

# Urban Point-of-Interest Recommendation by Mining User Check-in Behaviors

Josh Jia-Ching Ying, Eric Hsueh-Chan Lu, Wen-Ning Kuo and Vincent S. Tseng\*

Institute of Computer Science and Information Engineering

National Cheng Kung University

No.1, University Road, Tainan City 701, Taiwan (R.O.C.)

{jashying, ericlu416, p76994165}@gmail.com, \*Correspondence: tsengsm@mail.ncku.edu.tw

## ABSTRACT

In recent years, researches on recommendation of urban Points-Of-Interest (POI), such as restaurants, based on social information have attracted a lot of attention. Although a number of social-based recommendation techniques have been proposed in the literature, most of their concepts are only based on the individual or friends' check-in behaviors. It leads to that the recommended POIs list is usually constrained within the users' or friends' living area. Furthermore, since context-aware and environmental information changes quickly, especially in urban areas, how to extract appropriate features from such kind of heterogeneous data to facilitate the recommendation is also a critical and challenging issue. In this paper, we propose a novel approach named *Urban POI-Mine (UPOI-Mine)* that integrates location-based social networks (LBSNs) for recommending users urban POIs based on the user preferences and location properties simultaneously. The core idea of UPOI-Mine is to build a regression-tree-based predictor in the normalized check-in space, so as to support the prediction of interestingness of POI related to each user's preference. Based on the LBSN data, we extract the features of places in terms of i) *Social Factor*, ii) *Individual Preference*, and iii) *POI Popularity* for model building. To our best knowledge, this is the first work on urban POI recommendation that considers social factor, individual preference and POI popularity in LBSN data, simultaneously. Through comprehensive experimental evaluations on a real dataset from Gowalla, the proposed UPOI-Mine is shown to deliver excellent performance.

## Categories and Subject Descriptors

I.2.6 [Artificial Intelligence]: Learning; J.4 [Computer

Applications]: Social and Behavior Sciences

## General Terms

Measurement, Experimentation.

## Keywords

Point-Of-Interest Recommendation, Urban Computing, Data Mining, Location-Based Social Network, User Preference Mining.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*UrbComp'12*, August 12, 2012, Beijing, China.

Copyright 2012 ACM 978-1-4503-1542-5/08/2012 ...\$15.00.

## 1. INTRODUCTION

The markets of Location-Based Services (LBSs) [7] in urban areas, including navigational services, traffic management and location-based recommendation, have grown rapidly in recent years. Due to the needs of effectively improving smart urban living, it is beneficial for these LBSs to be able to recommend users Points-Of-Interest (POIs) where they may be interested in. Thus, effective and efficient urban POI recommendation techniques for LBSs targeting urban mobile users are desirable. The intuitive idea for POI recommendation is based on the personal check-in behaviors of a user. Although such strategy may reflect the personal preference of users, the recommended results are always the POIs that the user has been to. Hence, nowadays, new breed of item recommendation methods, called *social-based prediction*, have emerged. Such recommendation methods usually use the *social properties* of users, mined from the collections of users' social network, to recommend the probably interesting items (POIs) for a user. Figure 1 shows an example of social network information, which typically consists of social links of end users and some user-generated data (i.e., textual information, check-ins, etc.). Among the social-based recommendation studies, Collaborative Filtering (CF) techniques [12] have been widely used for influence of interestingness of item for mobile user. However, they tend to recommend popular items which most of similar users (or friends) are interested in but might not match individual user's preference actually. It may lead to the sparse data problem [15]. Additionally, these social-based recommendation methods only provide the POIs that the user has visited. This may also lead to the lost of applicability for recommendations.

Although the issues of social-based recommendation systems based on the users' check-ins have been discussed in the literatures, existing studies mostly consider only on the social properties of among users [12]. Notice that social links and check-ins typically reflect users' living-spheres, e.g., the environment

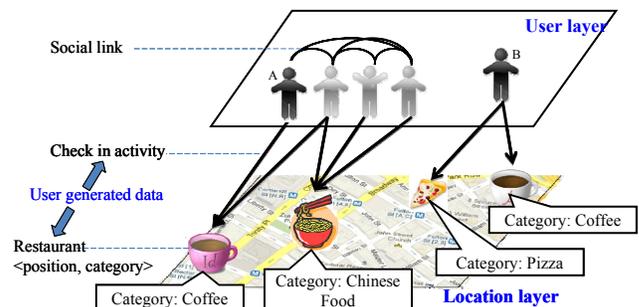


Figure 1. Location Based Social Network.

around users. As a result, CF-based recommendations are constrained by the users’ living-spheres if only social properties are considered. Take Figure 1 as an example, both user A and user B favor to drink coffee. Suppose that user A and user B are not friends, and user A’s friends always check in some Chinese food restaurants. Thus, some social-based Collaborative Filtering recommendations would recommend the Chinese food restaurant for user A. Additionally, such social-based Collaborative Filtering recommendation only consider the POIs that the user or her/his friends have been visited. It does not work well when previously unvisited POIs are considered. We argue that it is insufficient if merely using social links and check-in history to recommend POIs.

