On Information Coverage for Location Category Based Point-of-Interest Recommendation

Xuefeng Chen[1] **Yifeng Zeng**[2] **Gao Cong**[3] **Shengchao Qin**[2] **Yanping Xiang**[1] **Yuanshun Dai**[1]

[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, China, {cxflovechina, xiangyanping, uestcdaiys}@gmail.com

[2]School of Computing, Teesside University, UK, {Y.Zeng, S.Qin}@tees.ac.uk

[3]School of Computer Engineering, Nanyang Technological University, Singapore, gaocong@ntu.edu.sg

Abstract

Point-of-interest (POI) recommendation becomes a valuable service in location-based social networks. Based on the norm that similar users are likely to have similar preference of POIs, the current recommendation techniques mainly focus on users' preference to provide accurate recommendation results. This tends to generate a list of homogeneous POIs that are clustered into a narrow band of location categories (like food, museum, etc.) in a city. However, users are more interested to taste a wide range of flavors that are exposed in a global set of location categories in the city. In this paper, we formulate a new POI recommendation problem, namely top-K location category based POI recommendation, by introducing information coverage to encode the location categories of POIs in a city. The problem is NP-hard. We develop a greedy algorithm and further optimization to solve this challenging problem. The experimental results on two real-world datasets demonstrate the utility of new POI recommendations and the superior performance of the proposed algorithms.

Introduction

The increasing popularity of location-based social networks (LBSNs), e.g., *Foursquare*, *GyPSii* and *Loopt*, encourages more and more users to share their experience for point-of-interest (POI) in a cyber world (Zheng 2011; Zheng et al. 2010a). When users visit a POI such as store, museum, $etc.$, they post their physical locations, comments and tips that compose a set of $check\text{-}in$ data in the registered LBSNs. The quick aggregation of data naturally generates valuable service of POI recommendation that instructs users on exploring new places.

Most of the existing work on POI recommendation discovers users' preference implicitly through relating similar users on previous check-in activities in LBSNs (Konstas, Stathopoulos, and Jose 2009; Ye et al. 2011b; Ye, Liu, and Lee 2012; Yuan et al. 2013), and offers a list of POIs to users. Due to the limited budget (like time, money, etc.), users may visit K POIs that are ranked in terms of their *relevance* to users' preference in the recommendation. The recommendation, called *conventional top-K POIs* in this paper, tends to generate a set of homogeneous POIs (e.g., all about restaurant), which are often similar to the majority of the POIs visited by the previous users. However, users are often interested to be recommended with a set of heterogeneous POIs that can cover the different types (e.g., food, sports, etc) of locations in the targeted city.

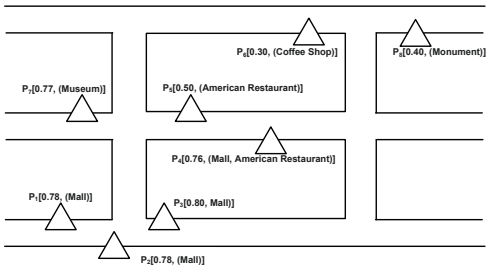


Figure 1: A place with eight POIs (Δ). Each POI is denoted by one tuple in which the number scores its relevance to a user and the term describes its location category.

Considering the example of eight potential POIs in Fig. 1, the top-3 POIs, $\{P_1, P_2, P_3\}$, are highly scored as they are the most relevant to the user's historical check-ins. However, the resulting recommendation is rather monotonic and concentrates only on *mall* in a new place. In contrast, the POIs, $\{P_3, P_4, P_7\}$, would be a better recommendation list as they achieve high relevance scores and simultaneously provide more flavor of the city, including mall, restaurant and museum, to users. For instance, tourists would like to enjoy shopping while browsing through museums or historical monuments around the city. The recommendation needs to consider a wide range of location categories that provide global features of the city. Note that LBSNs, such as *Foursquare*, categorize locations in a city and personalize the POI search. Hence improving information coverage on different categories is to exploit valuable information in LBSNs thereby providing a better POI recommendation.

