusters (c_{21} and c_{22}), the authority value I_{32}^1, I_{33}^1 and I_{34}^1 should be used to rank these locations.

The strategy that sets multiple hub scores for a user and multiple authority scores for a location has two advantages. First, we are able to leverage the main strength of HITS to rank locations and users with the contexts of geospatial region (query topic). Second, these hub and authority scores can be calculated offline. Therefore, we can ensure the efficiency of our system while allowing users specify any regions on a map.

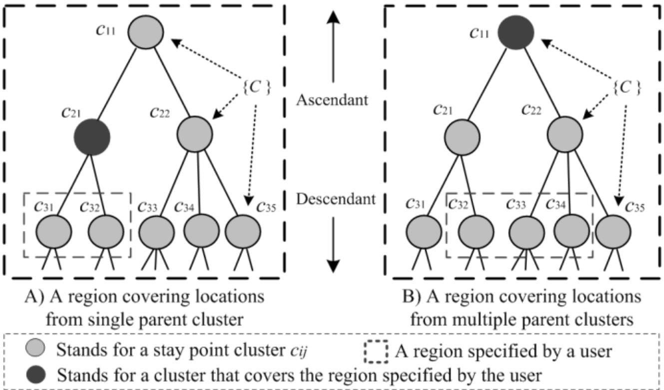


Figure 9. Some cases demonstrating the data selection strategy

5.2.3 Inference Given the locations pertaining to the same ascendant cluster, we are able to build an adjacent matrix *M* between users and locations based on the users' accesses on these locations. In this matrix, an item v_{ij}^{k} stands for the times that u_k (a user) has visited to cluster c_{ij} (the *j*th cluster on the *i*th level). Such matrixes can be built offline for each non-leaf node. For example, the matrix M formulated for the case shown in Figure 8 can be represented as follows, where all the five clusters pertain to c_{11}.

$$M = \begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \\ u_4 \end{array} \begin{array}{c} \begin{array}{ccccc} c_{31} & c_{32} & c_{33} & c_{34} & c_{35} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 2 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{array} \qquad (5)$$

Then, the mutual reinforcement relationship of user travel experience e_{ij}^{k} and location interest I_{ij}^{l} is represented as follows:

$$I_{ij}^{l} = \sum_{u_k \in U} e_{lq}^{k} \times v_{ij}^{k}; \qquad (6)$$

$$e_{lq}^{k} = \sum_{c_{ij} \in c_{lq}} v_{ij}^{k} \times I_{ij}^{l}; \qquad (7)$$

Where c_{lq} is c_{ij}'s ascendant node on the *l*th level, $1 \leq l < i$. For instance, as shown in Figure 9, c_{31}'s ascendant node on the first level of the hierarchy is c_{11}, and its ascendant node on the second level is c_{21}. Thus, if $l = 2$, c_{lq} stands for c_{21} and $(c_{31}, c_{32}) \in c_{21}$. Also, if $l = 1$, c_{lq} denotes c_{11}, $(c_{31}, c_{32}, \ldots, c_{35}) \in c_{11}$.

Writing them in the matrix form, we use $\boldsymbol{\mathcal{T}}$ to denote the column vector with all the authority scores, and use $\boldsymbol{E}$ to denote the column vector with all the hub scores. Conditioned by the region of cluster c_{11}, $\boldsymbol{\mathcal{T}}=(I_{31}^{1}, I_{32}^{1}, \ldots, I_{35}^{1})$, and $\boldsymbol{E}=(e_{11}^{1}, e_{11}^{2}, \ldots, e_{11}^{4})$.

$$\boldsymbol{\mathcal{T}} = \boldsymbol{M}^{\mathrm{T}} \cdot \boldsymbol{E} \qquad (8)$$

$$\boldsymbol{E} = \boldsymbol{M} \cdot \boldsymbol{\mathcal{T}} \qquad (9)$$

If we use $\boldsymbol{\mathcal{T}}_n$ and $\boldsymbol{E}_n$ to denote authority and hub scores at the *n*th iteration, the iterative processes for generating the final results are

$$\boldsymbol{\mathcal{T}}_n = \boldsymbol{M}^{\mathrm{T}} \cdot \boldsymbol{M} \cdot \boldsymbol{\mathcal{T}}_{n-1} \qquad (10)$$
$$\boldsymbol{E}_n = \boldsymbol{M} \cdot \boldsymbol{M}^{\mathrm{T}} \cdot \boldsymbol{E}_{n-1} \qquad (11)$$

Starting with $\boldsymbol{\mathcal{T}}_0 = \boldsymbol{E}_0 = (1,1,\ldots,1)$, we are able to calculate the authority and hub scores using the power iteration method.

Figure 10 depicts an off-line algorithm for inferring each user's hub scores and the authority scores of each location conditioned by the different regions. Here C_x is the collection of clusters on *x*th level. $C_x' \subset C_x$ denotes the collection of c_{ij}'s descendant clusters on the *x*th level. For instance, the C_2' of c_{11} is $\{c_{21}, c_{22}\}$, and C_3' of c_{11} is $\{c_{31}, c_{32}, \ldots, c_{35}\}$. $\{I_x^i\}$ represents the collection of authority scores of the locations contained in C_x conditioned by their ascendant node on the *i*th level.

5.3 Mining Classical Travel Sequences

With users' travel experiences and the interests of locations, we calculate a classical score for each location sequence within the given geospatial region. The classical score of a sequence is the integration of the following three aspects. 1) The sum of hub scores of the users who have taken this sequence. 2) The authority scores of the locations contained in this sequence. 3) These authority scores are weighted based on the probability that people would take a specific sequence.