The notion of urban computing has been proposed by Zheng *et al.* [18] [19]. To achieve city-wide computing for serving people and their cities, Zheng *et al.* [18] [19] believe that every device, person, vehicle, building, and street in the urban area can be a sensor to understand city dynamics. Many POIs in Location-Based Social Network (LBSN) have been labeled with useful tags[16][17] such as coffee shop or pizza restaurant, which are crucial for assisting users in searching and exploring new places as well as for sensing POIs’ properties. Although we also could map users’ current position to some POI databases or road networks for understanding the tag of POI that the users stay in, this process is not work well in an urban area. The reason is that the POIs in an urban area are very crowded so that we cannot accurately tag the location just by catching users’ positions. Figure 2 shows that there are two totally different types of restaurants in the same building. Fortunately, we could view each user of LBSN as a sensor to detect the semantic tags of POI for improving POI recommendation in an urban area. In Figure 1, restaurants are tagged with several semantic tags such as Coffee, Chinese Food, Pizza, etc. We observe that both user A and B have visited coffee shops, implying that their preferences in restaurant are very similar. Suppose user B is a loves to have pizza as shown in Figure 1. Thus, it is more reasonable recommending the pizza parlor than recommending the Chinese food restaurant for user A. According to above reasons, we exploit users’ preferences from visited POIs to recommend users the POIs of LBSN in an urban area.

To address the above-mentioned problem, we propose a novel approach named *Urban POI-Mine (UPOI-Mine)* for recommending users the POIs in urban LBSN based on not only social factors but also users’ preferences. As shown in (1), given a set of users  $U$  and a set of POIs  $P$ , the problem of POI recommendation can be formulated as predicting relevance score of a given POI for each user, relevance between user’s preference and interestingness of POI,

$$f(u, p) \rightarrow v, \text{ where } u \in U, r \in P, \text{ and } v \in [0, 1] \quad (1)$$

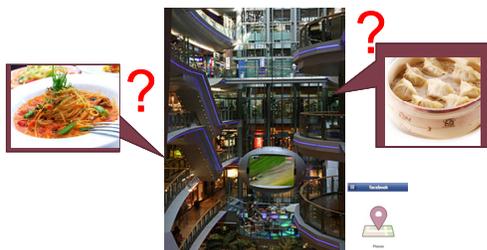


Figure 2. A scenario of POI in an urban area.

Hence, POI recommendation in LBSN can be addressed as a numerical value prediction problem. While numerical value prediction techniques have been developed for many applications, such as demographic prediction [4], bio life cycle analysis [10] and prediction of geographical natural [3], the problem has not been explored previously under the context of urban computing. Furthermore, the context-aware and environmental information changes quickly especially in urban areas. How to extract appropriate features to support the recommendation from such heterogeneous data is also a critical and challenge issue. To support POI recommendation based on users’ preferences and social properties, we address this problem by learning a regression-tree based predictor to realize the prediction of relevance score. A fundamental issue is to identify and extract a number of descriptive features for each place in the system. Selecting the right features is important because those features have a direct impact on the effectiveness of the prediction task. As mentioned earlier, only considering the POIs that the user or her/his friends have visited do not work well. Therefore, we explore the users’ preference in visited POIs and seek unique features of places captured in the users’ preference.

By dealing with the observations prompted in the above examples, we extract features of user-POI pair in three different but complementary aspects: 1) Social Factor (SF), 2) Individual Preference (IP), and 3) POI Popularity (PP). Features extracted from Social Factor, corresponding to a given POI for a user, can be derived from all check-ins among the user’s similar friends at the POI based on statistical analysis. In this paper, we extract check-in-based features (e.g., number of check-ins of friends who have a lot of common check-ins with the user) and spatio-based features (e.g., number of check-ins of friends whose living area is very close to the user) as relevance descriptions of specific POIs.

To involve the factor of user preference, we extract features from Individual Preference to capture the relatedness between users and POIs by exploiting the regularity of user check-in activities to POIs with the same tag. Since there are two kinds of tags are annotated on a POI, we could make good use of Individual Preference by deriving descriptive features of a given place from its “related” places. To facilitate extraction of features from Individual Preference, we build a preference table for each user that captures the relatedness between semantic tags and users by exploring regularities of user check-ins to the POI with the same semantic tag. We propose a family of preference formulations that capture the user-highlight and POI-category from the user check-in activities.

As mentioned earlier, such check-in data is very sparse. It leads to that the features extracted from both Social Factor and Individual Preference do not work well for a new user. We employ the popularity of POI to make a maximum likelihood estimation of the relative between user and POI. We argue that different types of POIs will be visited in different frequencies. For example, people may go to coffee shops every day but rarely visit French restaurants. Therefore, we normalize the check-ins of POI based on its semantic tag for representing its popularity. This popularity is thus treated as a feature of POI Popularity, along with features derived from Social Factor and Individual Preference, to feed the regression-tree model in the proposed UPOI-Mine approach.

This research work has made a number of significant contributions, as summarized below:

- We formulated the problem of POI recommendation in an

urban area as the problem of relevance score prediction. This problem has not been explored previously in the research community.

- We proposed *Urban POI-Mine (UPOI-Mine)*, a new approach for urban POI recommendation by mining urban users' check-in behaviors. and propose UPOI-Mine to learn a regression-tree for estimating relevance score of each user-POI pair. In the proposed *UPOI-Mine*, we explore simultaneously the factors namely i) Social Factor, ii) Individual Preference, and iii) POI Popularity by exploiting the LBSN data to extract descriptive features.
- We used a real dataset, which was crawled from Gowalla (<http://www.Gowalla.com/>), to evaluate the performance of our proposed UPOI-Mine in a series of experiments. The results show UPOI-Mine delivers superior performance over other recommendation techniques in terms of the popular measures NDCG and MAE.

The rest of this paper is organized as follows. We briefly review the related work in Section 2 and provide our urban POI recommendation approach UPOI-Mine in Section 3. Finally, we present the evaluation result of our empirical performance study in Section 4 and discuss our conclusions and future work in Section 5.

## 2. RELATED WORK

In this chapter, we review and classify relevant previous studies into three categories: 1) Similarity Measurement, 2) Recommendation Systems, and 3) Mobility Prediction.

