A Friend Recommendation Scheme in Social Network Environments

Kyoungsoo Bok, Hyeonwook Jeon
School of Information and Communication Engineering
Chungbuk National University, Chungdae-ro 1, Seowon-gu, Cheongju, Chungbuk 28644, Korea

Chunghui Lee
Life Insurance CI Group, Samsung SDS
Samsung SDS Tower East Campus, 123, Olympic-ro 35-gil, Songpa-gu, Seoul 05510, Korea

Jaesoo Yoo
School of Information and Communication Engineering
Chungbuk National University, Chungdae-ro 1, Seowon-gu, Cheongju, Chungbuk 28644, Korea

ABSTRACT

In this paper, we propose a friend recommendation scheme that takes into consideration the attribute information of a POI and a user's movement patterns. The proposed scheme broadly consists of a part that filters out other users who have different preferences by calculating preferences using the attribute information of users and a part that finds a moving trajectory close to that of a user with a pattern-matching scheme. To verify the superiority of the proposed scheme, we compare it with existing schemes through various performance evaluations.

Key words: Social Network, Friend Recommendation, Preference, POI.

1. INTRODUCTION

SNSs with the characteristics of the mobile environment have changed a lot to become more suitable for mobile social networks that provide mobility-based services [1], [2]. In particular, various services combining location-based services have been provided in recent years. One of the important features of SNSs is expanding and maintaining relationships between users [3], [4]. To this end, most SNSs provide a feature that recommends friends who are closer to a user in terms of common interests or profiles [5], [6]. In the location-based social network, not only static attributes (e.g., a user's profile and relationship information), but also dynamic attributes (e.g., a user's location and moving preferences) and situational awareness-based behavior are utilized [7], [8]. However, most SNSs consider only static information of users to recommend friends. That is, friends are recommended based on the user profiles or existing relationship information [4], [6], [9].

Recently, studies on recommendations of itineraries, friends, and locations over social networks have been conducted using clustering or pattern-matching schemes. In [10], a scheme that recommended a user-preferred travel itinerary was proposed for users travelling to unfamiliar areas. This scheme analyzed trajectories using a Global Positioning System (GPS) and recommended a proper itinerary in terms of the user's preferred start location, arrival location, and times considering the user's travelled locations, times, and popularity. However, the scheme proposed in [10] only considered the trajectories of other travelers, so the user's personal preferences and propensities were not reflected in the recommendation. In [11], a scheme that recommended other users who had similar preferences was proposed using category-based location data. However, since this scheme determined the similarity of users based on the types and orders of past Points of Interest (POIs) chosen by users, it was limited, in that it did not consider detailed POI information.

This paper proposes a recommendation scheme that can determine the detailed preferences of users by considering detailed POI information to recommend friends with similar user preferences over mobile social networks. The proposed scheme extracts preferences using past trajectories of users and recommends friends by comparing trajectories of users whose preferences are similar. The proposed scheme creates POI trajectories to determine trajectory similarity using only meaningful trajectory information of users, and it compares the created POI trajectory information, thereby calculating user

_____

* *Corresponding author, Email: yjs@chungbuk.ac.kr*

preferences. Using the calculated preferences, the trajectories of other users with different preference are filtered out. Through such trajectory filtering, computation amounts can be reduced in the process of trajectory comparison between users. In addition, trajectories similar to the user's POI trajectories are extracted using a pattern-matching scheme, and friends with similar user preferences and moving locations are recommended.

This paper is organized as follows. In Section 2, we describe the characteristics and execution process of the proposed scheme. In Section 3, we evaluate the performance of the proposed scheme. Finally, in Section 4, the conclusion of this paper is presented.

