A Potential Friend Recommendation Algorithm for Obtaining Spatial Information

Hang Zhang, Zhongliang Cai*
School of Resource and Environmental and Sciences, Wuhan University, China

* Corresponding author. Email: zlcai@whu.edu.cn
Manuscript submitted July 10, 2020; accepted October 18, 2020.
doi: 10.17706/jsw.16.2.46-54

Abstract: With the rapid development of social network, friend recommendation algorithm has become an important component of social application. Location-based social network (LBSN) enables users to record and share their locations anytime and anywhere, which is a high quality information source. In order to meet people's demand of expanding social circle and obtaining diversified spatial information when making friends, this paper proposes a potential friend recommendation algorithm based on the similarity of user's check-in behavior and spatial information acquisition level in the real world. Firstly, we employ kernel density estimation and time entropy to solve the problems of data sparsity and low concentration, then employ cosine distance to measure the check-in behavior similarity. Secondly, we analyze users' spatial distribution of check-in location and cognitive differences on spatial information. Finally, the method mentioned above is tested with dataset called Foursquare. The results of the experiment show that the proposed method has competitive performance.

Key words: Recommendation algorithm, LBSN (location-based mobile social network), spatial information acquisition, time entropy.

1. Introduction

In recent years, location-based social network (LBSN) has developed rapidly, which representative applications are Foursquare, Gowalla, QQ and so on. LBSN provides user with a social platform. Friend recommendation is a LBSN's major function. Compared with the behavior in the virtual environment, the behavior information in the physical world can also represent people's preferences, which contributes to improve the accuracy of friend recommendation [1]. As a high quality information source, we can extract spatio-temporal information from check-in data. The tracks of users show the spatio-temporal regularity [2] and people tend to visit those locations near their homes [3]. We found that 75% of users always move around within 50 kilometers of their residences by analyzing the Foursquare check-in dataset. The closer two strangers' check-in behaviors are, the more similar their habits, and the higher their probability of becoming friends [4]. We can recommend potential friends by calculating the similarity of active time and locations.

The friend recommendation algorithms mostly employ collaborative filtering [5], random walk [6], genetic algorithm [7], weighted Tyson graph [8] and other methods. These recommendation algorithms are divided into two types in terms of recommendation ideas: (1) friend recommendation algorithm based on the user's social circle in the real world. (2) stranger recommendation algorithm based on the nearby similar social network topology. However, these recommendation algorithms don't consider user's spatial information demand, and the similarity comparison between users mainly focuses on topological features, which don't

take full advantage of the users' spatio-temporal features included in the social data.

This paper proposes a potential friend recommendation algorithm based on check-in locations data. We analyze check-in behaviors to mine their similarity to meet the demand of expanding social circle and obtaining diversified spatial information through friends [1]. The contribution of this paper mainly includes the following two points: Firstly, while expanding the social circle, the proposed algorithm considers the user's demand for spatial information acquisition. Secondly, we employ kernel density estimation and time entropy to solve the problem of data sparsity and low concentration. According to the experimental results, the proposed method has competitive performance.

The rest of this paper is organized as follows: In Section 2, we show the details of the presented method. Section 3 will show the experiments, results and analysis. In Section 4, conclusions will be draw.

2. Method

2.1. Probability Distribution of Check-in Behavior and Similarity Estimation

The check-in behavior in the physical world implies the user's personal preferences and life habits. Therefore, user's behavior in the physical world can improve the quality of recommendation. Location-based mobile social network records user's check-in information in the physical world. A check-in behavior refers to the action of completing a check-in and adding a check-in data to the database. Each record includes the time and location of the check-in behavior. Dividing the day into 24 equal time slots, each time slot represents 1 hour. According to the check-in time, put check-in data into the slots. After normalization, we get the user's check-in frequency in each time slot. The frequency represents the user's check-in probability in each time slot, and the probability distribution represents check-in behavior features.

Assuming that U is a check-in dataset, time slot is $X = (0,1,2,\ldots,23)$, U_a and U_b is representative users included in U. Their time slot frequency vector is shown in Fig.1.