**Similarity Measurement.** For solving the problem of data sparse, many researches using Collaborative Filtering (CF)-based approach or item-based approach to estimate missing values. The fundamental problem of CF-based approach is how to evaluate user similarity and location similarity. In [12], Ellen Spertus *et al.* proposed six different measures for recommending online social networks. The six measures applied cosine distance, mutual information measure, TF-IDF and log-odds functions to measure the similarity of community.

**Recommendation System.** In recent years, rapid development of the Internet brings much information and various businesses. A variety of web sites provide huge data of music, images and commodity. How to recommend appropriate items to users is a critical problem. There are also a number of researches for recommendation systems. Traditional recommendation systems usually use CF-based method as the main concept. In [6], Tzvetan Horozov *et al.* proposed an enhanced CF solution for personalized POI recommendation. Some trust-based approaches such as [9], Paolo Massa and Paolo Avesani proposed a trust-aware recommender system. The system builds a trust metric and makes use of trust information to recommend items.

In [5], Mohsen Jamali and Martin Ester proposed TrustWalker. TrustWalker combines Random walk model and trust-based CF approach to predict the ratings of items for users. Trust-based CF approach uses trust values and user-to-item ratings to predict the item ratings. A trust value is the user-to-user rating. If you think someone's interest is similar with you, you can give this people a high trust value. TrustWalker can solve the "cold start problem" with the same precision. The content-based recommendation systems like [11], Chihiro Ono *et al.* used Bayesian network modeling user preference for recommending movies. Bayesian Networks is highly flexibility so it is appropriate for representing complex relations between users' preference and contexts. In [2],

Souvik Debnath *et al.* proposed a way to hybrid CF and content-based recommendation system. The approach can determine the weight values of attributes by the linear regression, which obtained from a social network.

**Mobility Prediction.** A user's location is completely related to his social relations and personal information. LBSN has become a popular application. Many users join a LBSN and share their life, photo, music and location history with their friends. They also get interesting information from their friends. More and more researches on location recommendation with social networks have been proposed with the development of LBSN.

In [1], Betim Berjani and Thorsten Strufe proposed Regularized Matrix Factorization (RMF) recommender to recommend appropriate spots, which mean locations, for users. This research used LBSN data that is crawled from a LBSN website, Gowalla. RMF recommender is a personalized recommender for places. RMF recommender first maps users and spots to a joint latent factor space. RMF recommender exploits regularized Singular Value Decomposition (SVD) model to predict the ratings of users to spots.

In [13], Kenneth Wai-Ting Leung *et al.* proposed a Collaborative Location Recommendation (CLR) framework based on co-clustering. This approach considers location, user and activity to build Community Location Model (CLM) graph. Then CLR approach uses Community-based Agglomerative-Divisive Clustering (CADC) algorithms to iteratively merge and divide nodes in CLM graph. After clustering users, locations and activities by CADC algorithm, it gets refined clusters of similar locations which are visited by similar users and have similar activities. For clustering CLM graphs, it models the similarity as follows:

- Two users are similar if they have similar activity patterns and have visited similar locations.
- Two locations are similar if they are visited by similar users and take place in similar activities.
- Two activities are similar if they are done by similar users and have similar location sequence.

However, the refined clusters are still too large for recommendation. For this reason, this approach classified users into three types: Pattern Users, Normal Users and Travelers by their entropies of visited locations. Then, this approach can recommend locations according different type users. For example, for a traveler, the recommender first finds similar travelers to this user. Then, the recommender finds the locations, which are visited by these similar travelers and have similar activities. Finally, the recommender recommends these locations to the user.

In [14], Ye *et al.* proposed a CF-based POI recommendation framework. This framework fuses user preference influence, social influence and geographic influence to infer the check-in probability for a given user to visit a POI. This approach also exploits a power-law distribution to build the geographical influence among POIs and uses CF method to depict user preference influence and social influence.

## 3. URBAN POI-MINE (UPOI-MINE)

In the proposed UPOI-Mine approach, we design a two-phase algorithm, as shown in Figure 3, to address the problem of user check-in behaviors mining for urban POI recommendation. The first phase deals with the feature extraction (see lines 1 to 5 in Figure 3), while the second phase takes care of the restaurant

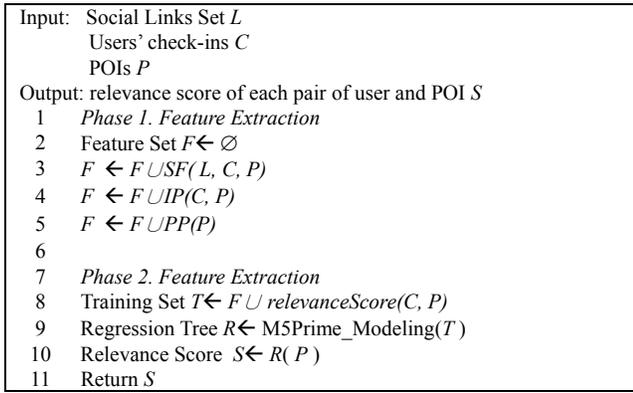


Figure 3. UPOI-Mine algorithm.

recommendation (see lines 7 to 11 in Figure 3). The task of feature extraction explores three aspects that are discussed in Introduction. For a pair of specific user and POI, we explore the Social Factor (SF) as population features which abstract the aggregated check-ins of the user's friends. On the other hand, we explore the Individual Preference (IP) between users and POIs in order to formulate descriptive features of a given user-POI pair. Moreover, to overcome the data sparse problem, POI Popularity (PP) is considered as a feature in our recommendation model. The features derived from Social Factor, Individual Preference and POI Popularity are used to learn a regression-tree model for predicting relevance score of each user-POI pair in the POI recommendation phase. Given a user-POI pair, the prediction by the regression-tree model estimates the relevance between the interestingness of POI and the user's preference. After checking all POIs, we obtain all qualified POIs for the user under examination.