As recognized in *Foursquare*, most of locations in a city can be categorized into different types such as mall, Austrian restaurant, coffee shop and so on [1], and these categories represent different characteristics of the city. Users would like to explore new types of locations that have not been visited by them. In Fig. 2, we confirm this observation in two real-

[1]https://developer.foursquare.com/categorytree

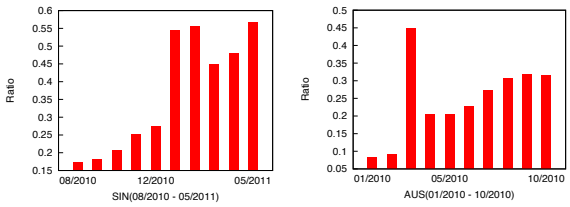


Figure 2: Ratio reports users' incline to exploring new location categories over time.

world datasets containing check-ins in Singapore (SIN) and Austin (AUS) respectively over one time period. We compute the ratio of new location categories (in contrast to those in the previous month) to all ones visited by an user for each month and report the average value for all users. Fig. 2 shows that the users explore around 25-30% (Y-axis: Ratio) new location categories for most of the months (X-axis: Month) in both datasets. The peaks appear due to a significantly larger number of check-ins made in the particular months. The existing POI recommendation methods tend to recommend a user the POIs that belong to the same set of categories as those visited by the user.

In this paper, we aim to improve the conventional top-K POIs on information coverage of location categories in the recommendation. We formulate the issue as one multi-objective optimization problem solving which recommends top-K location category based POIs (LC-POIs). The top-K LC-POIs optimize both their information coverage of a place and their relevance to users in the recommendation.

Following the same spirit of user-based collaborative filtering methods (Zheng et al. 2010b; Ye et al. 2011b), we compute relevance of POIs by gauging users' similarity based on their previous check-in activities. The technique has been well studied and adapted successfully in POI recommendation. Meanwhile, as optimizing information coverage expects a list of POIs to jointly enclose different location categories in a place, we resort to information coverage function for the computation purpose (El-Arini et al. 2009). Since all locations are categorized into several types in check-ins, we compute the degree to which one POI covers a category by counting how often the POI is checked as the category in the data. It represents the popularity of the POI labelled by the category in a city. We further weight the coverage degree with the category popularity.

Solving top-K LC-POIs problem is rather challenging as in principle we need to enumerate all possible sets of K POIs that can be retrieved from check-ins. However, we observe that the objective function, which models both information coverage and relevance, satisfies attractive property of submodularity (Nemhauser, Wolsey, and Fisher 1978) and monotony. By exploiting the property, we propose a greedy algorithm that is guaranteed to produce a near-optimal solution of top-K LC-POIs. To speed up the recommendation, we improve the efficiency of greedy algorithm by pruning the POIs with low relevance and information coverage. We also evaluate performance of the proposed algorithm on two real-world datasets.

Related Work

Most of POI recommendations follow the classical user-based collaborative filtering techniques that score POIs in terms of similarity between users' check-in activities (Zheng et al. 2010b; 2009; Ye, Yin, and Lee 2010). The techniques are further improved by taking into account social and geographical influence in the recommendation (Gao, Tang, and Liu 2012; Ye, Liu, and Lee 2012; Cheng et al. 2012; Liu et al. 2013).

Location content becomes an important input for improving LBSN service as it provides more semantic information to recommender systems (Noulas et al. 2011; Ye et al. 2011a; Bao, Zheng, and Mokbel 2012; Liu and Xiong 2013). To overcome the data sparsity problem, Yin *et al.* (Yin et al. 2013) utilized local features (e.g., attractions and events) to improve the model learning and inference procedure for the recommendation purpose. In parallel, Ye *et al.* (Ye, Zhu, and Cheng 2013) exploited region categories to predict the most likely location of users given their previous activities. In this paper, location categories are considered as important features in the recommended POIs, which become another dimension on evaluating recommendation performance.

