An Efficient Approach to Generating Location-Sensitive Recommendations in Ad-hoc Social Network Environments

Fei Hao[1], Shuai Li[2,3], Geyong Min[4], Hee-Cheol Kim[2], Stephen S. Yau[5], and Laurence T. Yang[1,6,*]

Abstract—Social recommendation has been popular and successful in various urban sustainable applications such as online sharing, products recommendation and shopping services. These applications allow users to form several implicit social networks through their daily social interactions. The users in such social networks can rate some interesting items and give comments. The majority of the existing studies have investigated the rating prediction and recommendation of items based on user-item bipartite graph and user-user social graph, so called social recommendation. However, the spatial factor was not considered in their recommendation mechanisms. With the rapid development of the service of location-based social networks, the spatial information gradually affects the quality and correlation of rating and recommendation of items. This paper proposes Spatial Social Union (*SSU*), an approach of similarity measurement between two users that integrates the interconnection among users, items and locations. The SSU-aware location-sensitive recommendation algorithm is then devised. We evaluate and compare the proposed approach with the existing rating prediction and item recommendation algorithms subject to a real-life dataset. Experimental results show that the proposed SSU-aware recommendation algorithm is more effective in recommending items with the better consideration of user's preference and location.

Index Terms—Rating prediction; social networks; spatial social union; recommendation; sustainablility.

I. INTRODUCTION

With the rapid development of smart city for sustainable economic growth and prosperity for their citizens, social recommendation, one of the most common activities in citizens' social networks, has been widely used in various urban sustainable applications including travel information guides, shopping guides, emergency services, local weather report, provision of traffic information and so forth [27], [28]. The conventional social networking services, such as Epinions and Flixster, have witnessed great success on online sharing, products recommendation and shopping services. In spite of their success, most social networking services did not take location information into account. In other words, the social recommendation with geographical information of location where mobile device user stays, namely location-sensitive recommendation, has been largely neglected. A typical location-sensitive recommendation is defined and represented with a three-dimensional space including user, item, and location, as shown in Fig. 1. From the social sustainability point of view, the location-sensitive recommendation technologies support two main merits: 1) provides the information on citizen's sustainability initiatives and items via location-based social media channels (*e.g. foursquare*); 2) minimizes citizen's impact on the environment by optimizing their mobility of daily life. For example, regarding a location-sensitive recommendation service of restaurants for tourists, a certain visitor or a group of visitors may wish obtain the information of the recommended nearby restaurants including the distance, price, and popularity, for a better tour schedule.
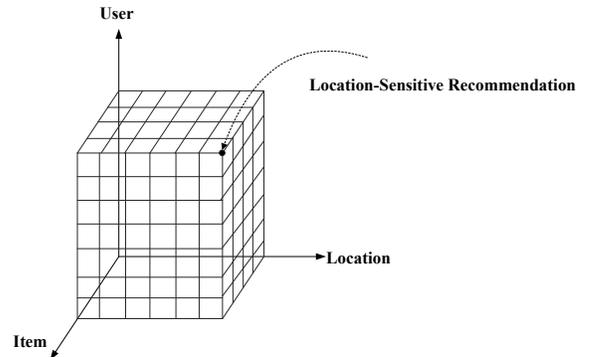

Fig. 1. Location-Sensitive Recommendation

In social recommendation, rating prediction and item recommendation are two main research issues. For example, for a new customer in E-commerce applications, how to efficiently predict his/her rating for a certain product and recommend some potential interesting products to him/her with social recommendation mechanism is a challenge issue. There has been a number of related work [4], [5], [11], [18], [19], [20], [21], [22] on rating prediction and social recommendation. Recently, Vasuki *et al.* [20] proposed affiliation/group recommendations based on the friendship network among users, and the affiliation/group network between users and groups. However, their method focused on path counts only and did not exploit other features and network characteristics which can be informative for link formation. In [19], [21], they proposed the recommendation systems with the incorporation of trust

[1] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
[2] Department of Computer Science, Inje University, Korea
[3] Department of Theoretical and Applied Sciences (DiSTA), University of Insubria, Italy
[4] College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter, U.K
[5] Department of Computer Science and Engineering, Arizona State University, USA
[6] Department of Computer Science, St. Francis Xavier University, Antigonish, Canada
[*] Corresponding author: L.T. Yang, Email:ltyang@gmail.com

and distrust information. The proposed framework was based on matrix factorization with regularization terms constraining the trust and distrust relations between users. Symeonidis *et al.* [4] proposed a method, namely Social Union, that combines similarity matrices derived from heterogeneous (user-item bipartite graph and user-user social graph) explicit or implicit social rating networks. Unfortunately, they did not consider the spatial features into their approach which cannot cope with the rating prediction and item recommendation in location-based (ad-hoc) social networks.

In this paper, we focus on generating the location-sensitive recommendations by rating prediction of items in ad-hoc social network environments and propose Spatial Social Union (SSU), an approach that combines multiple similarity matrices derived from user-item bipartite graph, user-user social graph, and user-location bipartite graph. SSU differs from the Social Union [4] because it takes into account not only the relation between user and item as well as the social relationships between users, but also the relationships between user and location. The major contributions of this paper are summarized as follows:

- First, three types of similarity matrices derived from user-item bipartite graph, user-user social graph, and user-location bipartite graph are provided and analyzed.
- Second, the similarity calculation approach, Spatial Social Union (SSU), that combines the three similarity matrices together is proposed.
- Third, we improve the FriendTNS algorithm [3] and devise the SSU-aware location-sensitive recommendation algorithm for items.
- Last, the proposed SSU-aware location-sensitive recommendation algorithm is evaluated using MovieLens dataset, which is a very popular movie recommendation service. We predict the rating of movie candidates and provide the top-15 recommended movies for the newly added user.

The remainder of this paper is structured as follows. Section II presents the related work on collaborative recommendation and location-based ad-hoc social networks. The problem statement and solution framework are given in Section III. Section IV presents the approach of similarity measurements and rating prediction based on user-item bipartite graph, user-user social graph and user-location bipartite graph, respectively. Then, the spatial social union is proposed based on the combination of similarity matrices induced from user-item bipartite graph, user-user social graph and user-location bipartite graph. Section V presents the proposed SSU-aware location-sensitive recommendation algorithm. Experimental results and analysis are shown in Section VI. Finally, Section VII concludes this paper.

II. RELATED WORK

A. Collaborative Recommendation

Generally, the collaborative recommendation systems are classified into two types: a content-based system and collaborative filtering. A content-based system usually selects items based on the correlation between the content of the items and the users' preferences [6]. Collaborative filtering systems are divided into two categories: memory-based and model-based. In the memory-based systems, the similarity between all users is calculated based on their ratings of items using some heuristic measures such as the cosine similarity and the Pearson correlation score. Then, a missing rating is predicted by aggregating the ratings of the k nearest neighbors of the user who need the recommendations. The model-based filtering systems assume that the users build up clusters based on their similar behavior in rating of items. Normally, a model needs to be learned based on the patterns recognized in the rating behaviors of users using clustering, Bayesian networks and other data mining techniques [7], [8]. The shortcoming of the model-based filtering system is that the poor knowledge of social networks and high training cost. Also, none of these collaborative filtering methods have been used to support database queries for spatial objects. Symeonidis *et al.* [4] proposed a generalized framework that exploits multimodal social networks to provide item recommendations in social rating networks. They proposed Social Union, a method that combines similarity matrices derived from heterogeneous explicit or implicit social rating networks. The Social Union has no any spatial properties which cannot cope with the recommendation in location-based social networks. Recently, a number of hybrid methods have been investigated. Yang *et al.* [32] presented a novel approach to improving recommendation accuracy by introducing the concept of inferred category-specific circles of friends. The idea is to determine the best subset of a user's friends, *i.e.*, an category-specific inferred circle, for making recommendations in an item category of interest. Ma *et al.* [33] considered the trust relationship between users and proposed a novel probabilistic factor analysis framework, which naturally fuses the users tastes and their trusted friends' favors together in recommender system. However, the possible diffusions of trusts between various users are not considered. Matrix Factorization has become the dominant technique for recommendation system for its ability to handle latent factors and also to accommodate additional information like biases, temporal dynamics and confidence level [22], [33], [34], [35]. In particular, Rendle [37] took the poor choice of good values for the regularization parameters into account and proposed a learning algorithm LibFM for matrix factorization model parameters. Their learning regularization parameters are as easy as learning model parameters and thus there is no need for any time-consuming search of regularization values because they are found on-the-fly.