### 3.1 Features from Social Factor

Our goal is to extract discriminative Social Factor features from check-ins among the user's similar friends at the POI. Intuitively, aggregating friends' relative check-ins of POI could be used for influencing the probability of that a user likes the POI. Formally, given a friend  $f$  and a set of POI  $P$ , the  $f$ 's relative check-ins of a POI  $p$  is formulated as (2).

$$\text{relative check-ins}(f, p) = \frac{\text{checkins}(f, p)}{\sum_{p' \in P} \text{checkins}(f, p')} \quad (2)$$

Accordingly, given a user-POI pair  $(u, p)$ , the features extracted from Social Factor could be generally formulated as (3).

$$SF(u, p) = \sum_{f \in F(u)} (\text{relative check-ins}(f, p) \times \text{similarity}(f, u)) \quad (3)$$

where  $F(u)$  is the set of user  $u$ 's friends.

As mentioned above, we can significantly observe that measuring similarity of two users is the key of Social Factor features. Intuitively, living sphere of users and their friends could be utilized for measure the similarity between users and their friends

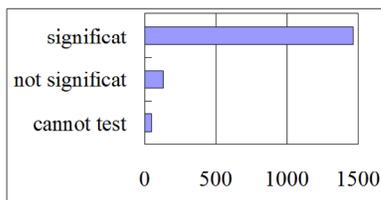


Figure 4. result of  $\chi^2$  test.



Figure 5. An example of base-point

due to the nature of living style and activities offered by their living areas. As a result, different similarities, naturally formed in aggregated behaviors of friends to various kinds of POI, are embedded in the friends' check-in activities. In a LBSN data, the most important information is user's common check-ins and distance among users for user similarity measurement. In the following, we propose to extract two population features to depict POI as below.

- Similarity by Common Check-ins (CheckSim) - We employ the  $\chi^2$  test for testing relation of check-in behaviors of Gowalla users and their friends. If the test shows that a relationship is significant, it means the user always checks in at a POI where her friends are also checked in. Based on the observations from the Gowalla dataset, shown in Figure 4, we find most of users will check in POI, which her friends also check in. Hence, the common check-ins is a good for measuring users' similarity. Based on the observations, we formulate the similarity of two users as (4).

$$\text{CheckSim}_{i,j} = \begin{cases} \cos(v_i, v_j) & , \text{if user } i \text{ and } j \text{ are friends} \\ 0 & , \text{otherwise} \end{cases} \quad (4)$$

where  $\cos()$  indicates cosine similarity and  $v_i$  indicates the check-in vector of user  $i$ . Take Table 1 as an example, the check-in vector of user  $i$  is  $\langle 1, 0, 2, 5, 0 \rangle$ , and the check-in vector of user  $j$  is  $\langle 0, 10, 0, 1, 0 \rangle$ . Thus, the CheckSim of user  $i$  and user  $j$  is

Table 1. An example of check-in log.

Restaurant ID	r1	r2	r3	r4	r5
user i	1	0	2	5	0
user j	0	10	0	1	0
user k	1	1	0	0	0
user l	1	1	1	1	5
total	3	12	3	7	5

$$\frac{(1 \times 0) + (0 \times 10) + (2 \times 0) + (5 \times 1) + (0 \times 0)}{\sqrt{1^2 + 0^2 + 2^2 + 5^2 + 0^2} \times \sqrt{0^2 + 10^2 + 0^2 + 1^2 + 0^2}} \approx 0.09$$

- Similarity by Relative Distance (DisSim) - As discussed above, most people will check in the place following their friends who live nearby them. Based on the idea, this design idea of DisSim focuses on the distance between users and their friends. To do so, we must first identify users' living areas. We argue that most of a user's check-in activities will happen in her/his living area. Thus, for each user, we find out her/his top  $k$  frequently visiting POIs and treat the central of these POIs as her/his *base-point*, as shown in Figure 5. Accordingly, we formulate the DisSim of two

users as follow.

$$DisSim(u, f) = \begin{cases} 1 - \frac{Distance(u, f)}{\max_{f' \in F(u)} \{Distance(u, f')\}}, & \text{if user } u \text{ and } f \text{ are friends} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where Distance() indicates the Euclidean distance of two base-points and  $F(u)$  indicates the set of user  $u$ 's friends.

### 3.2 Features from Individual Preference

As mentioned above, we could view each user of LBSN as a sensor to detect the semantic tags of a POI for improving POI recommendation in an urban area. In Gowalla website, there are two kinds of semantic tags, i.e., category and highlight, as shown in Figure 6. The category tag is annotated on a place when the place is created. Each place just have only one category tag, e.g., coffee, pizza, etc. On the other hand, any user (even the user never checks in the place) can arbitrarily annotate the highlight tag on a place.

Since the Gowalla will record the count of highlight tag, we could generally determine the possibility of that a tag  $t$  is annotated on a POI  $p$ , as shown in (6).

$$Possibility(t, p) = \frac{count(t, p)}{\sum_{t' \in T(p)} count(t', p)} \quad (6)$$

where  $count(t, p)$  indicates the number of times the tag  $t$  is annotated on the POI  $p$ , and  $T(p)$  indicates the set of tags of POI  $p$ . Take Figure 6 as an example, the possibility of that a tag 'coffee' is annotated on a POI is

$$\frac{2}{2 + 10 + 88} = 0.02$$

Accordingly, given a user-POI pair  $(u, p)$ , the features extracted from Individual Preference could be generally formulated as (7).

$$IP(u, p) = \sum_{t \in T(p)} (Possibility(t, p) \times Personal Preference(u, t)) \quad (7)$$

where  $T(p)$  indicates the set of tags of POI  $p$ .