Algorithm LocationInterestInference ($TBHG$**,** $LocH$**)**

Input: A *TBHG*=(*H*, *G*) and a collection of users' location histories *LocH*= $\{h^k | 1 \leq k \leq |U|\}$.
Output: the collection of users' hub scores $\boldsymbol{E}$, and the collection of locations' authority scores $\boldsymbol{\mathcal{T}}$.
$\boldsymbol{E}$=$\boldsymbol{\mathcal{T}} = \emptyset$;
For $i = 1; i < |L|; i++$ //for each level
    For $j = 1; j \leq |C_i|; j++$ // for each cluster on this level
        For $x = i+1; x \leq |L|; x++$ //search the descendant levels
            $C_x{}'$=LocationCollecting (x, c_{ij}, H);
            M=MatrixBuilding($C_x{}'$, $LocH$);
            ($\{e_{ij}^k\}$, $\{I_x^i\}$)=HITS-Inference(M);
            $\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{T}} \cup \{I_x^i\}$;
            $\boldsymbol{E} = \boldsymbol{E} \cup \{e_{ij}^k\}$;
Return ($\boldsymbol{E}, \boldsymbol{\mathcal{T}}$);

Figure 10. The algorithm for inferring the authority and hub scores

Using a graph of *TBHG*, Figure 11 demonstrates the calculation of the classical score for a 2-length sequence, A→C. In this figure, the graph nodes (A, B, C, D and E) stand for locations, and the graph edges denote people's transition sequences among them. The number shown on each edge represents the times users have taken the sequence. Equation (11) presents the classical score of sequence A→C, which includes the following three parts. 1) The authority score of location A (I_A) weighted by the probability of people's moving out by this sequence (Out_{AC}). Clearly, there are seven (5+2) links point out to other nodes from node A, and five out of seven of these links direct to node C. So, $Out_{AC} = \frac{5}{7}$, i.e., only five sevenths of location A's authority (I_A) should be offered to sequence A→C, and the rest of I_A should be provided to A→B. 2) The authority score of location C (I_C) weighted by the probability of people's moving in by this sequence (In_{AC}). 3) The hub scores of the users (U_{AC}) who have taken this sequence.

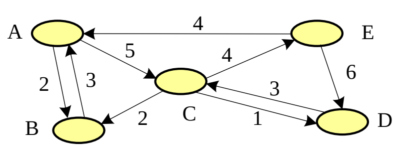


Figure 11. Demonstration on mining classical travel sequences from a graph

$$\begin{aligned} S_{AC} &= \sum_{u_k \in U_{AC}} (I_A \cdot Out_{AC} + I_C \cdot In_{AC} + e^k) \qquad (12) \\ &= |U_{AC}| \cdot (I_A \cdot Out_{AC} + I_C \cdot In_{AC}) + \sum_{u_k \in U_{AC}} e^k \\ &= 5 \times \left(\frac{5}{7} \times I_A + \frac{5}{8} I_C\right) + \sum_{u_k \in U_{AC}} e^k. \end{aligned}$$

Following this method, we calculate the classical score of sequence C→D,

$$S_{CD} = 1 \times \left(\frac{1}{7} \times I_C + \frac{1}{7} I_D\right) + \sum_{u_k \in U_{CD}} e^k. \qquad (13)$$

Thus, the classical score of sequence A→C→D equals to:

$$S_{ACD} = S_{AC} + S_{CD}. \qquad (14)$$

Using this paradigm we are able to calculate the classical score of any *n*-length sequences. Later, the top *m* *n*-length sequences with relatively high scores can be retrieved as *n*-length classical travel sequences. However, it is not necessary to find out the classical sequences with a long length, as people would not visit many places in a trip. Moreover, the process of searching for *n*-length classical sequences is time consuming, although this operation can be performed offline. Thus, in this paper, we start with mining 2-length classical sequences, and then try to find out some 3-lenth classical sequences by extending these 2-length sequences.

6. MINING LOCATION CORRELATION

In this section, we present the algorithm that computes the correlation between locations by considering the user travel experience and the sequence of the locations.

First, we claim that the correlation between two locations does not only depend on the number of users visiting the two locations but also lie in these users' travel experiences. The locations sequentially accessed by the people with more travel knowledge would be more correlated than those visited by those having little idea about the region. For instance, some overseas tourists might randomly visit some places in Beijing as they are not familiar with this city. However, the local people of Beijing are more capable than them of arranging a more proper and reasonable way to visit some places in Beijing.

Second, the correlation between two locations, *A* and *B*, also depends on the sequences, in which the two locations have been visited. 1) This correlation between *A* and *B*, $Cor(A, B)$, is asymmetric; i.e., $Cor(A, B) \neq Cor(B, A)$. The semantic meaning of a travel sequence *A*→*B* might be quite different from *B*→*A*. For example, on a one-way road, people would only go to *B* from *A* while never traveling to *A* from *B*. 2) The two locations continuously accessed by a user would be more correlated than those being visited discontinuously. Some users would reach *B* directly from *A* (*A*→*B*) while others would access another location *C* before arriving at *B* (*A*→*C*→*B*). Intuitively, the $Cor(A, B)$ indicated by the two sequences might be different. Likewise, in a sequence *A*→*C*→*B*, $Cor(A, C)$ would be greater than $Cor(A, B)$, as the user continuously accessed *A*→*C* while traveling to *B* after visiting *C*.

In short, the correlation between two locations can be calculated by integrating the travel experiences of the users visiting them in a trip in a weighted manner. Formally, the correlation between location *A* and *B* can be calculated as Equation 15.

$$Cor(A, B) = \sum_{u_k \in U'} \alpha \cdot e_k, \qquad (15)$$

where U' is the collection of users who have visited *A* and *B* in a trip; e_k is u_k's travel experience conditioned by the first shared ascendant regions (a cluster in *TBHG*) by the two locations, $u_k \in U'$. $0 < \alpha \leq 1$ is a dumping factor, which will decrease as the interval between these two locations' index in a trip increases. For example, in our experiment we set $\alpha = 2^{-(|j-i|-1)}$, where i and j are indices of *A* and *B* in the trip they pertain to. That is, the more discontinuously two locations being accessed by a user ($|i\text{-}j|$ would be big, thus α will become small), the less contribution the user can offer to the correlation between these two location.

As depicted in Figure 12, three users (u_1, u_2, u_3) respectively access locations (*A*, *B*, *C*) in different manners and create three trips $(Trip_1, Trip_2, Trip_3)$. The number shown below each node denotes the index of this node in the sequence. According to Equation (15), from $Trip_1$ we can calculate $Cor(A,B) = e_1$ and $Cor(B,C) = e_1$, since these locations have been consecutively accessed by u_1(i.e., $\alpha = 1$). However, $Cor(A,C) = \frac{1}{2} \cdot e_1$ (i.e., $\alpha = 2^{-(|2-0|-1)} = \frac{1}{2}$) as u_1 traveled to *B* before visiting *C*. In other words, the correlation (between location *A* and *C*) that we can sense from $Trip_1$ might not that strong as if they are consecutively visited by u_1. Likewise, we can learn $Cor(A,C) = e_2$, $Cor(C,B) = e_2$, $Cor(A,B) = \frac{1}{2} \cdot e_2$ from $Trip_2$, and infer $Cor(B,A) = e_3, Cor(A,C) = e_3, Cor(B,C) = \frac{1}{2} \cdot e_3$ from $Trip_3$. Later, we can integrate these correlation inferred from each user's trips and obtain the following results.