B. Recommendation in Location-based Ad-hoc Social Networks

A Location-based Ad-hoc Social Network does not only mean adding a location to a general social network so that people in the social structure can share location-embedded information by their mobile devices, but also consists of the new social structure made up of individuals connected by the interdependency derived from their locations in the physical world as well as their location-tagged media content, such as photos, video, and texts [24], [25]. The emergence of

location-based social networking services offered by providers such as Rummble, GyPSii, and Whrrl is revolutionizing social networking, allowing users to share real-life experiences, to see where their friends are, to search location-tagged content within their social graph, and to meet others nearby [9], [28], [29]. Most of the existing location-based social networking systems focus on specific services: sharing geo-tagged message and supporting privacy-preserving buddy search [10], [12], [13]. Ludford *et al.* [14] studied how people shared the location knowledge through different location types using two small scale controlled experiments. Ye *et al.* [15] investigated the location recommendation services for large-scale location-based social networks, by exploiting the social and geographical characteristics of users and locations/places. However, they only solved the location recommendation issue based on collaborative ratings of places made by social friends, but not items recommendation. In many situations social relationships are ad-hoc (*i.e.*, set up by (mobile) users located in a limited geographical area during a certain period in time) [23]. For example, an appearance of a mobile user in a specified location during a specified period in time is often associated with a certain social event. However, with the dynamical structure and increased complexity of location-based ad-hoc social network data, a more generic model containing multiple node types (multi-modal), multiple edge types (multi-relational) and multiple descriptive features (multi-featured) associated with each should be proposed [30]. Scellato *et al.* [36] described and evaluated a link prediction model based on place properties of a location-based social networks via studying a large real-world service, *Gowalla*. Wang *et al.* [38] proposed Tracommender, a context-aware recommender system, which uses background tracking information from smart phones to generate location-based recommendations. Kurashima *et al.* [39] proposed a new topic model, called Geo Topic Model, that can jointly estimate both the user's interests and activity area hosting the user's home, office and other personal places that analyzes the location log data of multiple users to recommend locations to be visited. Sarwat *et al.* [40] considered the spatial features of the recommended items and proposed LARS*, a location-aware recommender system that uses location-based ratings to produce recommendations. In this paper, we mainly study the items recommendation in a limited geographical area within a given period, namely location-sensitive recommendation in ad-hoc social networks.

III. PROBLEM DEFINITION AND SOLUTION FRAMEWORK

This section describes the basic fundamental definitions of spatial social rating networks and presents the formulation of the problem to be addressed in this paper.

A. Spatial Social Rating Network

Definition 1. *(Spatial Social Rating Network) A Spatial Social Rating Network (SSRN) is formalized as a quadruple $\Omega =< V, E, \Lambda, \Gamma >$ with V indicating the users; E indicating the social relationships between users where $e_{ij} \in E$ denotes the relationship between i and j; Λ indicating a triple with user, item, and rating $\Lambda =< V, I, R^I >$; and Γ*
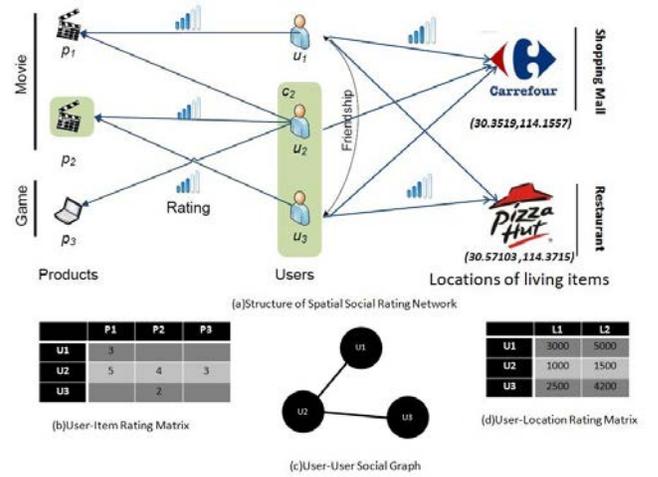

Fig. 2. Structure of a Spatial Social Rating Network

indicating a triple with user, location and rating (distance) $\Gamma =< V, L, R^L >$. $\Lambda =< V, I, R^I >$ means the users V give the ratings R^I to items I; $\Gamma =< V, L, R^L >$ means the ratings R^L obtained by calculating the distance between the users V and locations L.

Figure 2(a) illustrates the structure of an SSRN. The left part of Figure 2 shows that for a certain product p_2, there is a group of users u_2 and u_3 who have given ratings and published their comments on it. These users form a community c_2. In the right part of Figure 2, for a given shopping mall *Carrefour*, users u_1, u_2 and u_3 who have given ratings in terms of distance between users and location of shopping mall. These users also form a community $c_2 \bigcup u_1$. In other words, each product in an SSRN is associated with a community as well as each location in a SSRN is associated with a community [16]. Obviously, we refer to these communities as the potential *product community* and *local area community* respectively. Investigating the properties and users behavior in both *product community* and *local area community* is important to marketing and business analysts [17].

B. Problem Description and Formulation

In the real world, the process of location-sensitive recommendation scenario in ad-hoc social networks includes two central elements: the favors of these friends and the locations of the items which can essentially be modeled by the examples of the SSRN in Figure 2. As illustrated in the left part of Figure 2(a), user u_2 rated three items p_1, p_2 and p_3. In addition to the rating data, this user also maintains two relational information: friends list and ratings table on locations with his friends. The friends list stores all the social relationships between users. The ratings table on locations contains the distances information between users and locations.

By combing all the information from all users, we summarize three different data resources: the user-item rating matrix shown in Figure 2(b), the user social graph shown in Figure 2(c) and the user-location rating matrix shown in Figure 2(d). In particular, each user rates some items on a 5-point integer

scale to express the extent of the favor of each item (normally, 1, 2, 3, 4 and 5 represent "hate","don't like", "neutral", "like" and "love", respectively). On the other hand, the rating on the locations are measured according to the distance between user and location, such as the distance between user u_2 and location l_1 is 1000 km which indicates "*close*" semantically compared to distance between user u_2 and location l_1.

The problem we study in this paper is how to predict the missing values and make recommendations from the following two aspects:

- (**Single User**) For a single user, the goal of our items recommendation is to score all the items in the candidate sets and return the optimal related items. However, the rating prediction is to derive rating predictions for a specific user u we take into account the ratings of the top-m similar users to u. We can utilize these inter-dependencies to score the items using the probability $P(i|u,l)$. Further, the location-sensitive recommendation for a given user $u^{'}$ and a given location $l^{'}$ is formalized as follows,

$$\widetilde{I}(u^{'}, l^{'}) = argmax_{i \in I}^{n} P(i|u^{'}, l^{'}) \qquad (1)$$

where n is the number of recommended items and I is the collection of items in the training file.

- (**Group of Users**) For a group of users $\widetilde{U} = \{u_1^{'}, u_2^{'}, \cdots, u_n^{'}\}$, the location-sensitive recommendation problem is slightly different from the situation of single user because we should take into account the properties between item i and location l, denoted as C_{il} (such as distance, traffic situation between user's location and item's location, personalization on items) and obtain an overall recommendation. For simplicity, C_{il} is set as the numerical level, i.e, $C_{il} = \{1, 2, 3, 4, 5\}$. Intuitively, the bigger C_{il} is, the higher probability for recommendation. The location-sensitive recommendations for a group of users is formalized as follows,

$$\widetilde{I}(u_1^{'}, l^{'}) \bigcap \widetilde{I}(u_2^{'}, l^{'}) \bigcap \cdots \bigcap \widetilde{I}(u_N^{'}, l^{'})$$

$$\widetilde{I}(u_i^{'}, l^{'}) = argmax_{i \in I}^{n} P(i|u_i^{'}, l^{'}) C_{il} \qquad (2)$$

Obviously, we consider not only the rating of items but also the properties between item and location in this kind of recommendation. Hence, the recommended items are the intersection among the recommended items for each user.

This paper investigates the location-sensitive recommendations effectively and efficiently by employing the user-item bipartite graph, social graph and the user-location bipartite graph. Therefore, the proposed problem includes two aspects: rating prediction and recommendation.

C. Solution Framework

To overcome the drawbacks of items recommendation and rating prediction in generalized social networks, we investigate this problem in the location-based ad-hoc social networks, thus integrate the spatial property (location dimension) into generalized social networks and propose the Spatial Social
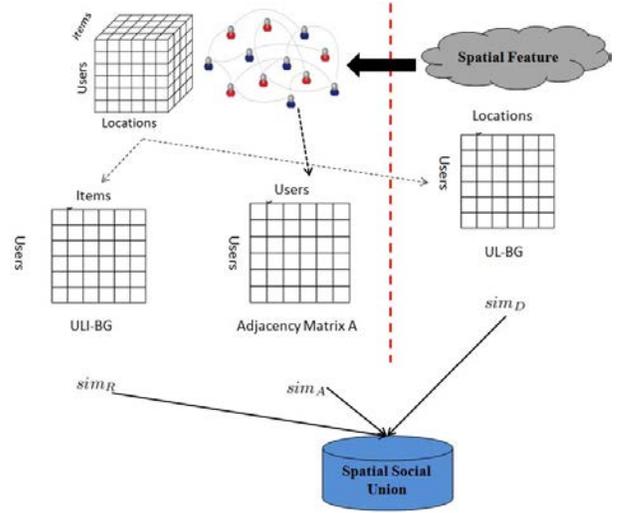

Fig. 3. The Solution Framework of Spatial Social Union

Union (SSU). The whole solution framework of SSU is depicted as follows.