We can significantly observe that measuring a user's individual preference of a semantic tag is the key of Individual Preference features. Intuitively, users' check-in history could reflect their preference of the type of POI. As a result, for each user, we aggregate the number of check-ins of the POI with the same tag to represent each user's personal preference of semantic tag. In the following, we propose to extract two features to depict users' preference.

- Preference in Category (CPref) - Based our observations from the Gowalla dataset, users' check-in activities are fluctuated. Some users frequently check in many places,



Figure 6. An example of semantic tag of a POI.

and some users rarely check in at all. Hence, to measure individual user's personal preference of POIs, we need to normalize the aggregated number of check-ins by user's total check-ins. Based on above, we formulate a user  $u$ 's personal preference of a category tag  $t$  as (8).

$$CPref(u, t) = \frac{\sum_{r \in C(t)} checkins(u, r)}{\#total\ check\ -\ ins\ of\ u} \quad (8)$$

where  $C(t)$  indicates the set of POIs with category tag  $t$ . Take Table 1 and Table 2 as an example, the check-in vector of user  $i$  is  $\langle 1, 0, 2, 5, 0 \rangle$ . Thus, the user  $i$ 's personal preference of a category tag  $A$  is

$$\frac{1 + 0 + 0}{1 + 0 + 2 + 5 + 0} = 0.125$$

Table 2. An example of category tag

Restaurant ID	p1	p2	p3	p4	p5
Category	A	A	B	C	A

Table 3. An example of highlight tag

Restaurant ID	p1	p2	p3	p4	p5
Highlight	a, b	b, c, d	a, d	a, c	g

Note that, as mentioned above, people will annotate highlight tag on a place and repeat the annotation many times. Based on the observation, to measure an individual user's personal preference of POIs, we could not directly normalize the aggregated number of check-ins by user's total check-ins. Take Table 1 and Table 3 as an example, suppose that we use user's total check-ins to normalize the aggregated number of check-ins, the user  $i$ 's personal preference of a highlight tag  $a$  is

$$\frac{1 + 2 + 5}{1 + 0 + 2 + 5 + 0} = 1$$

It is clear that if we use user's total check-ins to normalize the aggregated number of check-ins, the total of personal preference of a user will be greater than 1.0 and the scale of total of personal preference of a user will be very fluctuated.

- Preference in Highlight (HPref) - Based on the above observation and condition of highlight tag, we use the summation of a user's total check-ins of each highlight tag for normalization of the user's personal preference of POIs. To do so, given a set of highlight tags  $H$ , we formulate a user  $u$ 's personal preference of a highlight tag  $t$  as (9).

$$HPref(u, t) = \frac{\sum_{r \in hl(t)} Checkins(u, r)}{\sum_{\forall r' \in H} \sum_{r \in hl(r')} Checkins(u, r)} \quad (9)$$

where  $hl(t)$  indicates the set of POIs with highlight tag  $t$ . Take Table 1 and Table 3 as an example, the check-in vector of user  $i$  is  $\langle 1, 0, 2, 5, 0 \rangle$ . Thus, the user  $i$ 's personal preference of the highlight tag  $a$  is

$$\frac{1 + 2 + 5}{(1 + 2 + 5) + (1 + 0) + (0 + 5) + (0 + 2) + (0)} = 0.5$$

### 3.3 Features from POI Popularity

As discussed in Introduction, check-in data is very sparse. It leads to the features extracted from both Social Factor and Individual Preference does not work well for new users. We employ the popularity of POI to make a maximum likelihood estimation of the popularity of POI. However, the type of POI always affects

users' check-in will. For example, people may buy a cup of coffee in a coffee shop everyday but rarely go French restaurant. Base on this idea, we estimate the likelihood by conditional probability which is the probability of that users check in the POI given a category, so called relative popularity of POI. In the following, we propose to extract the feature to depict popularity of POI.

$$RP(p) = P(p|C(p)) = \frac{\#total\ check\ -\ ins\ of\ p}{\sum_{p' \in C(t)} (\#total\ check\ -\ ins\ of\ p')} \quad (9)$$

where  $C(t)$  indicates the set of POIs with category tag  $t$ . Take Table 1 and Table 2 as an example, the set of POIs with category tag  $A$  are  $p1$ ,  $p2$ , and  $p5$ . The total check-in of POI  $p1$ ,  $p2$ , and  $p5$  are 3, 12, and 5, respectively. Thus, the popularity of POI  $p1$  is

$$\frac{3}{3+12+5} = 0.15$$

### 3.4 POI Recommendation

After the phase of feature extraction, features derived from Social Factor, Individual Preference and POI Popularity are used as inputs for the POI recommendation phase to learn a regression-tree model. We choose M5Prime as the relevance score predictor because it has shown excellent performance in similar tasks [3]. The M5Prime is one kind of regression-trees, but it is not constrained by using M5Prime as the regression-tree model. When the model is trained, the training data first will be divided according to the decreasing of variance of target attribute. Then, each piece of training data will be used for building individual regression model. Meanwhile, the procedure of dividing training data will be recorded to build a decision tree as shown in Figure 7. This modeling process is NOT sensitive for bias data. In other words, M5Prime is not easily affected by data sparse problem.

In addition, M5Prime is an automatic and non-parametric model. Therefore, it is more convenient to use. Besides, as mention earlier, the LBSN data may growth rapidly, thus the efficiency is an unavoidable issue we should face. Fortunately, M5Prime needs lower run time. In our proposed UPOI-Mine, all user-POI pairs are used for building training model by M5Prime, i.e., a POI with high relevance score under examination is considered as the candidate with high probability that the user may like. A POI tends to be automatically recommended to a user if the POI is predicted as a high relevant score for the user.