2. RECOMMENDATION SCHEME

2.1 Characteristics

Existing schemes determine the similarity between users using pattern-matching or clustering schemes. In addition, similarity is determined through types and orders of a user's past POIs. However, user preferences cannot be determined completely only by using POI types. For example, when a user moves to a specific location, information about transportation required to reach the location or the price of the location can also influence user preferences. Therefore, this paper proposes a recommendation scheme that considers not only types and orders of POIs but also detailed attributes of POIs. This paper proposes a friend recommendation scheme that calculates user preferences using moving trajectories of users from the same location in daily living over mobile social network environments and recommends friends who have similar preferences to those of a user. In the proposed scheme, two similarity comparison schemes are considered: an attribute similarity comparison scheme and a trajectory similarity comparison scheme. The attribute similarity comparison scheme calculates user preferences using the attribute information of a user's past POIs to filter out other users who have different preferences from the user. The trajectory similarity comparison scheme calculates how similar two POI trajectories are by comparing them. That is, it calculates the similarity of moving patterns between two POI trajectories, and other users with similar moving patterns are recommended.

Fig. 1 shows the friend recommendation process of the proposed scheme. The proposed scheme consists of the following five steps. First, POI trajectories are created using GPS trajectories and a POI database. Since GPS trajectories are lines connecting locations where users have visited over time, not all regions over the lines necessarily reflect user preferences. Therefore, POI trajectories are created using POI information in past trajectories to compute preferences. Second, the similarities of all users are calculated using POI trajectory information. The calculated attribute similarities are used to compare preferences with other users. Third, users with different attribute similarities from recommended users are filtered out using the attribute similarities. Through the filtering, POI trajectories that have different preferences from the user are removed from the recommendation, thereby reducing the computation amount while calculating POI trajectory

similarities. Fourth, the trajectory similarity of a user is compared so that users with a trajectory similarity closer to a user's moving pattern can be recommended. Finally, k friends are recommended to the user.
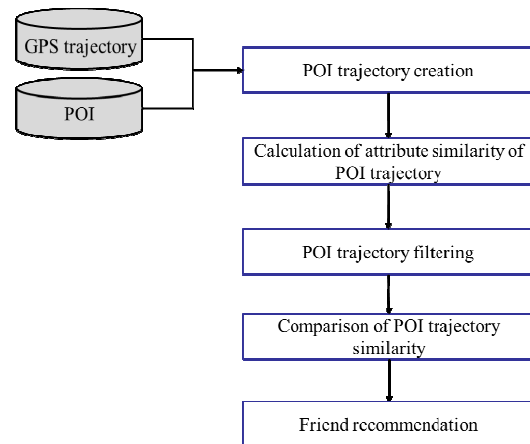

Fig. 1. Friend recommendation process

2.2 POI trajectory creation

In general, trajectory tr is the stored moved paths of user o_i over specific locations, which is represented by $tr(o_i) = (p_1, p_2, ..., p_k)$, where $p_i = (x, y, t)$. Here, x and y are x and y coordinates, and t is a timestamp. POI refers to locations in which a user is interested, such as a cinema or a museum. POI also contains specific information of each POI. Therefore, POIs are located within a certain radius based on the particular center location where the POI is found. Each POI has its own type and attributes, and the attribute values depend on the type. A POI type refers to the category to which a location (or place) belongs.

To calculate user preferences, a POI trajectory (ptr) that connects POIs where a user stays over a trajectory is calculated through the user's moved trajectory and a database storing POIs. POIs included in the user's POI trajectory do not contain POIs in which the user visited but include POIs in which the user visited within a radius of the POI for more than a certain period of threshold time. The POI trajectory is defined as $ptr(o_i) = (pp_1, pp_2, ..., pp_k)$, which is similar to the user trajectory definition. Here, it is $pp_i = (x, y, [t_s, t_e])$, where x and y refer to x and y coordinates of the center location of a POI a user visits, while t_s and t_e are timestamps at the time a user arrives and leaves the POI, respectively.