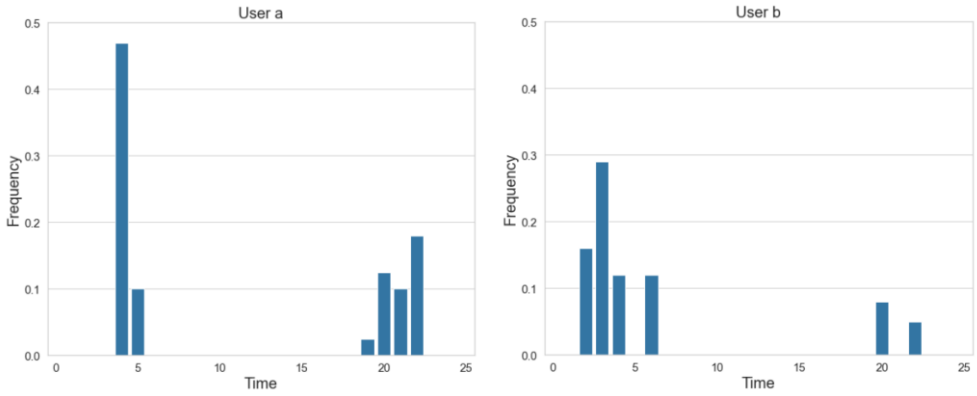

Fig. 1. Check-in frequency of user a and b.

However, when the day is divided, the check-in data of users is sparse in each slot. Using discrete check-in frequency directly, it is not accurate to estimate the probability distribution of check-in behavior. U_a and U_b don't check in most time slots. The sparse and discrete probability distribution enhances the difference of check-in behavior. U_a's check-in data is distributed in slot 4~5 and 19~22, while U_b's check-in data is distributed in slot 2~6 and 20~22, and there is no check-in data in time slot 5 and 21. In general, U_a is more centralized than U_b, and their distributions are similar overall.

To solve the problem of data sparsity, we employ kernel density method to estimate the user's check-in probability at other no checked time slots, generating continuous probability distribution. Kernel density estimation is a non-parametric probability density estimation method, and its general form is as equation (1).

$$\hat{f}_h(x) = \frac{1}{nh}\sum_{n}^{i=1}\left(\frac{x_i - x}{h}\right), x\epsilon R \tag{1}$$

where h is the probability density function, x_i is the statistics value of time slot i, and x is the estimated time point, n is the number of time slots. After kernel density processing, we obtain the continuous check-in probability distribution. The distributions of user U_a and U_b are shown in Fig.2. The probability after kernel density estimation can effectively extract the features of check-in data. Check-in differences in some time slots between them don't affect the overall feature.

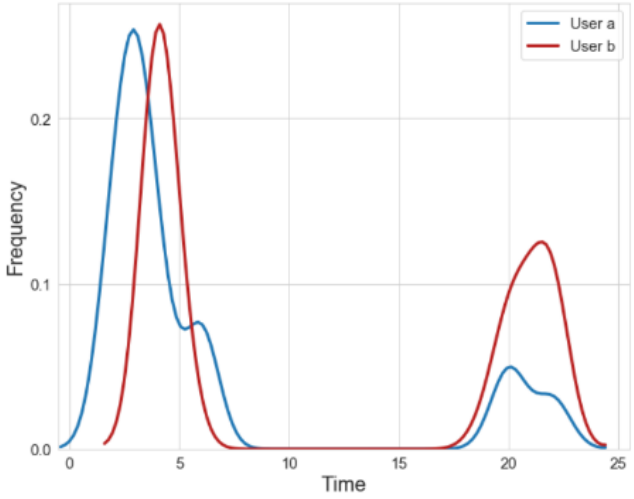

Fig. 2. Check-in probability density of user a and b.

We measure the similarity of check-in behavior by comparing their probability distributions. The check-in data in all time slots constitutes the check-in eigenvector. Assuming that we have user x and y whose check-in eigenvectors are $\vec{x}$ and $\vec{y}$. We employ cosine distance of $\vec{x}$ and $\vec{y}$ to measure the difference between them, the formula is as equation (2).

$$sim_{checkin} = sim(x,y) = \frac{\sum_{s\epsilon S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s\epsilon S_{xy}} r_{x,s}^2 \sum_{s\epsilon S_{xy}} r_{y,s}^2}} \tag{2}$$

where $sim(x,y)$ is the similarity between x and y, $r_{x,z}$ is the check-in probability of x in the slot s, $r_{y,s}$ is the check-in probability of y in the slot s, S_{xy} is the set of check-in probabilities of x and y in each time slot. The similarity between U_a and U_b is 0.8538.

