

Successive Point-of-Interest Recommendation in Intelligent Business Area

Rui Yang^{1, a}, Run Zhao^{2, b} and Dong Wang^{2, c}

¹School of Software, Shanghai Jiao Tong University, Shanghai 200240, China;

² Computer Science Department, Shanghai Jiao Tong University, Shanghai 200240, China.

^a5yyx@sjtu.edu.cn, ^b zhaorun@cs.sjtu.edu.cn, ^cwangdong@cs.sjtu.edu.cn

Keywords: Point-of-Interest, Intelligent Business Area, Recommendation.

Abstract. With the development of location-acquisition and wireless communication technologies, people's behavior in Business Area can be recorded. Therefore a class of location-based services, in a particular intelligent business area, based on the user's current location and historical information, providing POI recommendation, has increasingly majority needs of users. This paper presents a method to solve this problem

Introduction

Recently, with the development of location-acquisition and wireless communication technologies, such as GPS and Wi-Fi technology, people's behavior in Business Area can be recorded. People can share their location, and the locations can be used by many systems, these systems improve the user ability to get the information content with current location, time, conditions and personal characteristics that match the requirements, and therefore a class of location-based services to recommend POI has increasingly majority needs of users.

Such service can be described as follows: in a particular intelligent business area, based on the user's current location and historical information, providing POI recommendation. Such services with a wide range of usage scenarios in real life, as shown in Fig. 1, a user needs to determine entertainment location (such as KTV, etc.) after the shopping and dining and other activities, then the user needs based on the current location and its historical behavior to obtain the recommended location nearby.



Fig. 1 Scene schematic

The target of this paper is to solve this problem, and provide this service.

Related Work

The existing location recommendations can be summarized into two categories: 1) generic location recommendations and 2) personalized recommendations [1].

Generic location recommendations. Regardless of the users' preferences, generic location recommendation systems provide people with the most popular venues or travel routes in a city. For example, [2] mines the most interesting locations and travel sequences from a large number of user-generated GPS trajectories. Given a user-location matrix, a HITS-based inference model was also proposed to predict the interest level of a physical location and the knowledge of a user.

Personalized location recommendations. Some personalized recommendation systems request users to manually input her personal interests by categories (like restaurants) [3], which will be used to determine the POIs to be shown to the users. As a user's preferences are not actually binary decisions. Such systems usually bring a user too many or too few recommendations.

A branch of research starts to learn a user's interests from the user's location history and integrates the social environment of the user to make recommendations. Specifically, [4] stores people's location histories into a user-location matrix. Every entry in this matrix shows the number of visits of a user to the physical venue. Then, a user-based CF model can be used to get a user's interest to an unvisited venue. But all these research don't think about the particular scene of business area.

Our recommendation system differs from the above-mentioned work in the following aspects: 1) We think about the particular scene of intelligent business area. 2) We considered the affect of user's current location.

POI Recommender

This paper focus on the particular scene of intelligent business area. The scene with the following particularity: 1) the user's path has a certain regularity in the region, and therefore in the study of the historical behavior of the user, the path information is an important part. 2) The locations of business Area have specific category, and these category can be organized into tree structure universally, such as restaurant -> Chinese restaurant -> Sichuan restaurant. Specific user has specific preferences, on the recommendation of each POI should be considered a special semantic information. In addition, computing distance between locations in the business area is different with other area, cannot simply calculate by latitude and longitude, in fact, different points of interest due to the same distance in different malls, for the user is completely different. Based on the above analysis, we propose the following method.

Concept Definition. Firstly, this article defined the model of locations as Fig. 2.