## 4. EXPERIMENTS

In this section, we conduct a series of experiments to evaluate the performance for the proposed UPOI-Mine using Gowalla dataset. All the experiments are implemented in Java JDK 1.6 on an Intel Xeon CPU W3520 2.67GHz machine with 24GB of memory running Microsoft Windows win7. We first describe the data preparation on the Gowalla dataset and then introduce the evaluation methodology. Finally, we show our experimental results for following discussions.

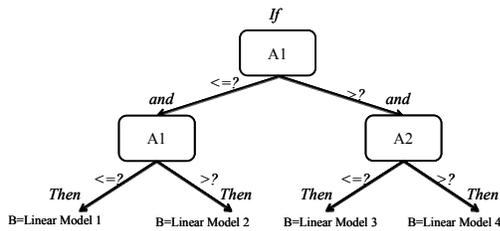


Figure 7. An example of M5Prime.

## 4.1 Gowalla Dataset

The Gowalla dataset is a check-in dataset collected from Gowalla website. The dataset contains 1,964,919 spots, 18,159 users, 5,341,191 check-ins and 392,246 friend links. We normalize the check-in for each user. Then, we use the normalized check-in as the ground truth, which is called "relevance score". We normalize the check-in values into the range [0, 5]. The process of normalization is described as (11).

$$Relevance\ Score = \begin{cases} 3, & \text{if } x = avg \\ 3 + \frac{x - avg}{max - avg} \times 2, & \text{if } x > avg \\ 3 - \frac{x - avg}{min - avg} \times 2, & \text{if } x < avg \end{cases} \quad (11)$$

where  $x$  indicates the real check-in number of a user. "min", "max" and "avg" indicate the smallest, largest and average check-in number of a user, respectively. Finally, we use the relevance score in our proposed UPOI-Mine to evaluate the score of POI based on social, location information and user preference factors.

## 4.2 Evaluation Methodology

UPOI-Mine is based on the ranking of POI score and thus can be viewed as an information retrieval system if we consider a user as a query term. Therefore, we employ the popular measurement Normalized Discounted Cumulative Gain (NDCG) [8] to measure the list of recommended POIs. NDCG is commonly used in information retrieval to measure the search engine's performance. A higher NDCG value for a list of search results indicates that the highly relevant items have appeared earlier (with higher ranks) in the result list. For each list of recommended POIs, we can obtain a score list, where the scores are provided by ground truth. Such list is called the relevance vector. For example, suppose that the prediction model estimates the relevance scores of POIs  $F1$ ,  $F2$ ,  $F3$ , and  $F4$  are 4, 3, 2, and 1, respectively. Hence, the POIs will be ordered as  $\langle F1, F2, F3, F4 \rangle$  by recommender. Suppose that the relevance score of ground truth for POI list  $\langle F1, F2, F3, F4 \rangle$  is  $G = \langle 2, 3, 0, 1 \rangle$ . That is the relevance scores of  $F1$  and  $F2$  are 2 and 3, respectively. The Discounted Cumulative Gain (DCG) of a relevance vector  $G$  is computed by Equation (12). The premise of DCG is that the highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result. Here the parameter  $b$  is to control where we start to reduce the relevance value. For example, if the relevance vector is  $\langle 2, 3, 0, 1 \rangle$  and  $b$  is set as 3, the  $DCG[4]$  is  $2+3+(0/\log_3 3)+(1/\log_3 4)$ . (In our experiments,  $b = 2$ .)

$$DCG[i] = \begin{cases} G[i], & \text{if } i = 1 \\ DCG[i-1] + G[i], & \text{if } i < b \\ DCG[i-1] + \frac{G[i]}{\log_b i}, & \text{if } i \geq b \end{cases} \quad (12)$$

In particular,  $NDCG@p$ , measures the relevance of top  $p$  as shown in Equation (13).

$$NDCG@p = \frac{DCG[p]}{IDCG[p]} \quad (13)$$

where  $IDCG[p]$  indicates the  $DCG[p]$  value of ideal ranking list. For example, given a ranking list of 5 items with relevance as  $\langle 4, 1, 3, 1, 1 \rangle$ , the ideal ranking list of this 5 items is  $\langle 4, 3, 1, 1, 1 \rangle$ . NDCG ranges from 0 to 1. The higher NDCG is, the better a ranking result list is. In the above example, the  $NDCG@5$  is

$$NDCG@5 = \frac{4 + \frac{1}{\log_2 2} + \frac{3}{\log_2 3} + \frac{1}{\log_2 4} + \frac{1}{\log_2 5}}{4 + \frac{3}{\log_2 2} + \frac{1}{\log_2 3} + \frac{1}{\log_2 4} + \frac{1}{\log_2 5}} = 0.913785$$

However, NDCG is not comprehensive, because it only focuses on ranking performance avoiding the absolute difference between estimated relevance and ground truth of relevance. For example, given a ranking list of 5 items with estimated relevance as  $\langle 5, 4, 3, 1, 1 \rangle$ , the ground truth of relevance of this 5 items is  $\langle 4, 3, 2, 2, 2 \rangle$ . We can observe that NDCG will be 1.0 that means the effectiveness of recommender is pretty good. Excepting the ranking performance, we also want to examine the how close our predictions are to the eventual outcomes. Therefore, we employ the Mean Absolute Error (MAE) to measure the list of recommended POIs as Equation (14).