2.3 Attribute similarity comparison

To calculate the attribute similarity of POI attribute information, a method of quantifying POI attribute information must be provided. However, some attribute information may be difficult to quantify objectively depending on the attribute type. Attributes are classified according to an attribute type, and similarity is calculated in line with the attribute type. An attribute can be classified into two types: a type that has a discrete value between 0 and 1 and a type that has a continuous value between 0 and 1. In addition, attributes are classified into

two types according to criteria that have values. The first type is attributes that cannot be quantified objectively according to a particular criterion. For example, let us assume that the POI of a cinema has an attribute called cinema size. An attribute such as cinema size is difficult to quantify objectively. However, by applying a common criterion, they can be classified as either large or small. The second type is attributes that can be quantified objectively according to a specific criterion.

Attribute similarity is calculated by comparing a user's POI trajectory and attribute information value. If a POI type has a number of attributes, the attribute similarity of each attribute in a POI is calculated. That is, if a POI type has two attributes, called a and b, the attribute similarity value of a and b, are calculated respectively. Attribute similarity $attsim()$ is calculated via Equation (1) in which the numbers of attributes per POI type of users u_1 and u_2 are compared with each other when m POI types are present. $d_l(u_1, u_2)$ is a distance per attribute type where a user visits over the POI trajectories between u_1 and u_2.

$$attsim(u_1, u_2) = \sum_{l=1}^{n} d_l(u_1, u_2) \qquad (1)$$

2.4 Recommendation

Trajectory similarity is compared with POI trajectory. In general, a user's trajectory has a different length from that of a compared trajectory. Therefore, to compare trajectory similarity, this paper compares a shorter user POI trajectory with a longer user POI trajectory by dividing the longer trajectory into the same length of the shorter trajectory and comparing them sequentially until the longer trajectory is completely compared. When the comparison of POI trajectories calculates whether each POI matches, 1 is given if they match; otherwise, 0 is given. Trajectory similarity values are assigned as the largest value of a POI trajectory comparison once the shortest POI trajectory is moved sequentially and compared with longer trajectories.

To calculate a trajectory similarity $trsim()$, the POI trajectory of u_1, $ptr1$, and the POI trajectory of u_2, $ptr2$, are compared, as indicated in Equation (2). Equation (2) refers to a formula by which the trajectory similarities of two POI trajectories are compared. First, as shown in Equation (3), $pp1_i$ and $pp2_j$, POI locations of POI trajectories of two users, are compared, and if they match, a value of 1 is given; otherwise, a value of 0 is given. A summation value of results of value 1 is multiplied by $\dfrac{2}{m1+m2}$. A result value calculated through this process is computed as many as $m1 - m2 + 1$ times, which is the number of comparisons of trajectories while a shorter POI trajectory is moved sequentially. Here, m1 is the number of POIs in a longer POI trajectory, while m2 is the number of POIs in a shorter POI trajectory. Once the number of comparisons is complete and the calculation is done, the largest

value among the result values is determined as the trajectory similarity value.

$$trsim(ptr1, ptr2) = max\{ \frac{2}{m1+m2} \sum_{i=1}^{m1} \sum_{j=1}^{m2} s(pp1_i, pp2_j) \} \quad (2)$$

$$s(pp1_i, pp2_j) = \begin{cases} 1, & if\ pp1_i = pp2_j \\ 0, & otherwise \end{cases} \qquad (3)$$

3. PERFORMANCE EVALUATION

The performance evaluation of the proposed scheme in this paper was conducted with virtual trajectory data by creating POIs. To experiment over various environments, the performance evaluation was conducted by changing the number of POIs and types as well as the number of trajectories. In addition, the same weight was assigned to all POI types to conduct the performance evaluation. The performance evaluation was implemented by Java on a computer with an Intel core 2 Duo E7200 2.53GHz, 4GB memory, and Microsoft Windows 7 OS. Table 1 shows the performance evaluation parameters. The number of POIs was set to 30~100, while the number of POI types was set to 5~10. The number of trajectories was set to 30~100. To evaluate recall and precision, an attribute similarity threshold was set to 0.1~0.5

Table 1. Performance evaluation parameters

Parameter	Value
No. of POIs	30~100
Type of POI (category)	5~10
No. of trajectories	30~100
Threshold of attribute similarity	0.1~0.5