One relevant topic is on the diversity of recommendation systems (Zhou et al. 2010; Zhang and Hurley 2008; Yu, Lakshmanan, and Amer-Yahia 2009). Qin and Zhu (Qin and Zhu 2013) used an entropy regularizer to characterize the item diversity in order to improve the top-K prediction. Lathia *et al.* (Lathia et al. 2010) exploited the temporal characteristics of user ratings and improved the diversity of recommended lists. Yin *et al.* (Yin et al. 2011) focused on mining and ranking the diversified trajectory pattern in social media while Zhang *et al.* (Zhang et al. 2014) diversified the spatial search results to improve service in road networks. To be best of our knowledge, the diversity of POI recommendations has not been explored so far. Our work may contribute into attractive research on diversifying POI recommendations.

Problem Formulation

Existing POI recommendation techniques compute a score for each POI and recommend the highly ranked ones. As the recommended POIs are computed based on most of users' personally relevant information, we call the resulting scores as the $relevance$ measurement in the recommendation. We will utilize the well-developed techniques to compute the relevance function.

Finding top-K LC-POIs is to optimize the relevance and information coverage of a set of POIs based on check-in data. We provide a sample of check-ins in Table 1. As some of the previously available check-ins lack proper categories, we locate the POIs through the reference of latitude and longitude and label them with the well-defined category hierarchy in *Foursquare*.

We first choose the state-of-art POI recommendation technique to compute the relevance function. Subsequently, we

Table 1: Sample of User Check-in Sequences.

User-ID	Check-in Time	POI-ID	(Lati., Long.)	Category
user-1	20120603, 17:23	POI-2	(1.31,103.85)	*Mall*
user-2	20120605, 08:23	POI-1	(1.31,103.85)	*Zoo*
user-1	20120703, 22:23	POI-3	(1.29,103.84)	*Bar*
...	...	...	...	...

formally develop information coverage function and define a multi-objective optimization problem for top-K LC-POIs recommendation. We prove hardness of the new recommendation problem.

POI Relevance

As suggested in most of the previous research, the relevance computation follows user-based collaborative filtering methods and can be implemented in a unified framework (Ye et al. 2011b). It is further improved by considering the temporal information in check-ins (Yuan et al. 2013).

Let $L = \{l_1, \cdots, l_m\}$ be a set of POIs. For a given user u at time t, we compute the relevance score for a POI in Eq.1.

$$R(l_j) = \alpha \times c^t_{u,t,l_j} + (1-\alpha) \times c^s_{u,t,l_j} \quad (1)$$

where c^t_{u,t,l_j} is the recommendation score that user u will visit l_j at time t and is computed based on the similarity of users' check-in activities, c^s_{u,t,l_j} the spatial influence of u's previously visited POIs and α the tuning parameter. Time t can be some day, e.g., Monday, or a particular time, e.g., night.

The relevance score $R(L)$ for a set of POIs is the sum of scores for all POIs. It is computed in Eq. 2.

$$R(L) = \sum_{l_j \in L} R(l_j) \quad (2)$$

Note that the relevance function in Eq. 2 is one of the state-of-the-art methods (UST: the user-spatial-temporal unified framework) for POI recommendation (Yuan et al. 2013). It recommends conventional top-K POIs based on the relevance factor. Our proposed method is equally applicable if other methods of computing the relevance are used.

Information Coverage

Information coverage considers how a set of POIs, $L = \{l_1, \cdots, l_m\}$, collectively enclose different location categories derived from check-ins. In general, one category contains a set of POIs and one POI may be labelled with multiple categories in check-ins. The degree to which one POI covers a category reflects the popularity of the POI in the corresponding category. The larger degree of covering all categories, the more information the set of POIs provide in the recommendation. A POI enjoys different popularity of being in one category when it is checked at different time points. We consider time-aware information coverage in this paper.