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (14)$$

where  $f_i$  indicates the estimated relevance scores of POI  $i$  and  $y_i$  indicates the ground truth of relevance scores. The lower MAE is, the fewer error is. In the above example, the MAE is

$$\frac{1}{5} \times (|5 - 4| + |4 - 3| + |3 - 2| + |1 - 2| + |1 - 2|) = 1.0$$

### 4.3 Experimental Results and Discussions

We divide the experiment into two parts: internal experiment and external experiments. For the internal experiments, we first compare the performance of our social, location information and user preference factors. Then, we compare the effectiveness of every feature in all of factors. For the external experiments, we compare the performance of UPOI-Mine with TrustWalker [5] and CF-based model [14] in terms of NDCG and MAE.

#### 4.3.1 Comparison of Various Features

This experiment evaluates how each factor and feature performs in the proposed UPOI-Mine in terms of NDCG@10 and MAE. Figure 8 shows the NDCG@10 value of UPOI-Mine, considering social, location information and user preference, respectively. We observe that the effect of user preference is better than that of other two factors. The result shows that the check-in behaviors of

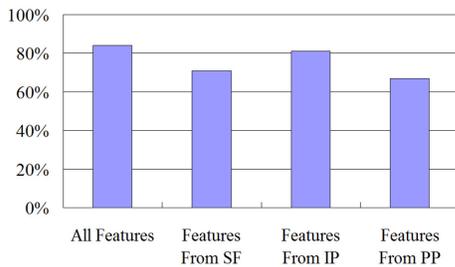


Figure 8. Comparison of deferent aspects in terms of NDCG@10.

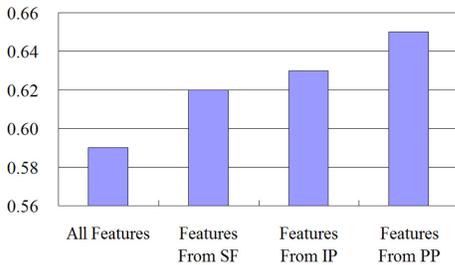


Figure 9. Comparison of deferent aspects in terms of MAE.

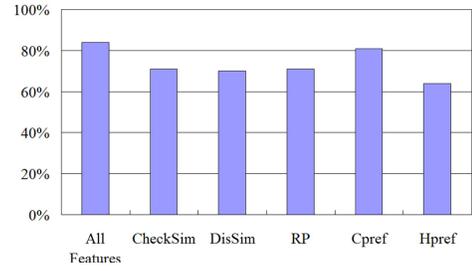


Figure 10. Comparison of deferent features in terms of NDCG@10.

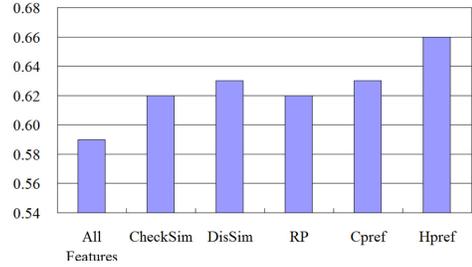


Figure 11. Comparison of deferent features in terms of MAE.

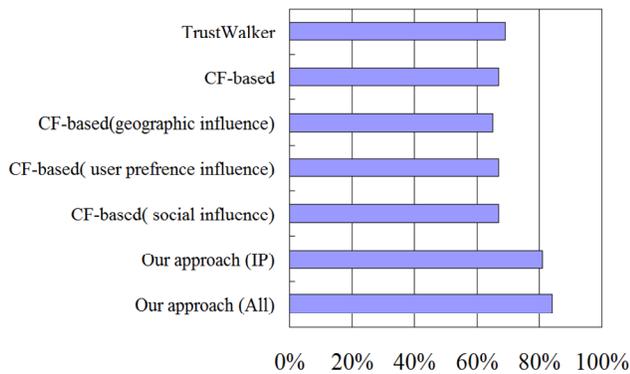
user-self still more important than that of friends. However, we still can use location information and social factor to provide recommendations when the user is new to this LBSN website. Figure 9 shows that the MAE is smallest when combining all factors. The reason is that the three factors can be complementary of one another when regression model is trained. In Figure 10 and Figure 11, we compare all features in terms of NDCG@10 and MAE respectively. We observed that the effect of category feature outperforms other features because the check-in of user-self is most important.

#### 4.3.2 Comparison of existing recommender

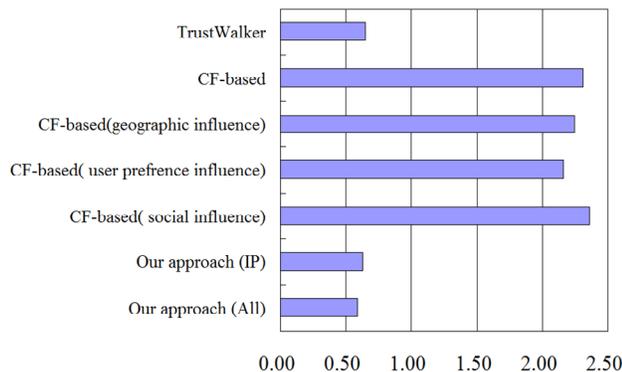
This experiment evaluates the performance of our proposed UPOI-Mine comparing TrustWalker [5] and CF-based [14] in terms of NDCG@10 and MAE. TrustWalker is a trust-based CF method. It uses social factor to recommend items to users. CF-based POI [14] considers geographic influence, social influence and user preference influence with collaborative filtering to recommend locations to users. Figure 12 shows UPOI-Mine outperforms TrustWalker and CF-based POI in terms of NDCG@10 and MAE. The reason is that we consider check-in of user-self in the factor of user preference while other methods do not. We also observe that the MAE of CF-based POI is large, as shown in Figure 13. CF-based POI only considers the common check-in of two users when computing social influence and user preference influence. Because of the sparsity of LBSN data, considering common check-in will lead to many zero scores. So the reason of large MAE value is that many zero scores which is far away for the relevance scores of users while UPOI-Mine and TrustWalker use social relation to generate scores in social factor.