2.2. Time Entropy

The cosine distance can represent the similarity of users' check-in behavior. However, although the probability distribution in most time slots is similar, the difference in the probability in a few time slots will greatly reduce the similarity. Indeed, active check-in time slots are more representatively than inactive ones. If one user checks in those inactive time slots and another user doesn't, the value of similarity between them will drop, even if the difference is not related to effective recommendation features. Being sensitive to the check-in time distribution causes many potential friends to be excluded.

We employ time entropy to highlight the behavior features of users in active time slots. The formula of time entropy is as equation (4).

$$Time_i = -p_i \log(p_i) \tag{4}$$

where p_i is the probability that the checks in at the time slot i. Time slots with more check-in data have

higher time entropy, which makes it have more weight in similarity calculation.

The improved calculation method with time entropy is as follows: Firstly, calculate the time entropy of the user's check-in probability in each time slot. Secondly, use the kernel density method to make smooth probability estimation. Finally, the time entropy after smoothing is used to calculate the cosine similarity. We use the improved method to calculate the similarity between U_a and U_b, $sum(u_a,u_b) = 0.9104 > 0.8538$.

We analyzed the check-in data of 500 selected users randomly in the Foursquare dataset, then used the above method to calculate the similarity of check-in behavior between every two users. Fig.3 shows a typical similarity distribution between users. Users with a similarity level of 0.7~0.9 accountes for the largest proportion of all users in the dataset. When the cosine distance of two users is greater than 0.8, the proposed method thinks they have similar check-in behavior and can be regarded as candidate recommendation users (candidate user for short).

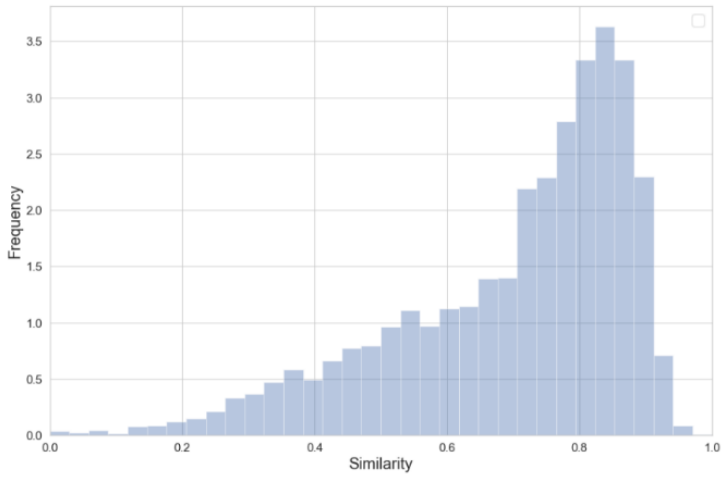

Fig. 3. Similarity distribution processed by time entropy.

2.3. User Recommendation Considering Spatial Information Acquisition

In this section, we will filter the candidate users to the final recommendation users who have both the enough same check-in location and different check-in location. If two check-in locations are close, they are called same check-in locations, while if two check-in locations are far, they are called different check-in location.

The users that are served by the proposed algorithm are called target user. We regard the intersection set of the check-in location of the target user and the candidate user as the same check-in location. Users with same check-in locations share similar experiences and geospatial cognition, which is the basis to become friends. We employ $sim_{spectical}$ to measure the similarity of geospatial cognition between user as equation (5).

$$sim_{spectical} = S_T \cap S_P \tag{5}$$

where $Sim_{spectical}$ is the same check-in location collection of target users and candidate users. S_T is the check-in location collection of target user, S_P is the check-in location collection of candidate user.

We regard the different check-in locations of target users and candidate users as a source of new spatial information for them. Different check-in locations between users means different experiences and differentiated geospatial cognition, which is a potential information source for providing spatial information to other users. We employ $dif f_{spectical}$ to measure difference of geospatial cognition between user as equation (6).

$$diff_{spectical} = (S_T - S_P) \cup (S_P - S_T) \tag{6}$$

where $diff_{spectical}$ is the different check-in location collection of target users and candidate users. $(S_T - S_P)$ is a location collection where the target users have checked in and candidate users haven't, $(S_P - S_T)$ is a location collection where candidate users have checked in and target users haven't.