The detailed working process of the solution framework includes the following steps: 1) Input data tensorization; 2) Projection of input data; 3) Similarity measurement; 4) Similarity aggregation; and 5) Rating prediction and recommendation.

- (**Input Data Tensorization**) As shown in Figure 3, the SSRN as an input, can be tensorized as a kind of tensor with three dimensions.
- (**Projection of Input Data**) Then, we make two projections on tensorized SSRN and derive the user-item bipartite graph (ULI-BG) and user-location bipartite graph (UL-BG), respectively. Besides, the user-user social graph (G) from the social networks is derived.
- (**Similarity Measurement**) Based on these derived graphs, similarity matrices between users can be constructed as sim_R, sim_A and sim_D.
- (**Similarity Aggregation**) Further, we propose an aggregation union, namely SSU which combines the various similarity matrices sim_R, sim_A and sim_D together and returns the similarity matrix between any two users.
- (**Rating Prediction and Recommendation**) At last, we adopt the finalized similarity matrix to predict the missing ratings and provide the the recommendations in terms of similarity.

Obviously, the left side of red line of Figure 3 shows the working mechanism of Social Union [4] that just considers the user-item rating matrix and user-user adjacency matrix. The main difference between our work and Social Union is that the spatial feature (*i.e,* location) is accommodated. The following three sections will present the detailed explanations of calculation on sim_R, sim_A and sim_D, respectively. Then, the SSU is proposed in Section IV.D. To enhance the readability, Table I lists the important notations used through this paper.

| Notation	Description
$G = (V, E)$	A social network with node set V, edge set E
N	The number of nodes in G
m	The number of edges in G
$d(v)$	The degree of node v
sim_R	The similarity matrix based on rating matrix
sim_A	The similarity matrix based on adjacency matrix
sim_D	The similarity matrix based on location matrix
$r_{x,i}$	The rating on item i given by user x
$p_{x,i}$	The predicted rating on item i given by user x
$d_{x,l}$	The rating on location l given by user x
$sim(u, v)$	The similarity between user u and v

TABLE I
NOTATIONS

-	Starbucks	YoYo
Alice	3	2
Bob	5	1
Charles	4	-

TABLE II
USER-ITEM MATRIX R

	Alice	Bob	Charles
Alice	1	0.9247	0.8321
Bob	0.9247	1	0.9806
Charles	0.8321	0.9806	1

TABLE III
RATING SIMILARITY MATRIX sim_R
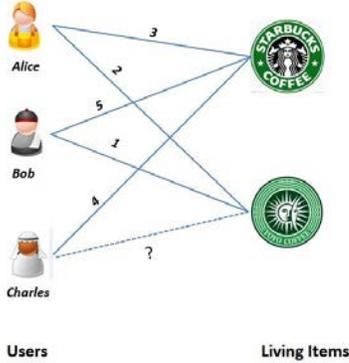
Fig. 4. User-item Bipartite Network

IV. SIMILARITY ANALYSIS AND MEASUREMENT BETWEEN USERS

This section firstly overviews the similarity analysis and measurement between users based on user-item bipartite graph and user-users social graph. Considering the spatial features in location-based ad-hoc social networks, we devise a similarity measurement scheme between users based on user-location bipartite graph. Based on the existing similarity matrices and our devised similarity matrix, an overall similarity calculation model, namely spatial social union (SSU) is proposed.

A. Similarity Measurement based on User-item Bipartite Graph

In both the general social networks and location-based social networks, users can give the various ratings to their concerned items. For example, users rate the quality-of-service and environment of coffee shops, restaurants and banks. Users can form several implicit social rating networks through their daily interactions. Therefore, we explain the rating prediction based on user-item bipartite graph (*ULI-BG*) in this section. Figure 4 presents an example of three users, *i.e, Alice, Bob, and Charles* who give various ratings to two famous coffee shops *i.e, Starbucks, YoYo*, namely, *items*. Here is a challenge issue: how to predict the rating on *YoYo coffee shop* given by *Charles*? A obvious method is to predict the rating with collaborative recommendation [1][2].

The *ULI-BG* of the above example can be also represented by a matrix R, where the rating of a user u over an item i is given by the element $R(u, i)$. An example of such a matrix

is shown in Table II, where *Alice, Bob, and Charles* are users and *Starbucks, YoYo* are items. The null cells (without rating) are denoted with dash.

We adopt the cosine similarity commonly used in collaborative recommendation [1], [2] to measure the similarity between two users u and v.

Definition 2. *(Cosine Similarity) For any two users u and v, the cosine similarity $sim(u, v)$ is represented as follows,*

$$sim(u, v) = \frac{\sum_{\forall i \in I}(r_{u,i} * r_{v,i})}{\sqrt{\sum_{\forall i \in I}(r_{u,i})^2}\sqrt{\sum_{\forall i \in I}(r_{v,i})^2}} \quad (3)$$

where $r_{x,i} = R(x, i)$.

Let us take Figure 4 as an example, the rating similarity matrix sim_R is constructed as follows,

To predict the rating of user *Charles* on *YoYo coffee shop*, we take into account the cosine similarity between users. Then, the predicted rating of *Charles* for *YoYo coffee shop* is calculated by using the following equation:

$$p_{u,i} = \frac{\sum_{v \in U}[sim(u, v) * r_{v,i}]}{\sum_{v \in U} sim(u, v)} \quad (4)$$

Based on Equation (4), the calculation for prediction of *Charles* on *YoYo coffee shop* is shown as follows:

$$p_{Charles, YoYo} = [sim(Charles, Alice) * r(Alice, YoYo)$$
$$+ sim(Charles, Bob) * r(Bob, YoYo)]/$$
$$(sim(Charles, Alice) + sim(Charles, Bob))$$
$$= (0.8321 * 2 + 0.9806 * 1)/(0.8321 + 0.9806) = 1.459 \quad (5)$$

B. Similarity Measurement based on User-User Social Graph

A user-user social graph is modeled as an undirected and un-weighted graph $G = (V, E)$, with the vertices in V representing the users in the social network and edges in E representing the relationships between users. In Figure 5, there exists a relationship between *Bob* and *Alice* as well as between *Alice* and *Charles*. This kind of relationships can be represented with *adjacency matrix* A. It is an $n \times n$ matrix, where $n = |v|$ is the number of nodes in G. Hence, the adjacency matrix of Table IV is shown as follows.

Here, we improve the existing similarity measurement method *FriendTNS* [3] and devise the *M-FriendTNS* similarity
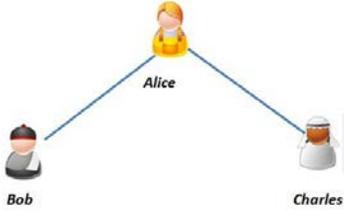
Fig. 5. User-User Social Graph

-	Alice	Bob	Charles
Alice	0	1	1
Bob	1	0	0
Charles	1	0	0

TABLE IV
USER-USER ADJACENCY MATRIX

measurement method to evaluate the optimal similarity between two users u and v based on the user-user topological relationships.

Definition 3. *(M-FriendTNS Similarity) For any two users u and v, the M-FriendTNS similarity $sim(u,v)$ is defined as follows,*

$$sim(u,v) = \begin{cases} 0 & if\ (u,v) \notin E \parallel u = v \\ 1 & if\ u = v \\ max \prod_{i=0}^{k} \frac{1}{d(u_i)+d(u_{i+1})-1} & otherwise. \end{cases}$$

(6)

where $d(u)$ and $d(v)$ are the degrees of nodes u and v, respectively. k is the hop count in the path from u to v. In particular, $u_0 = u$ and $u_{i+1} = v$. For non-adjacent nodes u and v, we simply multiply and maximize the similarity values between the intermediate nodes of the path counted by hops between u and v.