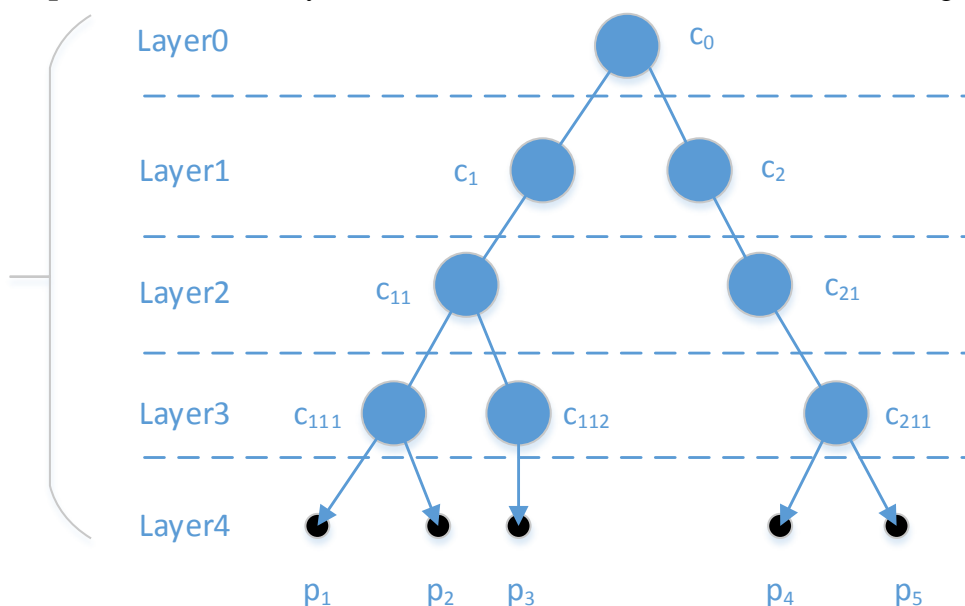


Fig. 2 the model of locations

In this model, p represents a POI in business area, c represents the category of p , in which, c_0 represents the root of all nodes in the system discussed in this article, which does not have practical significance. Under its category, using c_1, c_2, \dots, c_n represents the first stage of the category, using $c_{11}, c_{12}, \dots, c_{1n}$ represents the second stage of c_1 category, using $c_{111}, c_{112}, \dots, c_{11n}$ represents the third stage of c_{11} category, and so on. Taking into account the actual situation of the business district in the shop and the subsequent methods of operation efficiency in the system described in this paper, each store will have only three categories shown above. At the same time, l represents hierarchical category information location or locations' category. In which, l_0 represents the root layer where c_0 is, do not have the practical significance as c_0, l_4 where p represents the store layer, the rest of the hierarchy represents the corresponding category c , corresponding to the height of its location in the tree structure.

Secondly, this article defined the path information, t represents the collection of access points arranged in chronological order, namely path information. Trace $t = p_1 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$ represents a path information contains four locations, and users access these locations in the order of p_1, p_3, p_4, p_5 . In the method discussed herein, using the superscript indicate the user who complete this path, at the same time, since the path to store the access point, in the model the access point is l_i , using the subscript indicate the layer, so if t is the path of user 1, t can be represented as $t_4^1 = p_1 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$.

At the same time, path access point not only for the locations, may be a category information of locations. Category path is converted from location path's category information, category and location are correspondence, if user has been continuous access to the same category of location, the merger will be represented as a point in its path, like $t_4^1 = p_1 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$, its path of l_3 path will be represented as $t_3^1 = c_{111} \rightarrow c_{112} \rightarrow c_{211}$, its l_2 path will be represented as $t_2^1 = c_{11} \rightarrow c_{21}$, and so on.

ALGORITHM SIM($Q^{p,q}$)

Input: the similar path set of u_p and u_q ($Q^{p,q}$)
Output: the similarity score between u_p and u_q ($sim_{p,q}$)

1. $sim_{p,q} = 0;$ //similarity score of user
2. For each layer $l_i \in L$ AND $i > 0$ do
3. $sim_l = 0;$ // similarity score of l_i
4. $\alpha = f(i);$ // α is weighting factor of layer
5. For each $st = ST_i^{p,q}$ do
6. $sim_t = 0;$ // similarity score of similar path
7. $len = st.length;$ //length of similar path
8. $\beta = f'(len);$ // β is weighting factor of length
9. For each c_{ij} in st do
10. $IDF_{ij} = \log \frac{|U|}{n_{ij}};$ //every point's IDF
11. $sim_t = sim_t + IDF_{ij};$
12. $sim_l = sim_l + \beta \times sim_t;$ // layer similarity score with weighting
13. $sim_{p,q} = sim_{p,q} + \alpha \times sim_l;$ // user similarity score with weighting
14. $sim_{p,q} = sim_{p,q} / sim_{p,p};$ //normalization. The $sim_{p,p}$ is user similarity score of u_p and himself
15. return $sim_{p,q};$

Fig. 3 algorithm based on similar path

User Similarity Score Calculation. In consideration of the user route information, the user of the similarity calculation can be transformed into a user similarity computing path length. Obviously the longer the user similar path, the greater the similarity between users. The path can be got by the method: if user u_1 has path $t_4^1 = p_1 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$, and user u_2 has path $t_4^2 = p_2 \rightarrow p_3 \rightarrow p_5$, so the similar path of t_4^1 and t_4^2 will be $st = p_3 \rightarrow p_5$. Taking into account the above, you can use the same method to obtain a similar path on different layers. For example, using the same user u_1 and u_2 , and the same path t_4^1 and t_4^2 , the corresponding path of l_3 will be $t_3^1 = c_{111} \rightarrow c_{112} \rightarrow c_{211}$ and $t_3^2 = c_{111} \rightarrow c_{112} \rightarrow c_{211}$, so the similar path will be $st_3 = c_{111} \rightarrow c_{112} \rightarrow c_{211}$. Using this method can get all of the similar path set $Q^{1,2}$.