Let $A = \{a_1, \cdots, a_q\}$ be a set of location categories and $w^t_{a_q}(>0)$ the weight of category a_q at time t. We compute the information coverage of a set of POIs in Eq. 3.

$$I(L) = \sum_{a_q \in A} w^t_{a_q} cov^t_{a_q}(L) \quad (3)$$

where $cov^t_{a_q}(L)$ measures the degree to which category a_q is covered by at least one POI in the set L at time t. Thus we compute $cov^t_{a_q}(L)$ below.

$$cov^t_{a_q}(L) = 1 - \prod_{l_j \in L}[1 - cov^t_{a_q}(l_j)] \quad (4)$$

where $cov^t_{a_q}(l_j)$ is the degree to which POI l_j covers the category a_q at t.

The popularity of a POI labelled by the category a_q is implied by the number of checks-in that the POI receives at a_q. To equally prioritize POIs with a high volume of check-ins, we compute $cov^t_{a_q}(l_j)$, which is proportional to users' check-ins of l_j in the average check-ins labelled by a_q, in Eq. 5.

$$cov^t_{a_q}(l_j) = min[\frac{nc^t_{l_j^{a_q}}}{\frac{1}{\sum_{l_j} 1} \times \sum_{l_j} nc^t_{l_j^{a_q}}}, 1] \quad (5)$$

where $nc^t_{l_j^{a_q}}$ is the number of check-ins that are made at l_j and labelled by category a_q at time t. $\sum_{l_j} 1$ is the number of POIs visited at time t and $\sum_{l_j} nc^t_{l_j^{a_q}}$ counts all check-ins labelled by a_q at t. $cov^t_{a_q}(l_j)$ is 1 if $nc^t_{l_j^{a_q}}$ exceeds the average number of check-ins labelled by a_q.

Example 1. *Given 3 POIs (l_1, l_2 and l_3) labelled by a_q in one city, the numbers of check-ins are 50, 100 and 150 respectively on l_1, l_2 and l_3 at time t. Hence the average number of check-ins labelled by a_q is 100 for POIs at time t. Subsequently we get: l_1 covers a_q with a degree 0.5 while l_2 and l_3 cover a_q completely (with a degree 1).*

In general, a city is featured by its local attractions of categories that become common in check-ins. To offset the focused categories, we adapt the TF-IDF (Term Frequency-Inverse Document Frequency) technique to compute the category weight $w^t_{a_q}$ in Eq. 6.

$$w^t_{a_q} = \frac{nc^t_{a_q}}{\sum_{a_q} nc^t_{a_q}} \times (lg\frac{\sum_{a_q} z^t_{a_q}}{z^t_{a_q}} + \tau) \quad (6)$$

where $nc^t_{a_q}$ (=$\sum_{l_j} nc^t_{l_j^{a_q}}$) is the number of check-ins labelled by a_q at t and $z^t_{a_q}$ counts the number of city in which at least one POI is labelled by a_q at t. We set $\tau = \frac{1}{\sum_{a_q} z^t_{a_q}}$ to maintain a positive value of $w^t_{a_q}$.

As the information coverage function needs to consider the joint influence of a set of POIs, it differs from the relevance function (in Eq. 2) that can be calculated separately for each POI.

Top-K LC-POIs Recommendation

We proceed to formulate the top-K LC-POIs recommendation problem. The two factors, namely relevance and information coverage, are objectives that shall be balanced when we expect to optimize the recommendation list. For a given user u, the problem is to find a set of POIs that maximize the scoring function $\sigma(L)$ by computing their relevance and information coverage in check-in data. Formally the top-K

LC-POIs recommendation is modeled as one multi-objective optimization problem below.

$$\begin{array}{l}\textbf{Given}: \mathcal{D}, K, \beta, u, t \\ \textbf{Objective}: \\ max_{L \subseteq \mathcal{D}, |L|=K} \ \sigma(L) = (1-\beta) \times R(L) + \beta \times I(L)\end{array} \tag{7}$$

where $L \subseteq \mathcal{D}$ retrieves all POIs, denoted as $POI\text{-}ID$, from check-in data $\mathcal{D}$, and $\beta(\geq 0)$ is a tradeoff between relevance and information coverage measurements.