$$Cor(A,B) = e_1 + \frac{1}{2} \cdot e_2; \qquad Cor(A,C) = \frac{1}{2} \cdot e_1 + e_2 + e_3;$$

$$Cor(B,C) = e_1 + \frac{1}{2} \cdot e_3; \qquad Cor(C,B) = e_2; Cor(B,A) = e_3.$$

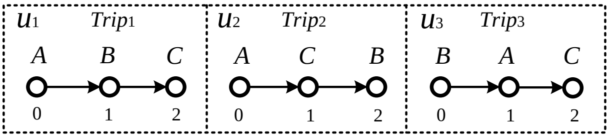


Figure 12. A case calculating the correlation between locations

Figure 13 formally describes the algorithm for inferring correlation between locations. Here, b is a constant, which is set to 2 in our experiment. $|Trip|$ stands for the number of locations contained in the $Trip$ and $Trip[i]$ represents the *i*th location in $Trip$. For example, regarding $Trip_1$ shown in Figure 12, $|Trip| = 3$, $Trip[0] = A$ (the first location), $Trip[1] = B$, $Cor(Trip[0], Trip[1]) = Cor(A,B)$. For the sake of simplification, we demonstrate the algorithm only using one layer of the hierarchy.

CalculateLocationCorrelation $(C, E, TP,)$

Input: A collection of locations (stay point clusters) *C*, A collection of users' travel experiences *E* and their location histories represented by trips *TP*.
Output: A matrix *Cor* describing the correlation between locations.

Foreach location $c_p \in C$ **Do**
  Foreach location $c_q \in C, p \neq q$ **Do**
    $Cor(c_p, c_q) = 0$; //initialize the location correlation
Foreach *Trip* **in** *TP* **Do**
  For $i = 0; i < |Trip|; i++$ //*i*th location contained in *Trip*
    For $j = i + 1; j < |Trip|; j++$
      $\alpha = b^{-(j-i-1)}$; // dumping factor, b is a constant
      $Cor(Trip[i], Trip[j]) += \alpha \cdot e_k$;
Foreach $c_p \in L$ **Do**
  Foreach $c_q \in L, p \neq q$ **Do** //normalization
    $Cor(c_p, c_q) = Cor(c_p, c_q) / \|Cor(c_p, c_0), \ldots, Cor(c_p, c_{|C|-1})\|_1$
Return *Cor*;

Figure 13. Algorithm learning the correlation between locations

Supposing we have n trips in a dataset and the average length of a trip is m, this mining algorithm takes $O(2|C|^2 + \frac{m(m-1)}{2} \cdot n)$ time. So, the overall computing complexity Q of our approach is the combination of inferring user travel experience and calculating location correlation, i.e., $Q = O(2w|C||U| + 2|C|^2 + \frac{m(m-1)}{2} \cdot n)$.

7. RECOMMENDATION

7.1 The Generic One

A user can specify any geospatial regions as an input by zoom in/out and panning a Web map. According to the zoom level, our recommender can find out the corresponding hierarchical level in the *TBHG*, and then collect the locations (clusters) fallen in the given region on this level. The hub and authority scores conditioned by the first shared ascendant cluster of these locations will be used to rank locations and users (refer to Figure 9). Later, the most k experienced users, top n interesting locations and top m classical travel sequences within the specified region can be returned to the users as the generic recommendations.

7.2 The Personalized One

7.2.1 Collaborative Filtering Collaborative filtering is a well-known model widely used in recommendation systems. The CF model can be partition into two categories; the user-based and item-based inference methods.

Notations: As shown in Equation (5), we have a matrix M describing the relation between each user and each location. Here, we can regard the times an individual has stayed in a location as their implicit ratings on the location. The ratings from a user u_p, called an *evaluation*, is represented as an array $R_p = \langle r_{p0}, r_{p1}, \dots, r_{pn} \rangle$, where r_{pj} is u_p's implicit ratings (the occurrences) in location j. $S(R_p)$ is the subset of the R_p, $\forall r_{pj} \in S(R_p)$, $r_{pj} \neq 0$, i.e., the set of items (locations) that has been rated (visited) by u_p. The average of ratings in R_p is denoted as $\overline{R_p}$, and the number of elements in a set S is $|S|$. The collection of all *evaluations* in the training set is $\mathcal{X}$. $S_j(\mathcal{X})$ means the set of evaluations containing item j, $\forall R_p \in S_j(\mathcal{X})$, $j \in S(R_p)$. Likewise, $S_{i,j}(\mathcal{X})$ is the set of evaluations simultaneously containing item i and j.

1) ***The Pearson correlation-based CF***. The Pearson correlation reference scheme [Adomavicius and Tuzhhilin 2005] is the most popular and accurate user-based CF model using the similarity between users, $sim(u_p, u_q)$, to weight the ratings from different individuals. Equation (16) and (17) give a formal description on calculating $P(r_{pj})$, the predicted u_p's ratings on location j. As the number of users in a system is much larger and increases much faster than the number of items, the user-based CF models are not that efficient than the item-based methods.

$$sim(u_p, u_q) = \frac{\sum_{i \in S(R_p) \cap S(R_q)} (r_{pi} - \overline{R_p}) \cdot (r_{qi} - \overline{R_q})}{\sqrt{\sum_{j \in S(R_p) \cap S(R_q)} (r_{pj} - \overline{R_p})^2 \cdot \sum_{j \in S(R_p) \cap S(R_q)} (r_{qj} - \overline{R_q})^2}} \quad (16)$$

$$P(r_{pj}) = \overline{R_p} + \frac{\sum_{R_q \in S_j(\mathcal{X})} sim(u_p, u_q) \times (r_{qj} - \overline{R_q})}{\sum_{R_q \in S_j(\mathcal{X})} sim(u_p, u_q)}; \tag{17}$$

2) ***The Slope One algorithms*** [Lemire and Maclachlan 2005] are famous and representative item-based CF algorithms, which are easy to implement, efficient to query and reasonably accurate. Given any two items i and j with ratings r_{pj} and r_{pi} respectively in some user evaluation $R_p \in S_{j,i}(\mathcal{X})$, we consider the average deviation of item i with regard to item j as Equation (18).

$$dev_{j,i} = \sum_{R_p \in S_{j,i}(\mathcal{X})} \frac{r_{pj} - r_{pi}}{|S_{j,i}(\mathcal{X})|}, \tag{18}$$

Given that $dev_{j,i} + r_{pi}$ is a prediction for r_{pj} based on r_{pi}, a reasonable predictor might be the average of all the predictions.