Broadly, two methods were used to evaluate the performance of the proposed scheme. First, the degree of matches in POI trajectories between users who requested recommendations and recommended candidate users was compared. Hereafter, such comparison elements of POI trajectories are referred to as precision and user similarity. Here, precision is the ratio of POI matches with a user who requests a recommendation based on the POI trajectories of recommendation candidate users. Recall refers to the ratio of POI matches with recommendation candidate users via the system based on the POI trajectories of a user who requests a recommendation. This paper calculates recommendation and user similarities by comparing POI matches over the POI trajectories. Second, results of POI trajectories between the user who requests a recommendation and recommendation candidate users obtained through the proposed scheme were compared to verify the similarity of POI trajectories between the user who requests a recommendation and recommendation candidate users visually.

Fig. 2 shows the user and recommendation similarities according to the attribute similarity threshold. User and recommendation similarities were measured by varying the

attribute similarity threshold value between 0.1 and 0.5 while fixing the trajectory similarity threshold to 0.3. The higher the attribute similarity threshold value, the lower the user and recommendation similarities (more or less). Through this result, it was found that as the attribute similarity threshold increases, other users who have POI trajectories less similar to those of a user are likely to be recommended, while user and recommendation similarities are inversely proportional to the attribute similarity threshold.
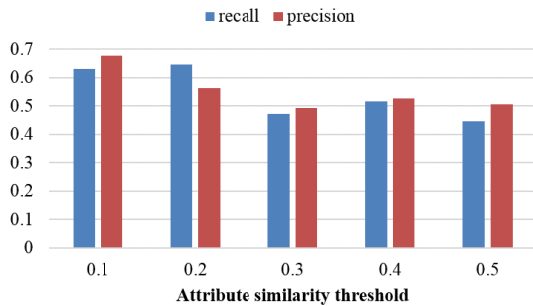

Fig. 2. Recall and precision according to the attribute similarity threshold

The proposed scheme had considerably higher user and recommendation similarities than MSTP-Similarity [3], according to the performance evaluation. In MSTP-Similarity, the number of POI types that matched between the POI trajectories of two users had a significant effect on friend recommendation. However, it cannot be said that two users actually visited the same POIs even if their POI types were similar to the POI trajectories. On the other hand, since the proposed scheme compared POI locations users visited through the trajectory similarity, it had more matched POI trajectories of two users than the existing scheme. Accordingly, the proposed scheme revealed considerably higher users and recommendation similarities than the MSTP-Similarity scheme. The experimental results showed that the proposed scheme improved recall by about 90% compared to the existing MSTP-Similarity scheme, while it improved precision by about 116%.

Fig. 3 and 4 show the POI trajectories of a user who requests a recommendation and recommendation candidate users. Fig. 3 shows a past POI trajectory of a user who requests a recommendation. In the figure, the icons are POIs, while the blue line is the past POI trajectory of the user requesting a recommendation. The user's POI trajectory shows that the user visited two types of POIs twice. Fig. 4 shows the POI trajectory results of recommended users by comparing the attribute and trajectory similarities of POI trajectories between the user requesting a recommendation and other users while the threshold values of the attribute and trajectory similarities were set to 0.2 and 0.3, respectively. The recommended users have red and black POI trajectories. The red and black lines show that the recommended users have visited the same types of POIs as the user who requested a recommendation. In addition, the red POI trajectory has two POIs that match the blue POI trajectory, while the black trajectory has two POIs that match the blue POI trajectory. As shown in these results, the proposed scheme recommends users who have past POI types and

moving patterns similar to those of a user who requests a recommendation through the comparison of attribute and trajectory similarities.

Fig. 3. POI trajectory of a user requesting a recommendation

Fig. 4. POI trajectories of recommended users

4. CONCLUSION

In this paper, we proposed a friend recommendation scheme that reflects the detailed preferences of users by considering POI types and attribute information. The proposed scheme considered not only POI types and orders but also detailed attributes of POIs by employing attribute similarity. It was shown through various performance evaluations that the proposed scheme outperforms the existing schemes. Furthermore, it was visually verified that users who had POI trajectories similar to those of the user requesting a recommendation were recommended as friends. In the near future, we will apply the proposed scheme to real mobile social networks.