Let us take Figure 5 as an example, the *user similarity matrix* sim_A of G, which is an $n \times n$ matrix with n rows and n columns labeled by the graph nodes. sim_A of Figure 5 is constructed as follows,

Based on Equation (4), the calculation for prediction of *Charles* on *YoYo coffee shop* based on user similarity matrix is shown as follows:

$$p_{Charles,YoYo} = [sim(Charles, Alice) * r(Alice, YoYo)$$
$$+ sim(Charles, Bob) * r(Bob, YoYo)] /$$
$$(sim(Charles, Alice) + sim(Charles, Bob))$$
$$= (0.5 * 2 + 0.25 * 1)/(0.5 + 0.25) = 1.667 \quad (7)$$

-	Alice	Bob	Charles
Alice	1	0.5	0.5
Bob	0.5	1	0.25
Charles	0.5	0.25	1

TABLE V
sim_A USER SIMILARITY MATRIX

Fig. 6. The Map of Starbucks Coffee Shops Distribution in Wuhan City

C. Similarity Measurement based on User-Location Bipartite Graph

This section mainly discusses the approach for rating prediction based on user-location bipartite graph. Intuitively, if two users are in the same area or they are very close geographically, these two users are considered to be very similar. For example, in the real world, if two users live in the same apartment, and both of them like Starbucks and YoYo coffee shops and give the ratings for them about the quality of services. From this point, we consider that the similarity based on user's location is important for rating prediction.

As shown in Figure 6, most of *Starbucks Coffee* shops are near the *Zhongshan Park*, and some others are near the *East Lake*. Obviously, customers *Alice* and *Bob* are living close to the *Zhongshan Park*, *Charles* is living nearby the *East Lake*. There is an important assumption: the closer their living places are, the higher similarity between them is. For example, for two users who are living in the same area and they usually visit *Starbucks Coffee* shops; they are likely to have a common interest and taste on rating for that coffee shops.

Firstly, we present how to construct an user-location bipartite graph (*UL-BG*); Then, we calculate the similarity between two users based on *UL-BG*.

Definition 4. *(User-Location Bipartite Graph) An User-Location Bipartite Graph (*UL-BG*) is a graph whose vertices can be divided into two disjoint sets: users set U and locations set L such that every edge connects a vertex in U to one in L with a weight w; that is, U and L are independent sets, where $w = dis(u,l), u \in U, l \in L$, $dis(u,l)$ denotes the distance between user u and location l.*

Figure 7 presents a user-location bipartite graph as an example. There are two road marks about Starbucks Coffee Shop: *Zhongshan Park* and *East Lake* with longitude and latitude information $(114.1557, 30.3519)$, $(114.3715, 30.57103)$. Suppose there are three customers *Alice*, *Bob*, and *Charles*. The distance between *Zhongshan Park* and *Alice* is represented as $d(Alice, ZhongshanPark)$=5000.

Similarly, the *ULI-BG* of the above example can be also represented by a matrix D (shown in Table VI), where the distance from a user u to a location l is given from the element $d(u,i)$. An example of such a matrix is given in Table 7, where *Alice, Bob, and Charles* are users and *Zhongshan Park, East Lake* are locations.
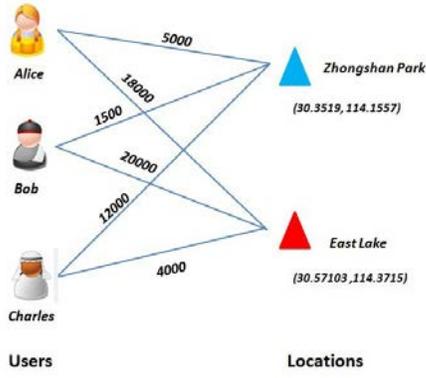
Fig. 7. User-Location Bipartite Graph

-	Zhongshan Park	East Lake
Alice	5000	18000
Bob	1500	20000
Charles	12000	4000

TABLE VI
USER-LOCATION MATRIX D

To calculate the similarity between users, we adopt the cosine similarity approach.

Definition 5. *(Cosine Similarity) For any two users u and v, the cosine similarity $sim(u, v)$ is represented as follows,*

$$sim(u,v) = \frac{\sum_{\forall l \in L}(d_{u,l} * d_{v,l})}{\sqrt{\sum_{\forall l \in L}(d_{u,l})^2}\sqrt{\sum_{\forall l \in L}(d_{v,l})^2}} \quad (8)$$

where $d_{x,l} = D(x, l)$.

Let us take Figure 7 as an example, the location similarity matrix sim_D is constructed as follows,

To predict the rating of user *Charles* on *YoYo coffee shop*, we consider the cosine similarity between users. Then, we calculate the predicted rating of *Charles* for *YoYo coffee shop* using following equation:

$$p_{u,i} = \frac{\sum_{v \in U}[sim(u,v) * r_{v,i}]}{\sum_{v \in U} sim(u,v)} \quad (9)$$

Based on Equation (9), the calculation for prediction of *Charles* on *YoYo coffee shop* is shown below.

$$p_{Charles,YoYo} = [sim(Charles, Alice) * r(Alice, YoYo)$$

$$+ sim(Charles, Bob) * r(Bob, YoYo)]/$$

$$(sim(Charles, Alice) + sim(Charles, Bob))$$

$$= (0.5586*2+0.3863*1)/(0.5586+0.3863) = 1.5912 \quad (10)$$

|	Alice	Bob	Charles
Alice	1	0.9808	0.5586
Bob	0.9808	1	0.3863
Charles	0.5586	0.3863	1

TABLE VII
LOCATION SIMILARITY MATRIX sim_D
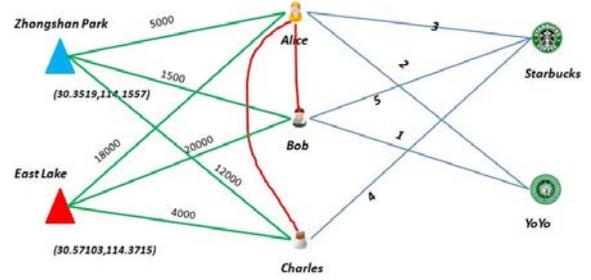
Fig. 8. A Case of Location-sensitive Recommendation in Ad-hoc Social Network

D. Spatial Social Union

In this section, we propose the Spatial Social-Union for location-based social networks which considers the spatial similarity feature by incorporating the user-location network in order to recommend and predict the rating for living-items efficiently. It provides the improvement based on Social Union [4] which just focuses on the user-item network and user-user friendship network.

Figure 8 presents a case of location-sensitive recommendation in ad-hoc social networks. The main difference between social recommendation and location-sensitive recommendation in ad-hoc social networks is that the user-location network is considered. Obviously, three types of similarity matrices: similarity matrix sim_R which is based on the user-item network, similarity matrix sim_A which is based on friendship network, and similarity matrix sim_D which is based on user-location network. To evaluate the similarity between two users u and v of a location-based ad-hoc social network, we combine sim_A, sim_R, sim_D matrices into a single similarity matrix.

$$sim(u,v) = \alpha * sim_A + \beta * sim_R + \gamma * sim_D \quad (11)$$

In Equation (11), α β, and γ are the weight parameters for calculating the similarity between two users u and v.

To determine the values of weight parameters, the least-square estimation method [41] is adopted.

$$sim(u,\overrightarrow{v}) = \alpha*sim_A(u,\overrightarrow{v})+\beta*sim_R(u,\overrightarrow{v})+\gamma*sim_D(u,\overrightarrow{v}) \quad (12)$$

where $\overrightarrow{v} = [v_1, v_2, \cdots, v_n]$ are different sampling points of similarity between users. $sim_A(u, \overrightarrow{v})$, $sim_R(u, \overrightarrow{v})$, and $sim_D(u, \overrightarrow{v})$ are the corresponding similarity values between the given user u and others. By collecting a set of training data $u_1, v_2, \cdots, v_m (m \le n)$, we have

$$\widehat{sim}(u,\overrightarrow{v}) = \widehat{sim} \times \widehat{W}$$

where $\widehat{sim}$ is a $3 \times m$ matrix:

$$\widehat{sim} = \begin{pmatrix} sim_A(u,v_1) & sim_R(u,v_1) & sim_D(u,v_1) \\ \vdots & \ddots & \vdots \\ sim_A(u,v_m) & sim_R(u,v_m) & sim_D(u,v_m) \end{pmatrix}$$

$$(13)$$

Here, $\widehat{W}$ is a $m \times 1$ output vector, i.e., $\widehat{W} = [\alpha, \beta, \gamma]^T$, and $\widehat{sim}(u, \overrightarrow{v})$ is a $m \times 1$ output vector, i.e., $\widehat{sim}(u, \overrightarrow{v}) = [\widehat{sim}(u, v_1), \cdots, \widehat{sim}(u, v_m))]^T$.

If $(\widehat{sim}(u, \overrightarrow{v}))^T \widehat{sim}(u, \overrightarrow{v})$ is nonsingular, then the following equation holds.

$$\widehat{W} = ((\widehat{sim}(u, \overrightarrow{v}))^T \widehat{sim}(u, \overrightarrow{v}))^{-1} (\widehat{sim}(u, \overrightarrow{v}))^T \widehat{sim}(u, \overrightarrow{v}) \tag{14}$$

Based on the weight parameters estimated from Eq.(14), we can calculate the similarity between any two users by applying Eq. (11).