$$P(r_{pj}) = \frac{1}{|\mathcal{W}_j|} \sum_{i \in \mathcal{W}_j} (dev_{j,i} + r_{pi}), \tag{19}$$

where $\mathcal{W}_j = \{i | i \in S(R_p), i \neq j, |S_{j,i}(\mathcal{X})| > 0\}$ is the set of all relevant items. Further, the number of evaluations simultaneously contain two items has been used to weight the prediction regarding different items. Intuitively, to predict u_p's rating of item *A* given u_p's ratings of item *B* and *C*, if 2000 users rated the pair of *A* and *B* whereas only 20 users rated pair of *A* and *C*, then u_p's ratings of item *B* is likely to be a far better predictor for item *A* than u_p's ratings of item *C* is.

$$P(r_{pj}) = \frac{\sum_{i \in S(R_p) \wedge i \neq j} (dev_{j,i} + r_{pi}) \cdot |S_{j,i}(\mathcal{X})|}{\sum_{i \in S(R_p) \wedge i \neq j} |S_{j,i}(\mathcal{X})|}. \tag{20}$$

7.2.2 Our Method We integrate the location correlation into the Slop One algorithm to achieve a more effective and accurate item-based CF model. Intuitively, to predict u_p's rating of location *A* given u_p's ratings of location *B* and *C*, if location *B* is more related to *A* beyond *C*, then u_p's ratings of location *B* is likely to be a far better predictor for location *A* than u_p's ratings of location *C* is. In contrast to the number of observed ratings (i.e., the number of people who have visited two locations) used by the weighted Slope One algorithm, the location correlation mined from multiple users' location histories considers more human travel behavior, such as the travel sequence, user experience, and transition probability between locations. Formally, our approach can be represented as

$$P(r_{pj}) = \frac{\sum_{i \in S(R_p) \wedge i \neq j} (dev_{j,i} + r_{pi}) \cdot cor_{ji}}{\sum_{i \in S(R_p) \wedge i \neq j} cor_{ji}}, \tag{21}$$

where cor_{ji} denotes the correlation between location i and j, and $dev_{j,i}$ is still calculated as Equation (18). Using Equation (20), we can predict an individual's ratings on the locations they have not accessed, and then rank these locations in terms of the predicted ratings. Later, the top *n* locations with relatively high ratings can be recommended to the individual.

8. EXPERIMENTS

In this Section, we first present the experimental settings. Second, we introduce the evaluation approaches. Third, major results are reported followed by some discussions.

8.1 Settings

8.1.1 Devices and Users Figure 14 shows the GPS devices we chose to collect data. They are comprised of stand-alone GPS receivers (Magellan Explorist 210/300, G-Rays 2 and QSTARZ BTQ-1000P) and GPS phones. Except for the Magellan 210/300, these devices are set to receive GPS coordinates every two seconds. Regarding the Magellan devices, we configure their settings to record GPS points as densely as possible because they are not allowed to be configured for recording data by fixed time interval. When an individual changes his/her heading direction or speed to some extent, a GPS point is recorded with such devices. Carrying these GPS-enabled devices, 107 users (49 females and 58 males) recorded their outdoor movements with GPS logs from May 2007 to Oct. 2008. Figure 15 presents demographic statistics on these users.

Figure 14. GPS devices used in our experiment

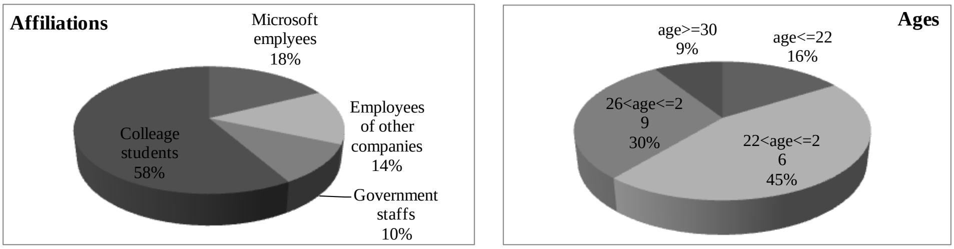
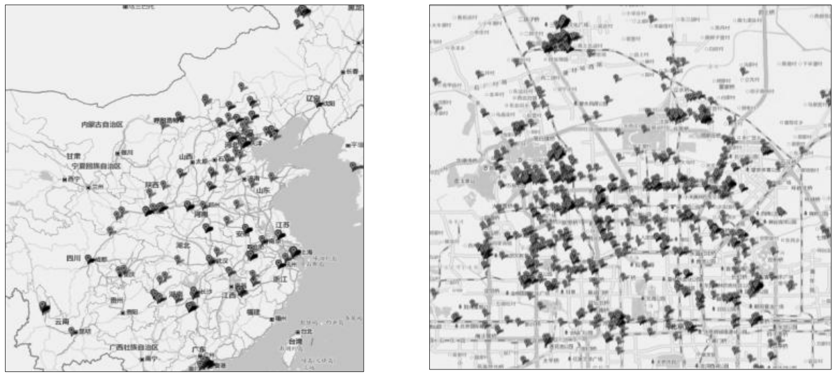


Figure 15. Demographic statistics of our experiment

8.1.2 GPS Data Figure 16 depicts the distributions of the GPS data used in the experiments. Most parts of this dataset were created in Beijing, China, and other parts covered 36 cities in China as well as a few cities in the USA, South Korea, and Japan. The volunteers were motivated to log their outdoor movements as much as possible by the payments based on the distance of GPS traces collected by them; the more data collected by them, the more money they obtained. As a result, the total distance of the GPS logs exceeded 166,372 kilometers, and the total number of GPS points is over 5 million. Considering the privacy issues, we use these datasets anonymously.



A) Data distribution in China B) Data distribution in Beijing

Figure. 16 Distribution of the GPS dataset we used in this experiment

8.1.3 Parameter Selection

Stay point detection: We set T_r to 20 minutes and D_r to 200 meters for stay point detection. In other words, if an individual stays over 20 minutes within a distance of 200 meters, a stay point is detected. These two parameters enable us to find out some significant places, such as restaurants and shopping malls, while ignoring the geo-regions without semantic meanings, like the places where people wait for traffic lights or meet congestion (refer to [Li et al. 2008] for details). As a result, we extracted 10,354 stay points from the dataset.

Clustering: We use a density-based clustering algorithm, OPTICS (Ordering Points To Identify the Clustering Structure), to hierarchically cluster stay-points into geospatial regions in a divisive manner. As compared to an agglomerative method like K-Means, the density-based approach is capable of detecting clusters with irregular structures, which may stand for a set of nearby restaurants or shopping streets. In addition, this approach would filter out a few sparsely distributed stay points, and ensure each cluster has been accessed by some users. As a result, a four-level *TBHG* is built based on our dataset (see Table I for details).