## 5. CONCLUSIONS

In this paper, we have proposed a novel approach named *Urban POI-Mine (UPOI-Mine)* for recommendation of interesting urban POIs by mining users' preferences. Meanwhile, we tackle the problem of mining user check-in behaviors in urban computing, which is a crucial prerequisite for effective recommendation of POIs in urban areas. The core task of POI recommendation in



**Figure 12. Comparison of various models in terms of NDCG@10.**



**Figure 13. Comparison of various models in terms of MAE.**

urban areas is nicely transformed to the problem of relevance score prediction. We evaluate the relevance score of each user-POI pair by learning a regression-tree. In the proposed *UPOI-Mine*, we have explored i) *Social Factor (SF)*, ii) *Individual Preference (IP)*, and iii) *POI Popularity (PP)* by exploiting the LBSN data to extract descriptive features. To our best knowledge, this is the first work on urban POI recommendation that considers social factor, preference relatedness and POI popularity in LBSN data, simultaneously. Through a series of experiments by the real dataset Gowalla, we have validated our proposed *UPOI-Mine* and shown that *UPOI-Mine* has excellent performance under various conditions.

## 6. ACKNOWLEDGMENTS

This research was partially supported by National Science Council, Taiwan, under grant no. NSC100-2631-H-006-001 and NSC100-2218-E-006-017.

## REFERENCES

- [1] B. Berjani and T. Strufe. A Recommendation System for spots in Location-Based Online Social Network. Proceedings of the 4th Workshop on Social Network Systems Article No. 4,2011.
- [2] S. Debnath, N. Ganguly, P. Mitra. Feature Weighting in Content Based Recommendation System Using Social Network Analysis. Proceedings of WWW, pages 1041-1042, 2008.
- [3] A. Etemad-Shahidi and J.Mahjoobi. Comparison between M5' model tree and neural networks for prediction of

significant wave height in Lake Superior. Ocean Engineering36(2009)1175–1181.

- [4] J. Hu, H.-J. Zeng, H. Li, C. Niu, Z. Chen. Demographic prediction based on user's browsing behavior. Proceedings of ACM WWW (WWW'2007) May 2007.
- [5] M. Jamali, M. Ester . TrustWalker: A Random Walk Model for Combining Trust-based and Item-based Recommendation. Proceedings of KDD, pages 397-406, 2009.
- [6] T. Horozov, N. Narasimhan, V. Vasudevan. Using location for personalized POI recommendations in mobile environments. Proceedings of SAINT, pages 124-129, 2006.
- [7] E. H.-C. Lu, W.-C. Lee and V. S. Tseng, "A Framework for Personal Mobile Commerce Pattern Mining and Prediction," IEEE Transactions on Knowledge and Data Engineering (TKDE), Volume 24, Issue 5, May 2012, Pages 769-782.
- [8] D. Manning, P. Raghavan and H. Schütze. Introduction to Information Retrieval. Cambridge University Press, 2008.
- [9] P. Massa and P. Avesani ,Trust-aware Recommender Systems, In Proceedings of RecSys, pages 17-24, 2007.
- [10] T. Menzies, J. S. D. Stefano, M. Chapman. Learning Early Lifecycle IV&V Quality Indicators. Proceedings of the Ninth International Software Metrics Symposium (METRICS'03)
- [11] C. Ono, M. Kurokawa, Y. Motomura, and H. Asoh. A Context-Aware Movie Preference Model Using a Bayesian Network for Recommendation and Promotion, In Proceedings of UM, pages 247-257, 2007.
- [12] E. Spertus , M. Sahami, O. Buyukkokten. Evaluating similarity measures: a large-scale study in the Orkut social network. In Proceedings of KDD, pages 678-684, ,2005.
- [13] K. W.-T. Leung,D. L. Lee,W.-C. Lee. CLR: A Collaborative Location Recommendation Framework based on Co-Clustering. In Proceedings of SIGIR , pages 305-314 , 2011.
- [14] M. Ye, P. Yin, W.-C. Lee, and Dik-Lun Lee. Exploiting Geographical Influence for Collaborative Point-of-Interest Recommendation. Proceedings of KDD, pages 1046-1054 , 2011.
- [15] S.-J. Yen, Y.-S. Lee, C.-H. Lin and J.-C. Ying, Investigating the Effect of Sampling Methods for Imbalanced Data Distributions, Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC'2006), pp. 4163-1468, October 2006.
- [16] J. J.-C. Ying, E. H.-C. Lu, W.-C. Lee, T.-C. Weng, V. S. Tseng. Mining User Similarity from Semantic Trajectories. In Proceedings of ACM SIGSPATIAL International Workshop on Location Based Social Networks (LBSN' 10), San Jose, California, USA, November 2, 2010.
- [17] J. J.-C. Ying, W.-C. Lee, T.-C. Weng, V. S. Tseng. Semantic Trajectory Mining for Location Prediction. Proceedings of The 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS' 11), Chicago, IL, Nov. 2011.
- [18] J. Yuan, Y. Zheng, and X. Xie, Urban Computing with Taxicabs, Proceedings of 13th ACM International Conference on Ubiquitous Computing (UbiComp 2011), Beijing, China, Sep. 2011.
- [19] J. Yuan, Y. Zheng, X. Xie. Discovering regions of different functions in a city using human mobility and POIs. Proceedings of 18th SIGKDD conference on Knowledge Discovery and Data Mining (KDD 2012).