REFERENCES

[1] R. Akhtar, S. Leng, I. Memon, M. Ali, and L. Zhang, "Architecture of Hybrid Mobile Social Networks for Efficient Content Delivery," Wireless Personal Communications, vol. 80, no. 1, 2015, pp. 85-96.

[2] B. Fan, S. Leng, K. Yang, and Q. Liu, "GPS: A method for data sharing in Mobile Social Networks," Proc. Networking Conference, 2014, pp. 1-9.

[3] J. Ying, E. Lu, W. Lee, T. Weng, and V. Tseng, "Mining User Similarity from Semantic Trajectories," Proc. International Workshop on Location Based Social Networks, 2010, pp. 19-26.

[4] N. V. Spirin, J. He, M. Develin, K. G. Karahalios, and M. Boucher, "People Search within an Online Social Network: Large Scale Analysis of Facebook Graph Search Query Logs," Proc. International Conference on Conference on Information and Knowledge Management, 2014, pp. 1009-1018.

[5] X. Xie, "Potential Friend Recommendation in Online Social Network," Proc. IEEE/ACM Green Computing and Communications-Cyber Physical and Social Computing, 2010, pp. 831-835.

[6] Z. Zhe and Z. Li, "A Method of Visualizing Friends Relations and Recommending Groups in Online Social Network," Proc. International Conference on Fuzzy Systems and Knowledge Discovery, 2012, pp. 836-839.

[7] J. Cao, Q. Hu, and Q. Li, "A Study of Users' Movements Based on Check-In Data in Location-Based Social Networks," Proc. International Symposium on Web and Wireless Geographical Information Systems, 2014, pp. 54-66.

[8] J. Bao, Y. Zheng, and M. F. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data," Proc. International Conference on Advances in Geographic Information Systems, 2012, pp. 199-208.

[9] X. Yu, A. Pan, L. A. Tang, Z. Li, and J. Han, "Geo-Friends Recommendation in GPS-based Cyber-physical Social Network," Proc. International Conference on Advances in Social Networks Analysis and Mining, 2011, pp. 361-368.

[10] H. Yoon, Y. Zheng, X. Xie, and W. Woo, "Smart Itinerary Recommendation Based on User-Generated GPS Trajectories," Proc. International Conference on Ubiquitous Intelligence and Computing, 2010, pp. 19-34.

[11] X. Xiao, Y. Zheng, Q. Luo, and X. Xie, "Finding Similar Users Using Category-Based Location History," Proc. ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems, 2010, pp. 442-445.

Kyoungsoo Bok
He received the B.S. in Mathematics from Chungbuk National University, Korea in 1998 and also received M.S. and Ph.D. in Information and Communication Engineering from Chungbuk National University, Korea in 2000 and 2005. He is now a research professor in Information and Communication Engineering, Chungbuk National University, Korea. His research interests are database system, location based service, mobile ad-hoc network, big data processing, and social network service.

Hyeonwook Jeon
He received the B.S. degree in the Department of Information and Communication Engineering in 2014 from Kangwon National University, South Korea. He is a student for M.S. degree in Department of Information and Communication Engineering from Chungbuk National University, South Korea. His research interests are the big data processing and streaming data processing

Chunghui Lee
He received B.S. and M.S. in Information and Communication Engineering from Chungbuk National University, Korea in 2011 and 2013. He is now an assistant Engineer in Life Insurance CI Group, Samsung SDS, Korea. His research interests are wireless sensor network, big data processing, and social network service.

Jaesoo Yoo
He received M.S. and Ph.D. in Computer Science from Korea Advanced Institute of Science and Technology, Korea in 1991 and 1995. He is now a professor in Information and Communication Engineering, Chungbuk National University, Korea. His research interests are database system, storage management system, sensor network, distributed computing, big data processing, and social network service.