The more frequent visits, the greater the weight of the check-in location. The location-based friend recommendation method that considers the number of visits is as equation (7) and the constraint is as equation (8).

$$u_{RC} = argmax\{diff_{user}\} = \underset{p \epsilon U_C}{arg\max}\{W_{sim} sim_{spectical} \cdot W_{diff} diff_{spectical}\} \tag{7}$$

$$= \underset{\epsilon U_C}{arg\max}\left\{\sum_{i=0}^{N=|sim_{spectical}|} W_{simi} \cdot \sum_{i=0}^{N=|diff_{spectical}|} W_{diffi}\right\}$$

$$|sim_{spectical}| > \lambda \tag{8}$$

where $diff_{user}$ is the weighted spatial cognition score, argmax$\{diff_{user}\}$ is the $Top-K$ recommended users, W_{simi} is the number of times that the target user and candidate user checked in at every same check-in locations, W_{diffi} is the number of times that the target user and candidate user checked in at every different check-in locations, λ is the lowest allowed number of the same check-in locations. W_{simi}. $W_{sim} sim_{spectical}$ represents spatial similarity, and $W_{diff} diff_{spectical}$ represents new geographic information.

In conclusion, there are three steps to select the final recommended users from the candidate users. Firstly, calculate $sim_{spectical}$ and W_{sim} for each user and exclude users whose $sim_{spectical}$ is less than λ to ensure similar spatial experiences between them. Secondly, calculate $diff_{spectical}$ and W_{diff}. Finally, calculate $diff_{user}$ to rank the final recommended users. The candidate recommendation users with high score are the final recommendation users.

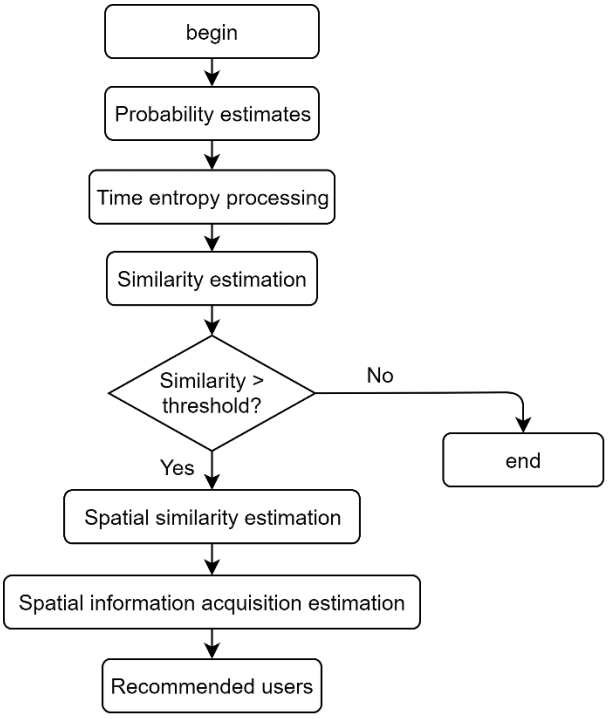

Fig. 4. Procedure of the proposed method.

2.4. Cold Start of the System

The cold start problem is inevitable in a recommendation system [9]. The solution employed in the proposed method is: 1) For newly registered users (without any record), the system recommends $Top-N$ users to target users based on the recommended times of nearby users. 2) Before a certain number of check-in data are recorded, the system recommends potential friends of their existing friends who meet the algorithm criteria to target users. 3) When the number of users' check-in records satisfies the above proposed algorithm, the system employs the proposed algorithm to make recommendations.

3. Experiment and Result

In this part, we use Foursquare, a famous social network check-in dataset including check-in data and friendship data, as the research object. The dataset records a total of 1048575 check-in records on 63832 locations from February 04, 2009 to October 22, 2010. Each check-in record consists of user ID, check-in location ID, longitude, latitude, and check-in time. The friendship file records 102,296 pairs of friends of 4121 users. The mean check-in times on each location is 16.43, and the mean check-in times of each user is 254.44. According to the above analysis, the dataset is sparse.

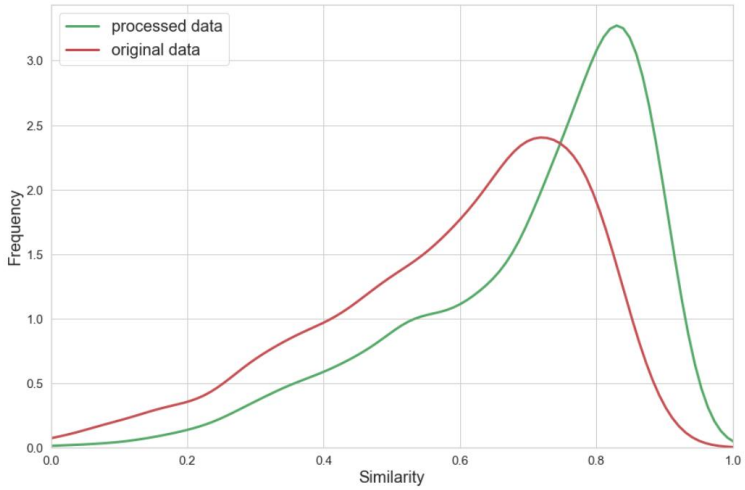

Fig. 5. Check-in distribution of original data and processed data.