V. RATING PREDICTION AND RECOMMENDATION WITH SPATIAL SOCIAL UNION

As for rating prediction for a specific user u, we take into account the ratings of the top-m similar users to u, where $m < n$ is a user-defined parameter. The predicted rating for user u can be estimated by the following Eq. (15).

Rating Prediction: Suppose $s_1, s_2, \cdots, s_m$ be the corresponding final similarity values of the top-m users $u_1, u_2, \cdots, u_m$ similar to u; the predicted rating for the user u are defined as follows:

$$\widehat{r}_{u,j} = avg + \frac{\sum_{i=1}^{m} s_i |r_{ij} - avg_i|}{\sum_{i=1}^{m} s_i} \tag{15}$$

$$s_i = f(\alpha * sim_R(u_i, u) + \beta * sim_A(u_i, u) + \gamma * sim_D(u_i, u)) \tag{16}$$

where j is any unrated item by user u; r_{ij} refers to the rating, and avg_i represents the average ratings value of user u_i in the ratings matrix R. avg denotes the average known ratings of the user u in the ratings matrix. $sim_R(u_i, u), sim_A(u_i, u)$, and $sim_D(u_i, u)$ indicate the user-living item, user-user, and user-location similarity matrices, respectively. $f(*)$ is a mapping function from $sim_R(u_i, u), sim_A(u_i, u)$, and $sim_D(u_i, u)$ to an overall similarity between u_i and u. It aggregates the three matrices to obtain the suitable similarity between u_i and u.

The SSU-aware location-sensitive recommendation based on rating prediction is described as follows.

Recommendation for Single User: Based on the scores $P(i|u, l)$, the set of recommended items for a given user u' and a given location l' will be $\widetilde{I}(u', l') = argmax_{i \in I}^n P(i|u', l')$ where n is the number of recommended items and I is the collection of items in the input data.

Recommendation for a Group of Users: Based on the scores $P(i|u, l)$ and the properties between user's location and item's location C_{il}, the set of recommended items for a group of user $\widetilde{U} = \{u'_1, u'_2, \cdots, u'_n\}$ and a given location l' will be $\widetilde{I}(u'_i, l') = argmax_{i \in I}^n P(i|u'_i, l')C_{il}$ where n is the number of recommended items and I is the collection of items in the input data.

First, two improved algorithms *M-FastFloyd* and M-*FriendTNS* are devised for quickly calculating the similarity matrix sim_A which is based on the friendship network. Then, the SSU-aware location-sensitive recommendation algorithm is provided by incorporating three similarity matrices sim_A, sim_R and sim_D.

A. M-FastFloyd Algorithm

Considering our problem scale as shown in Table VIII with Floyd Warshall algorithm, we modify and propose the Modified-FastFloyd (M-FastFloyd) algorithm that presents an efficient way to process the high dimensional matrix and vector elegantly. The M-Fast-Floyd is depicted in Algorithm 1. It takes matrix M and D as input and output correspondingly. In Line 3, $i2k$ is a matrix that replicates and tiles matrix M by row growth 1 and columns growth n for all the rows and column k. In Line 4, $k2j$ is a matrix that replicates and tiles matrix M by rows growth n and column growth 1 for row k and all the columns. In Line 5, D is the output matrix that gets the maximum among matrix M and element wise multiply matrix $i2k$ with matrix $k2j$.

Algorithm 1 Modified-Fast-Floyd Algorithm

Input:
    matrix *M*
Output:
    matrix *D*
1: $n \leftarrow$ row number of *M*
2: **for** *k* = 1 to *n*
3:     $i2k \leftarrow$ **replicate**(*M[:, k]*, 1, *n*)
4:     $k2j \leftarrow$ **replicate**(*M[K, :]*, *n*, 1)
5:     *D* = max(*M*, *i2k* element-wise multiply *k2j*)
6: **end for**
7: **return** *D*

B. M-FriendTNS Algorithm

From the perspective of ad-hoc social networks, we subtly assemble the specific group of awareness of the users who can share their opinions or attitudes for the items. Through the perspective of human factors the user oriented algorithm design is rather critical to devise the appropriate recommendation that matches the users requirement. In the sense that each individual has his or her own busy affairs, the algorithm should mine the implicit valuable information, e.g. the rating score of items from the awareness and relationship among the ad-hoc social network users. Therefore, we propose and design the Modified-FriendTNS (M-FriendTNS) algorithm as shown in Algorithm 2.

Lines 4 to 8 of the algorithm are for the users who are directly connected or have the relationship of first connection in the ad-hoc social network based on the assumption that it is symmetric and undirected graph. Lines 13 to 21 are to store the corresponding row and column index for the users who do not have the direct first connection. Lines 22 to 25 apply Algorithm 1 to calculate the maximum value from the user-user similarity matrix. Lines 26 to 30 flip up the lower triangular matrix by the principal diagonal.

C. SSU-aware Location-Sensitive Recommendation Algorithm

The most important point is to consider the location related information, e.g. distance, longitude and latitude. The proposed SSU-aware location-sensitive recommendation algorithm is depicted in Algorithm 3 that is powerful especially for the designated local geographical group of distinguished users in the ad-hoc social network environments.

Algorithm 2 M-FriendTNS: Modified-FriendTNS Algorithm

Input:
    user-user relationship matrix A
    the number of users N
Output:
    sim_A
 1: A = loadFile(path)
 2: $sim_A \leftarrow N \times N$ zeros matrix
 3: user degree: d $\leftarrow$ summation of matrix A's column
 4: **for** i = 1 to N
 5:    **for** j = 1 to N
 6:       **if** $A[i,j]$ == 1
 7:          $sim_A[i,j]$ = 1 /($d[i]$ + $d[j]$ - 1)
 8:       **end if**
 9:    **end for**
10: **end for**
11: $idx \leftarrow$ zeros matrix with the size of *[N*N, 2]*
12: $cnt \leftarrow 0$
13: **for** i = 1 to N
14:    **for** j = 1 to N
15:       **if** $A[i,j]$ == 0 **then**
16:          cnt++
17:          $idx[cnt][1]$ = i
18:          $idx[cnt][2]$ = j
19:       **end if**
20:    **end for**
21: **end for**
22: D = **ModifiedFastFloyd**(sim_A)
23: **for** i = 1 to cnt
24:    $sim_A[idx[i][1], idx[i][2]] = D[idx[i][1], idx[i][2]]$
25: **end for**
26: **for** i = 1 to N
27:    **for** j = 1 to N
28:       $sim_A[i][j] = sim_A[j][i]$
29:    **end for**
30: **end for**

Algorithm 3 SSU-Aware Location-Sensitive Recommendation Algorithm

Input:
    user-item rating matrix R
    user-user relationship matrix A
    user-location metric matrix D
    the number of users N
    a group of users $\widetilde{U} = \{u'_1, u'_2, \cdots, u'_n\}$
    property between item i and location l C_{il}
    a given targeted location l'
    type of recommendation ID Z
Output:
    rating prediction vector P for new user and a group of users
 1: **switch**(Z)
 2: **begin**
 3: **case** 1: /*recommendation for a single user*/
 4: $sim_R \leftarrow$ apply cosine similarity in R
 5: $sim_A \leftarrow$ apply M-FriendTNS in A
 6: $sim_D \leftarrow$ apply cosine similarity in D
 7: $sim_{SSU} = \alpha * sim_A + \beta * sim_R + \gamma * sim_D$
 8: $sim \leftarrow$ extract column 2 to the end of sim_{SSU}
 9: $P \leftarrow R * transpose\,of\,sim/(\sum sim)$
10: **if** $user$ has given some ratings **then**
11:    calculate the prediction except the given ratings
12: **else then**
13:    calculate the prediction for the brand new $user$
14: **end if**
15: make the recommendation for the new $user$
16: **break**
17: **case** 2: /*recommendation for a group of users*/
18: output=$\widetilde{I}(u'_1, l') \bigcap \widetilde{I}(u'_2, l') \bigcap \cdots \bigcap \widetilde{I}(u'_N, l')$
    $\widetilde{I}(u'_i, l') = argmax^n_{i \in I} P(i|u'_i, l')C_{il}$
19: **end**

The input of the algorithm includes the user-item rating matrix R, user-user relationship matrix A, user-location metric matrix D, and the number of users N which involves the newly added user, property between item and location C_{il}, a given targeted location l', and type of recommendation Z. The output is the rating prediction and recommendation for the new user as well as a group of users. Lines 4 to 6 adopt the classical cosine similarity calculation and our proposed M-FriendTNS algorithm. Line 7 is the procedure of adapting where the parameters can be tuned manually or automatically. Line 9 applies our SSU model to solve the predicted rating scores. Lines 10 to 15 make the prediction based on the hypothesis that the new user who really wants some recommendations, e.g. he or she does not know the related quality of all the items. Lines 17 to 19 are to recommend the items for a group of users by taking the property C_{il} into account.