Table I. Information of the *TBHG* used in the experiment

Level	Num. of Clusters	Average size of clusters KM	Average num of user/cluster	Average num stay points/cluster
1	1	11,450.7	107	10,354
2	32	14.5	6.7	267.5
3	70	2.1	8	112.7
4	159	0.26	6.5	46.2

Trip partition: Based on the commonsense knowledge, we set T_p=15 hours and obtain 5,318 trips (the average length of these trips is 3.2).

8.2 Evaluation Approaches

8.2.1 Evaluation Framework Figure 17 illustrates the framework of the evaluation, in which we respectively explore the effectiveness of the generic and personalized recommendations by performing a user study. In this study, 29 subjects (14 females and 15 males), who have been in Beijing for more than 6 years, were invited to answer the evaluation questions. At the same time, all of them have an 3-month+ GPS trace set accumulated in our system. Given the region within the fourth ring road of Beijing, we respectively retrieved the top 10 interesting locations, top 5 classical travel sequences and top 10 personalized locations by using our methods and some baselines.

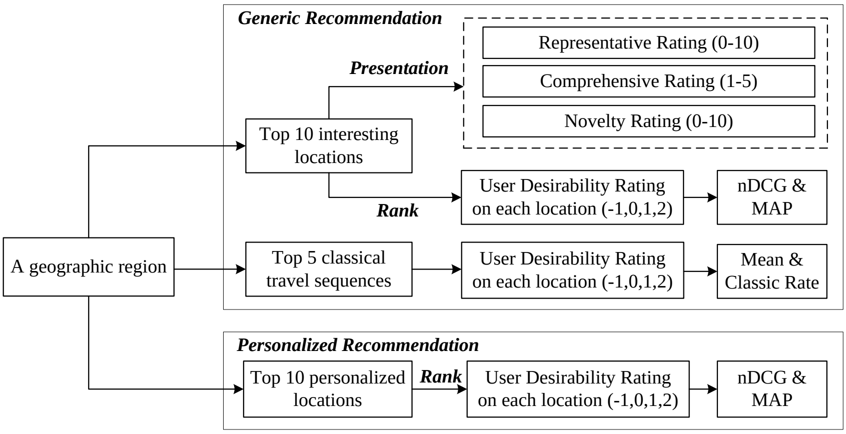


Figure 17. Framework of the evaluation

Regarding the interesting locations from the generic recommendation, we conduct the following two aspects of evaluations. One is the ***Presentation***, which stands for the ability of the retrieved interesting locations in presenting a given region. The other is the ***Rank***, which represents the ranking performance of the retrieved locations based on relative interests.

1) Presentation. Each subject answers the following evaluation questions:

- *Representative*: How many locations in this retrieved set are representative of the given region (0-10)?
- *Comprehensive*: Do these locations offer a comprehensive view of the given region (1-5)?
- *Novelty:* How many locations in this retrieved set have interested you even though they only appeared recently (0-10)? In the study, the subjects were able to view the points of interests (POIs) fallen in each location as well as the photos taken there.

2) Rank. Each subject had to individually rate the interest of each retrieved location with a value (-1~2) shown in table II. Then, we aggregated these subjects' ratings for each location, and select the mode of the ratings for the location. If the mode of two rating levels is identical, we prefer the lower ratings.

Table II. Users' interests in a location

Ratings	Explanations
2	I'd like to plan a trip to that location.
1	I'd like to visit that location if passing by.
0	I have no feeling about this location, but don't oppose others to visit it.
-1	This location does not deserve to visit.

With regard to evaluating the retrieved classical travel sequences, we required the subjects to rate each sequence in the set with the scores shown in table III. Also, we aggregate these ratings as the method mentioned above.

Table III. Users' interests in a travel sequence

Ratings	**Explanations**
2	I'd like to plan a trip with this travel sequence.
1	I'd like to take that sequence if visiting the region.
0	I have no feeling about this sequence, but don't oppose others to choose it.
-1	It is not a good choice to select this sequence.

Different from evaluating the top *n* interesting locations (first aggregate the ratings of different users and then study the ranking performance of the aggregated result), we first respectively calculate the ranking performance of the top 10 personalized locations retrieved for each user (a user's rating on a personalized location is also based on Table II) and then aggregate ranking performance of different users.

8.2.2 Measurements

Measurements for presentation: We compare our method with the baselines using the mean score of the ratings offered by the subjects. In addition, we perform a T-test for each comparison to justify the significant advantages of our method.

Measurements for ranking: We employ two criteria, *nDCG* (normalized discounted cumulative gain) and *MAP* (Mean Average Precision), to measure the ranking performance of the retrieved interesting locations. *MAP* is the most frequently used summary measure of a ranked retrieval run. In our experiment, it stands for the mean of the precision score after each interesting location is retrieved. Here, a location is deemed as an interesting location if its interest level equals to 2. For instance, the *MAP* of an interest rating vector, $G = < 2, 0, 2, 0, 1, 0, 0, 2, 0, -1 >$, for the top 10 location, is computed as follows.

$$MAP = \frac{1+2/3+3/8}{3} = 0.681.$$

nDCG is used to compute the relative-to-the-ideal performance of information retrieval techniques. The discounted cumulative gain of *G* is computed as follows: (In our experiments, b = 3.)

$$CG[i] = \begin{cases} G[1], & if\ i = 1 \\ DCG[i-1] + G[i], & if\ i < b \\ DCG[i-1] + \dfrac{G[i]}{log_b i}, & if\ i \geq b \end{cases}$$

Given the ideal discounted cumulative gain *DCG'*, then *nDCG* at *i*-th position can be computed as $NDCG[i] = DCG[i]/DCG'[i]$.

Measurement for classical sequence: We use the mean score of these subjects' ratings, along with a T-test for each comparison, to distinguish our method from baselines. At the same time, we investigated the *classical rate*, which represents the ratio of sequences with a score of 2 in the set, of different methods.

8.2.3 Baselines

Baselines for mining interesting locations: Here, we explore the effectiveness of two baseline methods, *rank-by-count* and *rank-by-frequency*. Regarding the former one, the more users visiting a location the more interesting this location might be. In the latter, the more frequent people accessed a location the more interesting this location might be. The visited frequency of a location is the ratio between the number of the users visiting this location and the time span, from the first day one user accessed this location to the last day at least one individual visited it.