The limited number (K) of POIs to be recommended is supplied by a user who may have limited budget (e.g., time and money) in a visit. Another parameter β is determined by the user's plan on either exploiting some areas (following her/his personal experience) or exploring the entire city (tasting all flavor of a city). Intuitively, β is set to be small when the user has visited the city for several times and expects to focus on some specific areas in a new visit. This however depends on the user's personal interests.

We observe that the top-K LC-POIs recommendation is a complex combinatorial optimization problem with two objectives. We prove it to be NP-hard.

Proposition 1. *The top-K LC-POIs recommendation problem formulated in Eq.7 is NP-hard.*

Proof. We develop the proof by converting the problem into a unit cost version of the budgeted maximum coverage problem (Khuller, Moss, and Naor 1999). Given a unit cost version of the budgeted maximum coverage (UBMC) problem instance φ : a collection of sets $S = \{S_1, S_2, \cdots, S_m\}$ with a unit cost C, a domain of elements $X = \{x_1, x_2, \cdots, x_n\}$ with associated weights $\{w_1, w_2, \cdots, w_n\}$, and a budget B, we can construct a top-K LC-POIs recommendation problem instance ω by setting $\beta = 1$, $K = \lfloor B/C \rfloor$ and $I(S')$ corresponds to the total weight of the elements covered by S'. Hence, S' is the set having a maximum weight in φ iff S' is the top-K LC-POI set of ω. As the UBMC problem has been proved to be NP-hard, the top-K LC-POIs recommendation problem is NP-hard as well. ■

Property Analysis and Greedy Algorithm

It is rather difficult to solve the top-K LC-POIs recommendation problem. By leveraging the monotone submodularity of the scoring function ($\sigma(L)$ in Eq. 7), we present a greedy algorithm to find top-K LC-POIs and improve its efficiency.

Monotone Submodularity

Let $\mathcal{V}$ be a finite set. A set of function F: $\mathcal{V} \rightarrow \mathbb{R}$ is called *submodular* if it satisfies the *diminishing returns* property (Nemhauser, Wolsey, and Fisher 1978), $F(B \bigcup s) - F(B) \geq F(\hat{B} \bigcup s) - F(\hat{B})$, for all $B \subseteq \hat{B} \subseteq \mathcal{V}$ and $s \notin B$. $F(B \bigcup s) - F(B)$ is the *marginal increase* of F when an element s is added into B. Submodularity characterizes the notion that supplementing elements to a small set B provides more than doing it to a larger set $\hat{B}$.

To prove the monotone submodularity of $\sigma(L)$, we first analyze the properties of $R(L)$ and $I(L)$ respectively. The relevance analysis is straightforward because $R(L)$ computes scores for every POI independently in Eq. 1.

Proposition 2. *The relevance function $R(L)$ is monotone and submodular.*

Proof. Let $L_1 \subseteq L_2 \subseteq L$ and $l_i \in L$. For any $l_i \notin L_1$, we compute

$$R(L_1 \bigcup l_i) - R(L_1) = [R(L_1) + R(l_i)] - R(L_1) = R(l_i)$$

As the relevance score for a POI is nonnegative, $R(L)$ is monotone. Similarly, we have $R(L_2 \bigcup l_i) - R(L_2) = R(l_i)$. This leads to $R(L_1 \bigcup l_i) - R(L_i) \geq R(L_2 \bigcup l_i) - R(L_2)$. Hence the relevance function $R(L)$ is submodular. ■

Intuitively, users know more about a city when they visit more places in a city. However, visiting one place after traveling a small part of the city provides more knowledge to them than visiting the place after traveling a larger part of the city. This indicates the property of information coverage function on the monotone submodularity. We formulate it in Proposition 3.