In this case, consider the impact of the different paths to the user similarity: Obviously, the path of the same layer, the longer similar path, the bigger similarity between users; and taking into account this specific scenario district, the location has category information, Although there is no access to the same location, but if there is access to the same the same category of location, there is also similar between the users, so the similar path are at different layers has different impact on the similarity of the user, the degree of influence between the different layers id different, the lower layer, the bigger impact of similarity, therefore, this paper add different weighting factor between different layers, so

that the lower layer, the greater the impact. Accordingly the similarity is calculated by the steps shown in . Fig. 3

In this algorithm, firstly get all of the similar path, its effect on the layer of similarity calculated by the TF-IDF method, secondly the similarity will be weighted by length to get the user similarity of this layer. After that the similarity will be weighted by layer, the similarity score between users is calculated, returning the result after normalization. In this algorithm, the longer path or the lower layer the greater influence of similarity score, so α , β function can use the function values $\alpha = 2^{l-1}$, $\beta = 2^{len-1}$. Using this algorithm can get the similarity score matrix (m_1) between all the users.

POI Recommendation with User Similarity Score Matrix. To get the POI recommendation, firstly using users' history path to get all users' Markov transition matrix as Fig. 4. In this case, if determined current location, can be obtained by different users at the current location, transiting to another location probability tensor, and these tensor can be composed to possibilities matrix (m_2) of all the users transiting to other locations from current location. Then using matrix multiplication to m_1 and m_2 to get user-location matrix which represents the users' probability of transiting from current location to other locations. After that, we can find the target user and get the top K locations as the result of recommendation.

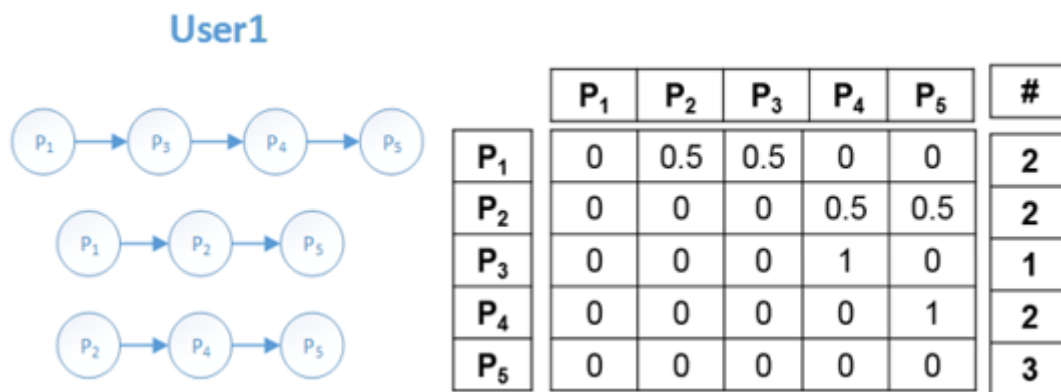


Fig. 4 Markov transition matrix

Summary

This paper proposed a method to solve the problem that based on the user's current location and historical information, providing POI recommendation to user in intelligent business area. The method defined the model of location. And using this model to calculate the user similarity score between all the users, and composed as similarity score matrix. At the same time, using users' history path to get all users' Markov transition matrix, if determined current location, getting possibilities matrix of all the users transiting to other locations from current location. Last using matrix multiplication to get top K locations as the result of recommendation.

References

- [1] J. van der Geer, J.A.J. Hanraads, R.A. Lupton, The art of writing a scientific article, J. Sci. Commun. 163 (2000) 51-59.
- [2] Y. Zheng, L. Zhang, X. Xie, and W.Y. Ma. Mining interesting locations and travel sequences from gps trajectories, WWW. (2009) 791-800.
- [3] K. Kodama, Y. Iijima, X. Guo, and Y. Ishikawa. Skyline queries based on user locations and preferences for making location-based recommendations, LBSN. (2009) 9-16.
- [4] M. Ye, P. Yin, W.C. Lee, and D.L. Lee. Exploiting geographical influence for collaborative point-of-interest recommendation, SIGIR.(2011) 325-334.