We implement the proposed method with the Foursquare check-in dataset. We selected users with user ID of 400 as target users to complete the experiment, who has 4120 recommended candidates. From the experimental results, we can know that there are 1,511 candidate recommended users with a similarity of more than 0.8 with the target users. The maximum similarity was 0.9585. We calculated the spatial information acquisition scores of candidate users, the check-in behavior similarity, spatial similarity and new geographic information of *TOP-10* recommendation users. Detail results are shown in Table 1.

In this paper, we employ F_1 and $spatial_{infor}$ to measure the accuracy of recommendation and the spatial information obtained from the recommended users. Among all the users, the target users' existing friends are more similar with target user and more likely to be recommended. F_1 is calculated by $precision$ and $recall$. The formula of $precision$ is as equation (9).

$$precision = \frac{\sum_{i=1}^{n}|R(i) \cap T(i)|}{\sum_{i=1}^{n}|R(i)|} \tag{9}$$

where N is the number of experiments, and $R(i)$ is the set of candidate users recommended to user i. $T(i)$ is the existing friends' set of user i. $Precision$ represents the proportion of existing friends in the in the

recommended users.

Table 1. TOP-10 Recommended User Check-in Behavior Indicators for User 400

Recommend rank	User id	Check-in behavior Similarity	Spatial similarity	New geographic information
1	243	0.9236	667	1308
2	321	0.9196	511	1564
3	522	0.8833	267	1677
4	469	0.9290	414	788
5	733	0.8821	156	2044
6	1031	0.8967	150	2100
7	536	0.8636	148	2052
8	20	0.8798	122	2153
9	127	0.8639	284	901
10	405	0.8486	116	1891

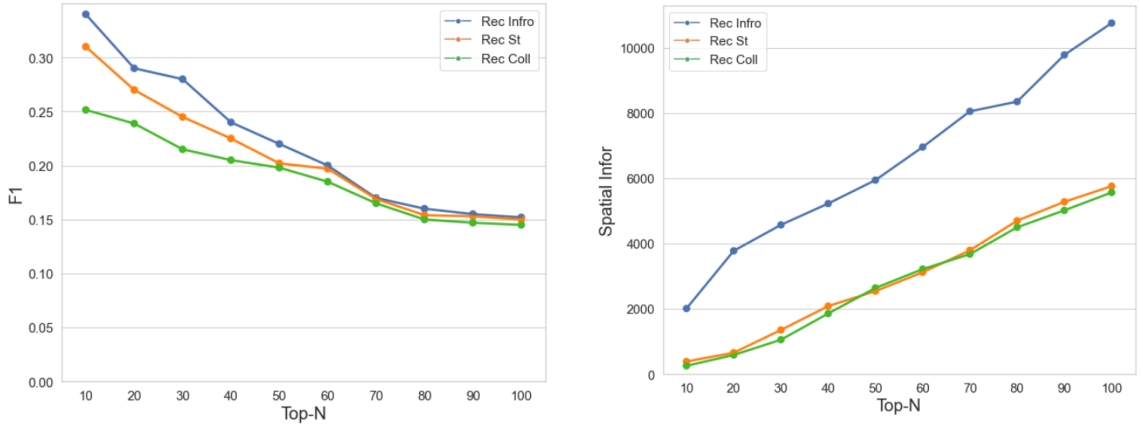

Fig. 6. F_1 and $spatial_{infor}$ based on different methods.

The formula of $recall$ is as equation (10).

$$recall = \frac{\sum_{i=1}^{n}|R(i) \cap T(i)|}{\sum_{i=1}^{n}|T(i)|} \tag{10}$$

The recall rate represents the proportion of existing friends in the recommended users. The F_1 is defined as equation (11).

$$F_1 = \frac{2 \times precision \times recall}{precision + recall} \tag{11}$$

F_1 value is the harmonic average of $precision$ and $recall$, which comprehensively represents the $precision$ and $recall$. The higher the F_1 value is, the more effective the recommendation algorithm is.