VI. EXPERIMENTS AND PERFORMANCE ANALYSIS

This section is to evaluate the proposed SSU-aware location-sensitive recommendation algorithm using a real-world dataset. First, we study how to collect the dataset and config the experiment setup. Then, the analysis and discussion of the experimental results are presented.

A. Data Set Collection

Movies are typical items in our daily life, so we adopt the real movies data set from MovieLens[1] which was collected by the GroupLens[2] Research Project. This data set consists of: (1) $100,000$ ratings (1...5) from 943 users on 1682 different numbered movies; (2) each user has rated at least 20 different movies and some demographic information for the users, e.g. age, gender, occupation, zip etc. The users and movie items are numbered consecutively from 1, and the original data is randomly ordered as the format of $user\_id|item\_id|rating\_score|timestamp$. The movie information includes the original format of $movie\_id|movie\_title|release\_date|type$.

However, the above data set does not provide the user-user social relationship graph in ad-hoc social networks.

Argument	Value
Users number	943
Items number	1,682
Rating score Range	1...5
Properties between item and location Range	1...5
Items text representation	$Title + Date$
Relational user ID range	$1...49,290$
Distance range	$1...50,000$ (km)

TABLE VIII
PARAMETERS PRE-DEFINE SETTINGS AND DESCRIPTIONS

Fig. 9.   Number of Rating per User in User-Item Graph

Fig. 10.   The Data Visualization of Users and Users Relationship Distribution

Considering the newly added user $Charles$ whom we make prediction and recommendation with, so we have 944 users. Moreover Epinions[3] includes the trust information of $49,290$ users, each trust information represented as $source\_user\_id|target\_user\_id|trust\_statement\_value$, $source\_user\_id$, $target\_user\_id$ are in $[1...49,290]$, and also in the dataset there are only positive trust statements and not negative ones (distrust), i.e. $trust\_statement\_value$ is always 1.

As mentioned before, we take into account the properties between item and location, i.e, C_{il} in order to evaluate the feasibility of the proposed location-sensitive recommendation for a group of users in ad-hoc social network environments. According to the traffic situation, distance between item and user, we category C_{il} with numerical level $C_{il} = \{1, 2, 3, 4, 5\}$ accordingly.

B. Experiment Design And Analysis

From the analysis of the real user-user relationship data set of Epinions, we found that: (1) the mathematical set of $target\_user\_id$ is not equal to the set of $source\_user\_id$. In other words, there exist some $target\_user\_id$ that does not appear in $source\_user\_id$; (2) the distinguished $target\_user\_id$ is $33,960$ among all the $487,182$ line records that are disorder. At the same time, in this paper we consider that: (1) our user-user relationship undirected graph is symmetric; (2) in our case we have 944 users including one new user $Charles$; (3) the user numbered order in MovieLens is different from Epinions.

Since the users in MovieLens and Epinions dataset are generic without special requirements, we consider a common group of users and their relationships between MovieLens and Epinions data set. We design and analyze the mapping user and preprocessing the data as follows:

1) sort the each line record by $source\_user\_id$, then by $target\_user\_id$;
2) count the distinguished element number and store in set_a with the $source\_user\_id$;
3) randomly pick 944 ones from $33,960$ discriminated keys by uniform distribution ;
4) number the 944 ones which are in range of $[1...49,290]$ consecutively from 1;
5) find the ones exist in $source\_user\_id$ and numbered it respectively among the $target\_user\_id$;
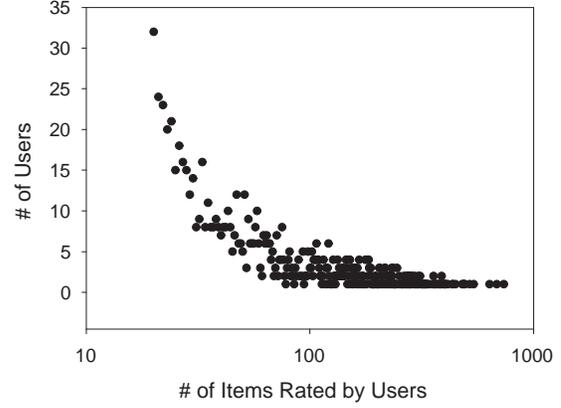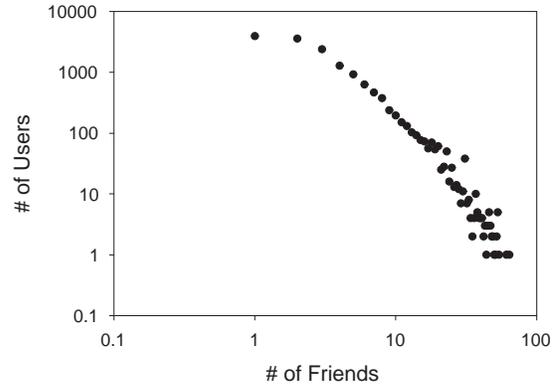
6) flip up the lower triangular matrix based on the principal diagonal;
7) write the result by represent each line with triple elements in the output file.

In order to illustrate the relationship between user and item, we removed the $timestamp$ field and sorted each line by $user_id$. This relationship is displayed in Figure 9.

Furthermore, the $movie\_id$, $movie\_title$, $movie\_release\_date$ are extracted to represent the item that we want to make prediction and recommendation for the new user $Charles$. Based on Equations (3) and (4), via the user-item matrix we calculate the ULI-BG similarity matrix and get the top K average ratings to the new user $Charles$. Based on the proposed M-Fast-Floyd algorithm and M-FriendTNS algorithm, we calculate the M-FriendTNS similarity matrix and corresponding prediction ratings. Finally, about the location information especially the distance within a city or a specific zone, e.g., a common city's radius is around 50,000m. By this heuristic information we generate random 1,000m to 50,000m from the uniform distribution for the UL-BG, i.e. user location matrix, then we calculate the similarity matrix and corresponding prediction ratings. The real social relationships between users extracted from Epinions is depicted in Figure 10.

---

[3]http://www.epinions.com

ID \ parameters	α	β	γ
1	0.2	0.4	0.4
2	0.4	0.3	0.3
3	0.6	0.2	0.2
4	0.8	0.1	0.1

TABLE IX
COMBINATION OF α, β, γ FOR EVALUATION

ID	movie's name	time	rating	properties C_{il}
1	Star Wars	1977	2.8	2
2	Fargo	1996	2.3	2
3	Return of the Jedi	1983	2.2	1
4	Contact	1997	2.1	2
5	Raiders of the Lost Ark	1981	2.0	3
6	The Godfather	1972	1.9	2
7	The English Patient	1996	1.8	1
8	Pulp Fiction	1994	1.8	3
9	Scream	1996	1.7	2
10	The Empire Strikes Back	1980	1.7	1
11	Air Force One	1997	1.6	2
12	Liar Liar	1997	1.6	3
13	Independence Day (ID4)	1996	1.6	2
14	Titanic	1997	1.6	1
15	Jerry Maguire	1996	1.6	3

TABLE X
TOP-15 RECOMMENDED MOVIES FOR *Charles*

C. Analysis and Discussions of Experiment Results

During the experiments, various combinations of α, β, γ are obtained by training them with least square estimation method, and are used to evaluate the top-K average ratings for different algorithms. Table IX lists the training sets of α, β, γ in SSU. The robust experimental results are shown in Figure 11. Our SSU model is the proximity curve with UL-BG model and M-Friend model, which means firstly we consider the location information and then the user-user proximity relationship in the ad-hoc social network, not the whole or global social network information.

The SSU-based top-15 recommended movies for newly added user "*Charles*" with $\alpha = 0.8, \beta = 0.1, \gamma = 0.1$ are presented in Table X.

The performance curve of the proposed SSU model is among the UL-BG model curve and M-FriendTNS model curve when the parameters α, β and γ are as shown in Table IX. Moreover, the slightly differences among the four different results in Figure 11 demonstrate the robustness of our SSU-aware location-sensitive recommendation algorithm.

We also make comparison to Spatial Social Union recommendation algorithms with Social Union, in terms of precision and recall. This reveals the robustness of each algorithm in attaining high recall with minimal losses in terms of precision. We examine the top-N ranked list, which is recommended to a test user, starting from the top item.