Baselines for mining classical travel sequences: We compare our method with three baselines; *rank-by-count*, *rank-by-interest* and *rank-by-experience*. With regard to the first baseline, we rank a sequence based on the number of the users who have taken this sequence. Regarding the second one, we only take into account the interests of the locations contained in a sequence to rank the travel sequences. In the third baseline method, we only consider the experiences of the users who have taken this sequence.

Baseline for the personalized recommendation: We respectively investigate the performance of three baseline schemes: 1) Our approach only using user travel experience, i.e., each pair of locations occurring in a trip share the same correlation. 2) Our method only considering the sequence between locations, i.e., all users has the same travel experiences. 3) the Pearson Correlation-based approach described in Section 7.2.1.

8.3 Results

8.3.1 Related to Interesting Locations

Presentation ability: Figure 18 illustrates the top 10 interesting locations, which were respectively inferred out by our method and two baselines using the region within the fourth ring road of Beijing (the zoom level corresponds to the 3rd level of the *TBHG*).

A) Our method B) *Rank-by-count* C) *Rank-by-frequency*

Figure 18. Top 10 interesting locations of different approaches

Based on these results, 29 subjects individually answered the evaluation questions with the ratings mentioned in Table II. As shown in Table IV, our method is more capable than the baselines of finding out representative locations in the give region (T-test result: $p_1<0.01$, the comparison between ours and the *Rank-by-count*; $p_2<0.01$, the comparison between ours and *Rank-by-frequency*). Meanwhile, the top 10 locations retrieved by our method presented a more comprehensive view of this region over the baselines ($p_1<<0.01$, $p_2<<0.01$). In addition, using our method, more novel locations that interest the subjects have been retrieved ($p_1<0.01$, $p_2<0.01$). These regions represent the development of new Beijing, while having not been noticed by many people. Regarding the baselines, *Rank-by-count* outperformed *rank-by-frequency* in finding out the representative locations ($p<0.01$) and presenting a comprehensive view of the region ($p<0.01$). However, the former method does not show a clear advantage beyond the latter in detecting the novel interesting locations ($p>0.2$).

Table IV. Comparison on the presentation ability of different methods

	Ours	***Rank-by-count***	***Rank-by-frequency***
Representative	**5.4**	4.5	3.1
Comprehensive	**4**	3.4	2.3
Novelty	**3.4**	2.4	2.2

Ranking ability: Table V depicts the ranking ability of different methods using *nDCG@*5, *nDCG@*10 and *MAP* as measurements. Although the set of interesting locations retrieved by our method and *rank-by-count* had a 60 percents overlap, our method showed clear advantages beyond baseline methods in effectively ranking this location set.

Table V. Ranking ability of different methods

	Ours	***Rank-by-count***	***Rank-by-frequency***
*nDCG@*5	**0.823**	0.714	0.598
*nDCG@*10	**0.943**	0.848	0.859
MAP	**0.759**	0.532	0.365

8.3.2 Results Related to Classical Sequences Using two measurements (mean score and classical rate), Table VI distinguishes the performance of our method from the baselines in finding out the classical sequences in the given region. Clearly, our method considering both users' travel experiences and location interests outperforms *rank-by-count* ($p<<0.01$), *rank-by-interest* ($p<0.01$) and *rank-by-experience* ($p<0.01$). Meanwhile, when respectively taking into account users' travel experiences ($p<0.01$) or location interests ($p<0.01$), the performance of *rank-by-count* had been significantly improved. These results proved that user travel experience and location interests respectively play an important role in retrieving the classical travel sequences and offered a greater contribution when being used together. (See 8.2.2 for the meaning of *classical rate*)

Table VI. Performance of different methods in finding classical sequences

	Ours (Interest + Experience)	***Rank-by-count***	***Rank-by-interest***	***Rank-by-experience***
Mean score	**1.6**	1.2	1.4	1.5
Classical Rate	**0.6**	0.3	0.4	0.4

8.3.2 Related to Personalized Recommendation

Effectiveness: Using the average *NDCG* and *MAP*, Table VII compares the effectiveness of different methods in conducting the personalized location recommendation. Clearly, our approach (*Experience* + *Sequence*) outperforms the weighted Slope One algorithm (T-Test of *NDCG@5*, p=0.0053<0.01; T-Test of *MAP*, p=0.0049<0.01). Although our method is slightly weaker than the Pearson correlation-based CF model in terms of the average *NDCG* and *MAP*, the T-test result (*NDCG@5*, p=0.678>>0.01; *MAP*, p=0.741>>0.01) shows that the advantage of the Pearson correlation is not significant. In short, some users thought the recommendation generated by our method is even better than that of the Pearson correlation-based scheme. Thus, we can claim that at least our method is as effective as the Pearson correlation-based one.

Table VII. Ranking performance of different methods (personalized recommendation)

	Ours	**The Pearson Correlation-Based CF model**	**The Weighted Slope One Algorithm**
NDCG@5	**0.840**	0.862	0.762
NDCG@10	**0.922**	0.938	0.891
MAP	**0.798**	0.804	0.665

Efficiency: Suppose we have such a GPS dataset generated by T users. From this dataset, we discover k locations and n trips; the average length (number of locations) of a trip is m. Thus, to predict a user's interest level in a location, the upper bound of computing complexity (times) of different methods are as follows:

The Pearson correlation-based CF model: $O(k \times (T-1)^2)$;

The Weighted Slope One algorithm: $O(T \times k(k-1))$;

Our method (*Exp* + *Seq*): $O(T \times k(k-1) + Q)$,

where $Q = 2wkT + 2k^2 + m(m-1)n/2$ is the total computing complexity of inferring the location correlation, and w is the iteration times.

Using the given GPS dataset, Figure 19 depicts the upper bound of computing complexity of different methods in calculating a prediction. Clearly, our method is much more efficient than the Pearson correlation-based CF model, while being slightly slower than the weighted Slope One algorithm. In short, our algorithm is as effective as the Pearson correlation-based model and almost as efficient as the weighted Slope one algorithm. Alternative, we can say our method is more efficient than the Pearson correlation-based model and more effective than the Weighted Slope One algorithm.

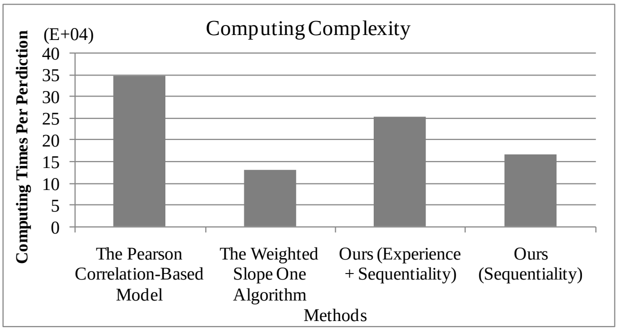


Figure 19. Average computing complexity in computing a prediction

8.4 Discussions

8.4.1 Discussion on Human Location History Beyond the static POI/YP dataset, people's location histories can provide us with richer knowledge of geographical spaces.