$Spatial_{infor}$ represents the number of spatial information provided by recommended friends to target users. The formula of $spatial_{infor}$ is as equation (12).

$$spatial_{infor} = \sum_{i=1}^{N} M_{i,top_n}/N \tag{12}$$

where N is the number of experiments. M_{i,top_n} is the mean number of space information provided by the $Top-N$ recommended users of target user u_i.

In order to show the performance of our method, we compare the proposed method, as Rec_{infor}, with the following methods, which are Real-time stranger recommendation algorithm based on spatio-temporal correlation, as Rec_{st}, and recommendation algorithm based on collaborative filtering, as Rec_{coll}. We

randomly selected 200 users as target users, and carried out $Top-N$ recommendation.

Fig. 6 is the results of the experiment, showing the changes of F_1 and $spatial_{infor}$ with the recommended number of people. The F_1 of Rec_{infor} has best performance when the number of recommendations is small. The spatial information obtained by Rec_{infor} is more 200% better than the Rec_{st} and Rec_{coll}.

4. Conclusion

In order to meet people's demand for spatial information acquisition when making friends, this paper proposes a potential friend recommendation algorithm based on the spatio-temporal similarity of users' check-in behaviors in the real world. Firstly, we analyzed the distribution of user check-in data in the time slots, solved the problem of data sparsity with kernel density estimation, and improved the data concentration with time entropy, which better match multi-peak data. Secondly, we analyzed the spatial distribution of user check-in locations and propose to use same and different check-in locations to measure the users' spatial similarity and the level of spatial information acquisition. Finally, we used the Foursquare dataset to complete the experiment, which verified that our method had competitive recommendation performance and higher level of spatial information acquisition.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Zhongliang Cai, Hang Zhang conducted the research; Hang Zhang analyzed the data; Hang Zhang designed and tested the algorithm; Hang Zhang wrote the paper; Zhongliang Cai guided the writing; all authors had approved the final version.

References

[1] Xu, B., Chin, A., & Wang, H. (2011). Using physical context in a mobile social networking application for improving friend recommendations. *Proceedings of the 2011 Int'l Conf. on Internet of Things, and 4th Int'l Conf. on Cyber, Physical and Social Computing*.

[2] Gonzalez, M. C., Hidalgo, C. A., & Barabasi, A. L. (2008). Understanding individual human mobility patterns. *Nature*, *453(7196)*, 779–782.

[3] Yin, H., Sun, Y., Cui, B., Hu, Z., & Chen, L. (2013). Lcars: A location-content-aware recommender system. *Proceedings of the 19th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*.

[4] Mahmud, J., Zhou, M. X., Megiddo, N., Nichols, J., & Drews, C. (2013). Recommending targeted strangers from whom to solicit information on social media. *Proceedings of the 2013 Int'l Conf. on Intelligent User Interfaces*.

[5] Bian, L., & Holtzman, H. (2011). Online friend recommendation through personality matching and collaborative filtering. *Proceedings of the UBICOMM*.

[6] Yu, X., Pan, A., Tang, L. A., Li, Z., & Han, J. (2011). Geo-Friends recommendation in GPS-based cyber-physical social network. *Proceedings of the 2011 Int'l Conf. on Advances in Social Networks Analysis and Mining (ASONAM)*.

[7] Chu, C. H., Wu, W. C., Wang, C. C., Chen, T. S., & Chen, J. J. (2013). Friend recommendation for location-based mobile social networks. *Proceedings of the 2013 the 7th Int'l Conf. on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*.

[8] Silva, N. B., Tsang, I. R., Cavalcanti, G. D. C., & Tsang, I. J. (2010). A graph-based friend recommendation system using genetic algorithm. *Proceedings of the 2010 IEEE Congress on Evolutionary Computation*

(CEC).

[9] Akshita, S. A., *et al*. (2013). Recommender system: Review. *International Journal of Computer Applications*, *71(24)*, 38-42.

Hang Zhang was born in Hei LongJiang, China. He is currently pursuing the bachelor degree in geographic information science from Wuhan University. His research interests mainly include geographic big data and 3D geographic information system

Zhongliang Cai was born in Shandong, China. He received the PhD degree in map making and geographic information engineering, School of Resource and Environmental Sciences, Wuhan University, China, 2004. Now he works as a professor in School of Resource and Environmental Sciences in WHU and his research interests include digital map theory and method, geographic information service and GIS software developing method.