For the MovieLens data set (N is between $[1, \cdots, 15]$), the experimental statistics illustrate that the precision of each algorithm falls as N increases. In contrast, as N increases, recall for all algorithms increases in general. Note that the proposed SSU algorithm has the same recommendation performance with SU algorithm because they have the same precision-recall curve. Importantly, SSU incorporates the spatial factors
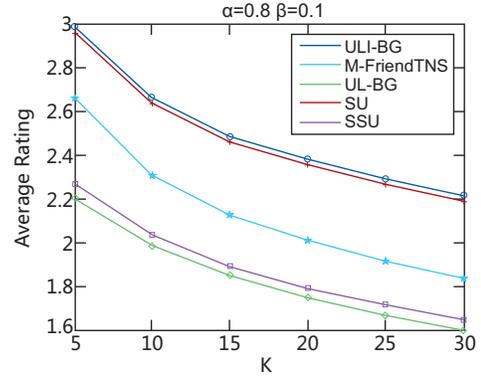

(a) $\alpha = 0.8, \beta = 0.1$
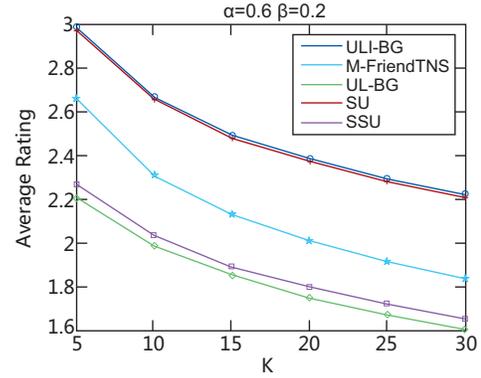
(b) $\alpha = 0.6, \beta = 0.2$
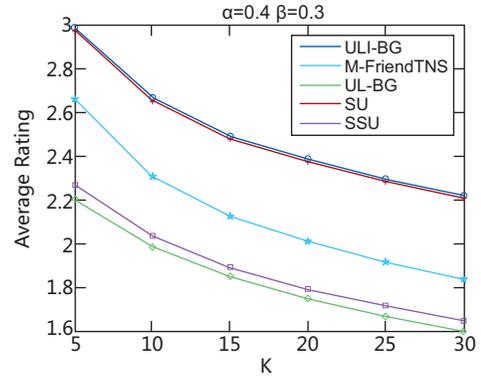
(c) $\alpha = 0.4, \beta = 0.3$
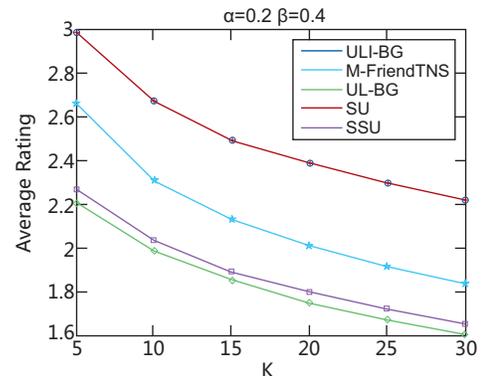
(d) $\alpha = 0.2, \beta = 0.4$

Fig. 11. Average rating for various algorithms in terms of k with various combinations of α, β
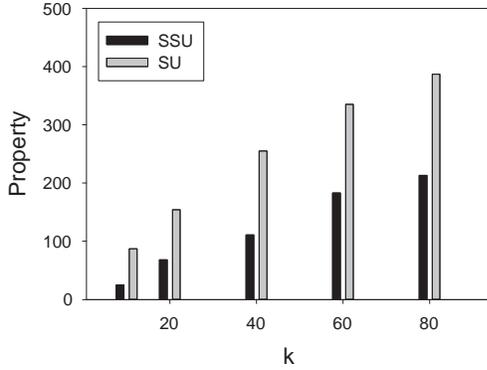
Fig. 12. Performance Comparison for Various Size of Group of Users in Terms of Properties C_{ij}

into the recommendation mechanism for the location-sensitive recommendation. This experiment shows that SSU algorithm is more practical in generating the location-sensitive recommendation for the users.

In this section, additional experiments are conducted to evaluate the performance of the proposed algorithm SSU with the goal of validating the effectiveness of the proposed algorithm for generating the location-sensitive recommendations for a group of users. The properties C_{il} between item i and location l, as a critical evaluation metric, can reflect the sustainability of the recommendation mechanism.

Firstly, it is obvious to obtain that the sum of properties $C_{il} = 30$ are greatly less than that of SU algorithm $C_{il} = 105$ for a single user case. Secondly, Figure 12 illustrates the performance comparison for various size of group of users ($k = 10, 20, 40, 60, 80$) in terms of C_{il}. For example, if k=10, the sum of properties $property$ is calculated as follows,

$$property = \sum_{j=1}^{10} C_{il}^j \qquad (17)$$

where C_{il}^j refers to the properties between item i and location l for user j.

As shown in Figure 12, the proposed recommendation algorithm SSU can significantly reduce the value of the properties C_{il} compared to SU recommendation algorithm. This is to say, our proposed recommendation algorithm can save the energy and travelling time from the user's location to item's location.

From the sustainability point of view, SSU recommendation algorithm can save the energy and shorten the travelling distance between item and location rather than SU algorithm. Our SSU model extracts the collective intelligence from location-based quantitative information and important human factors in the ad-hoc social networks. Furthermore, based on our proposed prototype the requisite usability testing is quite important for improving the ease-of-use of the corresponding system. While the algorithm presented here is a fundamental technique to improve accuracy of recommendations, however, the ultimate effectiveness of recommender systems go beyond the quality of the algorithm. To build the systems acceptable to users, it is also necessary to understand users. At the next stage, the development of algorithms should work together with issues related to user perspectives such as trust and distrust, explanations of recommendations, system transparency, recommender interface design as well as user requirements. In particular, the algorithm can even provide more intelligent and more accurate solutions when one understands and analyzes users' behaviors and preferences thoroughly.

VII. Conclusion

In this paper, we aim to investigate the rating prediction and generate location-sensitive recommendations in ad-hoc social networks. We present Spatial Social Union (*SSU*), an approach that combines three types of similarity matrices derived from user-item bipartite graph, user-user social graph as well as user-location bipartite graph. Further, the SSU-aware location-sensitive recommendation algorithm is devised. We evaluate and compare the proposed approach to the existing rating prediction and item recommendation algorithms with a real-life dataset. Experimental results show that our SSU algorithm is more effective in predicting rating of items and recommending items in location-based ad-hoc social networks. As the dramatic growth of online social network sites continues, the social recommendation in location-based ad-hoc social networks is widely used everywhere. From a social sustainable perspective, we plan to develop similar techniques in other urban sustainable applications, *e.g.* E-health field, to confirm that our approach is universally applicable in various domains.

References

[1] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl: Item-based Collaborative Filtering Recommendation Algorithms. *In Proc. WWW'01*, pages 285–295, 2001.
[2] R. McLauglin and J. Herlocher: A Collaborative Filtering Algorithm and Evaluation Metric that Accurately Model the User Experience. *In Proc. SIGIR'04*, pages329–336, 2004.
[3] P. Symeonidis, E. Tiakas, and Y. Manolopoulos. Transitive node similarity for link prediction in social networks with positive and negative links. *In Proc. 4th ACM RecSys'10*, pages 183–190, 2010.
[4] P. Symeonidis, E. Tiakas, Y. Manolopoulos: Product Recommendation and Rating Prediction based on Multi-modal Social Networks, *In Proc. ACM RecSys'11*, 2011.
[5] C. Gentile, S. Li and G. Zappella: Online Clustering of Bandits. *In the Journal of Machine Learning Research (JMLR), Workshop and Conference Proceedings, Vol. 32, Proc. of ICML 2014.*
[6] A. M. Ferman, J. H. Errico, P. van Beek, and M. I. Sezan. Content-based Filtering and personalization using structured metadata, *In JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 393–393, 2002.
[7] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *In IEEE Transactions on Knowledge and Data Engineering*,17(6), pages 734–749, 2005.
[8] H. Yildirim and M. S. Krishnamoorthy. A random walk method for alleviating the sparsity problem in collaborative filtering. *In Proc. of the 2008 ACM conference on Recommender systems*, pages 131–138, 2008.

[9] T. Bhuiyan, Y. Xu and A. Josang. Integrating Trust with Public Reputation in Location-based Social Networks for Recommendation Making, *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 107–110, 2008.

[10] Y. Cai and T. Xu. Design, Analysis, and Implementation of a Large-Scale Real-Time Location-Based Information Sharing System. *In Proc. of ACM MobiSys'08*, pages 106-117, 2008.

[11] S. Li, F. Hao, M. Li, and H. Kim. Medicine Rating Prediction and Recommendation in Mobile Social Networks. *In Proc. of Springer LNCS (Lecture Notes in Computer Science), the 8th International Conference on Grid and Pervasive Computing and Colocated Workshops*, Vol. 7861, pages 216-223, 2013.