First, from the location history we are able to discover some places, which attract multiple users' interests, using a data-driven approach. Thus, 1) it is not necessary to manually pre-define some locations; 2) the detected locations would be more reasonable to be recommended to others; 3) we can find out the geo-regions with irregular structures, such as a shopping street and a lake; 4) the places developed recently can be automatically discovered.

Second, from the location history, we can discover the correlated locations pertaining to different business categories. For example, our method can detect that a restaurant is correlated with a cinema, or a lake and a museum are highly correlated.

Third, the location history implies some key factors, such as the travel time, distance, reachability and sequentiality between locations, which should be taken into account to plan a trip or perform a travel recommendation. For example, if two locations A and *B* co-occurred in multiple users' trips, at least we can guarantee these two locations are reachable. Further, if people always travel to location *B* from *A*, it might imply that there would be a one-way road between these two locations. Meanwhile, people prefer to travel to a shopping mall nearby them rather than a distant one unless the quality of the distant one deserves a relatively long travel.

Given the above-mentioned reasons, we believe that human location history is a much better data source than others, like POI/YP datasets, in revealing the correlation between locations.

8.4.2 Discussion on Interesting Locations With data shown in Table IV, we observe that users' travel experiences are useful in not only retrieving representative locations in a region but also finding out more novel and interesting locations beyond baseline methods. Intuitively, some interesting places, which contain high-quality restaurants or nice shopping malls developed recently, would not be visited by many people. However, a location covering some landmarks, which is not that interesting but with a relatively long history, might be accessed by more people. Hence, the *rank-by-count* cannot handle this kind of problem well. Meanwhile, a user would frequently access the restaurant nearby their working place for convenience rather than food quality or having fun. Therefore, a location frequently visited by people might not be interesting.

8.4.3 Discussion on Classical Travel Sequences The results shown in Table VI justify the contributions of users' travel experiences and location interests in mining classical travel sequences. First, intuitively, without considering such information, the sequence from a railway station to a nearby hotel might be detected as a classical travel sequence because some tourists usually stay in the hotels nearby the station. Obviously, this is not a good recommendation for users. Second, if only using individuals' travel experiences, we would mine out some life routine of an experienced user rather than classical sequences. For instance, sometimes, an experienced user would have dinner at a restaurant nearby their home and then go to a supermarket not far away from this restaurant. Since the user has a relatively high hub score, their life routine, like from the restaurant to the supermarket, might be detected as a classical travel sequence. Third, if only considering location interest, some impractical sequences would be found out. For example, the Summer Palace and the Forbidden City are two very interesting locations in Beijing. An experienced user would not visit them in a sequence as they are far from each other and each deserves a one day tour. However, a few tourists without much travel knowledge about Beijing might carelessly visit these two places in a sequence, hence make this sequence classical.

8.4.3 Discussion on Location Correlation 1) *User Travel Experience*: As shown in Table IV, our methods considering a user's travel experience is more capable than the Pearson correlation of predicting the correlation between locations. Intuitively, if we do not differentiate the experiences of various users, the locations randomly visited by some tourists without much knowledge about the given geo-region would also become correlated. Thus, the recommended locations might not be that interesting as if they are generated from some experienced users' location histories. With a user's travel experience, we can also reduce to some extent the cold start problem in the existing recommendation systems, where a location would not be recommended until this location has been rated (accessed) by many people. In our method, if a place has been visited by some experienced users, the place would become correlated with other locations, hence might be retrieved as a recommendation. So, people are more likely to discover some newly developed landmarks or shopping malls.

2) *Sequentiality*: As depicted in Table IV and Tabel V, our methods considering the sequentiality feature have shown clear advantages beyond the Person correlation-based mobile tour guide and the weighted slope one algorithm. At the first glance, people would argue that sometimes the locations accessed by an individual in a trip might share the same degree of correlation among each other. For example, *A*, *B*, *C* are three similar shopping malls. The perfect inference result should be $Cor(A, B) = Cor(A, C) = Cor(B, C)$. However, an individual would access these locations in a sequence of $A \rightarrow B \rightarrow C$. According to our approach weighting the user travel experience by the sequentiality the locations has been visited, $Cor(A, B) = Cor(B, C) > Cor(A, C)$. This looks not right. But, remember we have many users' location histories; if these locations really share the similar degree of correlation, different users would access them in a variety of sequences, such as $A \rightarrow C \rightarrow B$ and $B \rightarrow A \rightarrow C$. Therefore, the finally integrated results would be correct. On the contrary, if people always travel to another three places *D*, *E*, *F* in a sequence of $D \rightarrow E \rightarrow F$ there must be some reason behind the phenomena; that is the different degree of correlation between locations.

9. CONCLUSION

In this article, we learn the generic and personalized travel recommendations from a large number of user-generated GPS traces. In the generic recommendation, we model multiple users' location histories with *TBHG*, and mine the top *n* interesting locations and the top *m* classical travel sequences in a given geospatial region based on the *TBHG* and a HITS-based inference model. This HITS-based infers the interest of a location and a user's travel experience by taking into account 1) the mutual-reinforcement relation between the two values and 2) the geo-region conditions. Considering these two inferred values, we mine the classical travel sequences among locations. To achieve the personalized recommendation, we first calculate the correlation between locations by employing the user travel experiences and the sequence that locations have been visited. Then, we incorporate this correlation into an item-based CF model, which predict a user's interest in an unvisited location in terms of the user's location history and that of others.

To evaluate these two types of recommendations, we perform a user study based on a real-world GPS trace dataset collected by 107 users over a period of one year. In this study, we invite 29 users living in Beijing for more than 6 years and request them to rate the recommendations generated by different methods. As a result, our method showed clear advantages beyond *rank-by-count* and *rank-by-frequency* by providing a better presentation ability and ranking performance. Meanwhile, when employing both users' travel experiences and location interests, we achieved the best performance in detecting classical travel sequences. Regarding the personalized location recommendation, our approach is more effective than the weighted Slope one algorithm with a slightly additional computation. In addition, in contrast to the Pearson correlation-based CF model, our method is much more efficient while keeping the similar effectiveness.

REFERENCES

ABOWD, G. D., ATKESON, C. G., HONG, J., LONG, S., KOOPER, R., PINKERTON, M. 1997. Cyberguide: a mobile context-aware tour guide, *wireless network*, 3(5), 421-433.

ADOMAVICIUS, G. AND TUZHHILIN, A. 2005. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE *Trans. on Knowledge and Data Engineering*. 17(6), 734-749.

ASHBROOK, D., AND STARNER, T. 2003. Using GPS to learn significant locations and predict movement across multiple users. *Personal Ubiquitous Computing*. 7(5), 275–286.

BEEHAREE, A. AND STEED, A. 2007. Exploiting real world knowledge in ubiquitous applications. *Personal and Ubiquitous Computing*, 11(6), 429-437.

Bikely 2006: http://www.bikely.com/.

CHEN, G., AND KOTZ, D. 2000. A survey of context-aware mobile computing research. Tech. rep. TR2000-381, 2000, Dartmouth College Hanover, NH.

CHEN, Z., SHEN, H. T., ZHOU, X., ZHENG, Y. AND XIE, X. 2010. Searching Trajectories by Locations – An Efficiency Study, In *Proceedings of ACM SIGMOD International Conference on Management of Data*. ACM Press.

COUNTS, S. AND SMITH M. 2007. Where were we: Communities for sharing space-time trails. In *Proceedings of the 15th International Symposium on Advances in Geographic Information Systems*, Samet H., Shahabi C. and Schneider M., Eds. New York. 10–18.

EAGLE, N., AND PENTLAND, A. 2006. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4), 255-268.

GEOLIFE, 2007: http://research.microsoft.com/en-us/projects/geolife/.

GONOTTI, F., NANNI, M., PINELLI, F., AND PEDRESCHI, D. 2007. Trajectory pattern mining. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ACM Press, 330-339.

HARIHARAN, R. AND TOYAMA, K. 2004. Parsing and modeling location histories In *Proceedings of the 3rd International Conference on Geographic Information Science*. Springer, 106–124.

HOROZOV, T., NARASIMHAN, N., AND VASUDEVAN, V. 2006. Using Location for Personalized POI Recommendations in Mobile Environments. In *Proceedings of the International Symposium on Applications on Internet*, IEEF Press, 124-129.

KRUMM, J., AND HORVITZ, E. 2006. Predestination: Inferring destinations from partial trajectories. In *Proceedings of the 8th International Conference on Ubiquitous Computing*. Springer-Verlag 243–260.

LEMIRE D. AND MACLACHLAN A. 2005. Slope One: Predictors for Online Rating-Based Collaborative Filtering. In Proceedings of SIAM Data Mining.

LI, Q., ZHENG Y., XIE X., CHEN Y., LIU W. AND MA W. 2008. Mining user similarity based on location history. IN *Proc. of the 16th International Conference on Advances in geographic information system*, ACM Press: 1-10.

LIAO, L., PATTERSON, D. J., FOX, D., AND KAUTZ, H. 2005. Building personal maps from GPS data. *Annals N.Y. Acad. Sci. 1093*, 249–265.

LIAO, L., FOX, D., AND KAUTZ, H. 2004. Learning and Inferring Transportation Routines. In *Proceedings of the National Conference on Artificial Intelligence*. AAAI Press, 348–353.

LINDEN, G., SMITH, B. AND YORK, J. 2003. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. IEEE Internet Computing, 7, 1, 76-80.

MAMOULIS, N., CAO, H., KOLLIOS, G., HADJIELEFTHERIOU, M., TAO, Y., AND CHEUNG, D. W. 2004. Mining, Indexing and Querying Historical Spatiotemporal Data. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ACM Press, ACM Press, 236-245.

PARK, M. H, Hong, J. H., AND CHO, S. B. 2007. Location-Based Recommendation System Using Bayesian User's Preference Model in Mobile Devices. In *Proceedings of Ubiquitous Intelligence and Computing*. Springer Press, 1130-1139.

PATTERSON, D., LIAO, L., FOX, D., AND KAUTZ, H. 2003. Inferring high-level behavior from low-level sensors. In *Proceedings of the 8th International Conference on Ubiquitous Computing*. Springer,73–89.

SIMON, R., AND FRÖHLICH, P. 2007. A Mobile Application Framework for the Geospatial Web. IN *Proceeding of the 16th International Conference on World Wide Web*, ACM Press. ACM Press: 381-390.

SPORTSDO. 2007. http://sportsdo.net/Activity/ActivityBlog.aspx.

TAKEUCHI, Y. AND SUGIMOTO, M. 2006. CityVoyager: An Outdoor Recommendation System Based on User Location History. In *Proceedings of Ubiquitous Intelligence and Computing*, Springer Press: 625-636.

ZHENG, Y., LI, Q., CHEN, Y., XIE, X., AND MA, W. Y. 2008a. Understand mobility based GPS data. *In Proceedings of the 10th International Conference on Ubiquitous Computing*. ACM Press, 312–321.

ZHENG, Y., CHEN, Y., LI, Q., XIE, X., AND MA, W. Y. 2010a, Understanding Transportation Modes Based on GPS Data for Web Applications. ACM Transaction on the Web, 4(1), 1-36.

ZHENG, Y., LIU, L.,WANG, L., AND XIE, X. 2008b. Learning transportation mode from raw GPS data for geographic applications on the Web. In *Proceedings of the 11th International Conference on World Wide Web*. ACM Press, 247–256.

ZHENG, Y., CHEN,, XIE, X., AND MA, W. Y. 2009a. GoLife2.0: A Location-Based Social Networking Service. In *Proceedings of International Conference on Mobile Data Management 2009*. IEEE Press, 357–358.

ZHENG, Y.,WANG, L., ZHANG, R., XIE, X., AND MA,W. Y. 2008c. GeoLife: Managing and understanding your past life over maps. In *Proceedings of the 9th International Conference on Mobile Data Management*. IEEE Press, 211–212.

ZHENG, Y., XIE, X., AND MA, W. Y. 2008d. Search your life over maps. In *Proceedings of the International Workshop on Mobile Information Retrieval*. 24–27.

ZHENG Y., ZHANG L., XIE X. AND MA W. Y. 2009b. Mining interesting locations and travel sequences from GPS trajectories. In *Proceeding of the 18th International Conference on World Wide Web*, ACM Press, 791-800.

ZHENG, Y., ZHANG, L., MA, Z., XIE, X., MA, W.Y. 2010b. Recommending friends and locations based on individual location history. To appear in ACM Transaction on the Web.

ZHENG, W. C., ZHENG, Y., XIE, X. AND YANG, Q. 2010c. Collaborative location and activity Recommendations with GPS History Data. In *Proceeding of the 19th International Conference on World Wide Web*, ACM Press, 791-800.

Received May 2007; accepted July 2007