[12] A. Khoshgozaran and C. Shahabi. Private Buddy Search: Enabling Private Spatial Queries in Social Networks. *In Proc. of SIN'09*, pages 166-173, 2009.

[13] L. Siksnys, J. Thomsen, S. Saltenis, and M. L. Yiu. Private and Flexible Proximity Detection In Mobile Social Networks. *In Proc. of MDM'10*, pages 75–84, 2010.

[14] P.J. Ludford, R. Priedhorsky, K. Reily, and L. Terveen. Capturing, sharing, and using local place information. *In Proc. of CHI'07*, pages 1235–1244, 2007.

[15] M. Ye, P. Yin, W. Lee. Location recommendation for location-based social networks. *In Proc. of GIS'10*, pages 458–461, 2010.

[16] Y. Matsuo and H. Yamamoto. Community gravity: measuring bidirectional effects by trust and rating on online social networks. *In Proc. of WWW'09*, pages 751–760, 2009.

[17] H. Li, S. Bohowmick, A. Sun. AffRank: Affinity-Driven Ranking of Products in Online Social Rating Networks. *In Journal of the American Society for Information Science and Technology*, 62, pages 1345–1359, 2011.

[18] J. Golbeck. Personalizing Applications through Integration of Inferred Trust Values in Semantic Web-based Social Networks. *In Proc. of Semantic Network Analysis Workshop at the 4th International Semantic Web Conf.*, 2005.

[19] M. Jamali and M. Ester. A Matrix Factorization Technique with Trust Propagation for Recommendation in Social Networks. *In Proc. of ACM RecSys'10*, pages 135–142, 2010.

[20] V. Vasuki, N. Natarajan, Z. Lu, and I. Dhillon. Affiliation Recommendation using Auxiliary Networks. *In Proc. of ACM RecSys*, pages 103–110, 2010.

[21] H. Ma, M. R. Lyu, I. King. Learning to Recommend with Trust and Distrust Relationships. *In Proc. of ACM RecSys*, pages 189–196, 2009.

[22] H. Ma, Dengyong Zhou, C. Liu, M. R. Lyu, I. King. Recommender Systems with Social Regularization. *In Proc. of WSDM*, pages 287–296, 2011.

[23] V. Podobnik, I. Lovrek. An Agent-Based Platform for Ad-Hoc Social Networking. *Agent and Multi-Agent Systems: Technologies and Applications Lecture Notes in Computer Science* 6682, pages 74–83, 2011.

[24] Y. Zheng. Location-based social networks: Users. *In Computing with Spatial Trajectories, Zheng, Y and Zhou, X, Eds. Springer*, 2011.

[25] Y. Zheng. Tutorial on Location-Based Social Networks. *In Proc. of WWW'12*, 2012.

[26] J. Tang, S. Wu, J. Sun, and H. Su. Cross-domain Collaboration Recommendation. *In Proc. of KDD'12*, pages 1285–1293, 2012.

[27] P. Pu , L. Chen , R. Hu. A user-centric evaluation framework for recommender systems. *In Proc. of of ACM RecSys'11*, pages 157–164, 2011.

[28] M. Sarwat , J. Bao , A. Eldawy , J. Levandoski , A. Magdy , F. Mokbel. Sindbad: a location-based social networking system. *In Proc. of ACM SIGMOD'12*, pages 649–652, 2012.

[29] Y. R. Lin, J. Sun, H. Sundaram, A. Kelliher, P. Castro, and R. Konuru. Community Discovery via Metagraph Factorization. *In ACM TKDD*, pages 17–17, 2011.

[30] E. C.-H. Ngai, M. B. Srivastava, and J. Liu. Context-Aware Sensor Data Dissemination for Mobile Users in Remote Areas. *In Proc. of INFOCOM'12*, pages 2711–2715, 2012

[31] M. Jamali, T. Huang, and M. Ester. A Generalized Stochastic Block Model for Recommendation in Social Rating Networks. *In Proc. of ACM RecSys'11* , pages 53–60, 2011.

[32] X. Yang, H. Steck, and Y. Liu. Circle-based recommendation in online social networks, *In Proc. KDD'12*, pages 1267–1275, 2012.

[33] H. Ma, I. King, and M.R. Lyu. Learning to recommend with social trust ensemble, *In Proc. SIGIR'09*, pages 203–210, 2009.

[34] H. Ma, H. Yang, M.R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization, *In Proc. CIKM'08*, pages 931–940, 2009.

[35] K. Zhou, S. Yang, and H. Zha. Functional matrix factorizations for cold-start recommendation, *In Proc. SIGIR'11*, pages 315–324, 2011.

[36] S. Scellato, A. Noulas and C. Mascolo. Exploiting Place Features in Link Prediction on Location-based Social Networks, *In Proc. SIGKDD'11*, pages 1046–1054, 2011.

[37] S. Rendle. Learning recommender systems with adaptive regularization, *In Proc. WSDM'12*, pages 133–142, 2012.

[38] Y. Wang, A. Uzun, U. Bareth, and A. Kupper. Tracommender-Exploiting Continuous Background Tracking Information on Smartphones for Location-Based Recommendations, *in Proc. MOBILWARE'12*, pages 250–263, 2012.

[39] T. Kurashima, T. Iwata, T. Hoshide, N. Takaya, and K. Fujimura. Geo topic model: joint modeling of user's activity area and interests for location recommendation, *in Proc. WSDM'13*, pages 375–384, 2013.

[40] M. Sarwat, J.J. Levandoski, A. Eldawy, M.F. Mokbel. LARS*: An Efficient and Scalable Location-Aware Recommender System, *In IEEE TKDE*, 2013.

[41] I. Yeo, C.C. Liu, and E. J. Kim, "Predictive dynamic thermal management for multicore systems", *In Proc. ACM/IEEE Design Automation Conference, 2008. DAC 2008*, pages 734–739, 2008.

Fei Hao is an assistant professor in Shanxi Agricultural University. Prior to that, he was a distinguished researcher at Huazhong University of Science and Technology. He received the B.S. and M.S. degrees in School of Mathematics and Computer Engineering from Xihua University, Chengdu, China. He completed his Ph.D course work from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea. He has published over 40 research papers in International and National Journals as well as conferences. His research interests include social computing and big data mining.

Shuai Li is a Ph.D. researcher in Computer Science and Computational Mathematics at DiSTA (Department of Theoretical and Applied Sciences), University of Insubria (Italy). He also serves as a research member at Department of Computer Science and Technology at Tsinghua University (China), and Telefonica Research Laboratories (Spain). Prior to that, he earned the M.Sc. degree in Computer Science from Inje University (Korea) and worked for U.S. Bell Laboratories about one year. His main research focuses on Large-Scale Machine Learning, Data Mining and Artificial Intelligence.

Geyong Min is a Professor of High Performance Computing and Networking in the Department of Mathematics and Computer Science within the College of Engineering, Mathematics and Physical Sciences at the University of Exeter, United Kingdom. He received the PhD degree in Computing Science from the University of Glasgow, United Kingdom, in 2003, and the B.Sc. degree in Computer Science from Huazhong University of Science and Technology, China, in 1995. His research interests include Future Internet, Computer Networks, Wireless Communications, Multimedia Systems, Information Security, High Performance Computing, Ubiquitous Computing, Modelling and Performance Engineering.

Hee-Cheol Kim received the MS degree in computer science from SoGang University in 1991, and the PhD in computer science from Stockholm University/Royal Institute of Technology, Sweden in 2001. He is currently an associate professor at the department of computer engineering in Inje University, Korea. His research interests are in the areas of human computer interaction, data mining and social computing related to wellness and healthcare. He has published more than 50 papers concerning these areas.

Stephen S. Yau is Professor of Computer Science and Engineering and the director of Information Assurance Center at Arizona State University, Tempe, Arizona, USA. His current research is in cloud and services computing systems, cyber security, software engineering, and social networks. He received the B.S. degree from National Taiwan University, Taipei, the M.S. and Ph.D. degrees from the University of Illinois, Urban, all in electrical engineering. He served as the president of IEEE Computer Society and on the boards of directors of the IEEE and Computing Research Association. He is a Life Fellow of the IEEE and a Fellow of American Association for the Advancement of Science.

Laurence T. Yang received the B.E. degree in Computer Science and Technology from Tsinghua University, China, and the Ph.D. degree in Computer Science from University of Victoria, Canada. Prof. Yang is affiliated with School of Computer Science and Technology, Huazhong University of Science and Technology, China, as well as with Department of Computer Science, St. Francis Xavier University, Canada. His research interests include parallel and distributed computing, embedded and ubiquitous/pervasive computing, big data. He has published more than 220 papers in various refereed journals (about 40Transactions/Journals and the others mostly in Elsevier, Springer, and Wiley Journals). His research has been supported by the National Sciences and Engineering Research Council of Canada (NSERC) and the Canada Foundation for Innovation.