Proposition 3. *The information coverage function $I(L)$ is monotone and submodular.*

Proof. Let $L_1 \subseteq L_2 \subseteq L$ and $l_i \in L$. For any L_1 and $l_i \notin L_1$, we compute

$$\begin{array}{l}cov^t_{a_q}(L_1 \bigcup l_i) - cov^t_{a_q}(L_1) = \prod_{l_j \in L_1}(1 - cov^t_{a_q}(l_j)) - \\ \prod_{l_j \in L_1 \bigcup l_i}(1 - cov^t_{a_q}(l_j)) = cov^t_{a_q}(l_i) \prod_{l_j \in L_1}(1 - cov^t_{a_q}(l_j))\end{array}$$

Since both $cov^t_{a_q}(l_i)$ and $cov^t_{a_q}(l_j)$ are in the range [0,1], we get $cov^t_{a_q}(L_1 \bigcup l_i) \geq cov^t_{a_q}(L_1)$. Thus $cov^t_{a_q}(L)$ is monotone. We proceed to compute

$$\begin{array}{l}[cov^t_{a_q}(L_1 \bigcup l_i) - cov^t_{a_q}(L_1)] - [cov^t_{a_q}(L_2 \bigcup l_i) - cov^t_{a_q}(L_2)] \\ = cov^t_{a_q}(l_i) \prod_{l_j \in L_1}(1 - cov^t_{a_q}(l_j))(1 - \prod_{l_j \in L_2 - L_1}(1 - cov^t_{a_q}(l_j))\end{array}$$

We have $cov^t_{a_q}(L_1 \bigcup l_i) - cov^t_{a_q}(L_1) \geq cov^t_{a_q}(L_2 \bigcup l_i) - cov^t_{a_q}(L_2)$. Hence $cov^t_{a_q}(L)$ is submodular.

In Eq. 3, $I(L)$ is a linear combination of $cov^t_{a_q}(L)$ weighted by $w^t_{a_q}(>0)$. The monotone submodularity is closed under a linear combination with a nonnegative weight. This concludes that the information coverage function is monotone submodularity. ■

Given the above propositions, we obtain the attractive property of the scoring function $\sigma(L)$ in Proposition 4.

Proposition 4. *The scoring function $\sigma(L)$ is monotone submodularity.*

Proof. It follows that $\sigma(L)$ is a linear combination of $R(L)$ and $I(L)$ both of which are monotone submodularity and the weighting parameter β is nonnegative. ■

Greedy Algorithm and Its Optimization

Greedy Algorithm. As shown in Proposition 1, finding top-K LC-POIs is NP-hard in the recommendation. However, the monotone submodularity property suggests a greedy algorithm with theoretical guarantees for maximizing a multi-objective function (Nemhauser, Wolsey, and Fisher 1978). In Alg. 1, the greedy algorithm starts with an empty set of POI (line 1) and repeatedly adds the POI incurring the largest marginal score increase to the POI set L until $|L|$ =

K (lines 5-7). The algorithm can achieve near-optimal solutions of top-K LC-POIs with a (1-$\frac{1}{e}$) approximation on the optimal score.

Since the greedy algorithm needs to check all of the POI candidates in every round (line 6), the time complexity is $\mathcal{O}[K|\mathcal{D}|\mathcal{T}(\sigma(L))$, where $|\mathcal{D}|$ is the size of check-in data and $\mathcal{T}(\sigma(L))$ the run time for computing the scoring function.

Pruning Optimization. The large size of check-ins ($|\mathcal{D}|$) prevents a quick response from the greedy algorithm on finding top-K LC-POIs in real time. We improve its efficiency by pruning the POIs having a small relevance and information coverage value. The operations of *Pruning Optimization* are embedded in the greedy algorithm (lines 2-4). The POI to be pruned has a smaller value of $\sigma(l_i)$ compared to the score $R(l_k)$ of the K^{th} POI that is ranked by the relevance measurement in check-ins. With a general setting of $K \ll |\mathcal{D}|$, the operations prune a significantly large number of POIs. Meanwhile, the pruning maintains the solution quality of the greedy algorithm. We prove the property.

Algorithm 1: Greedy Algorithm with Pruning Optimization

Input: $\mathcal{D}$, K, β, u, t
Output: A set of POIs, L, with $|L|$=K

1 Initialize L=$\emptyset$;
2 Compute $\sigma(l_i)$ for each $l_i \in \mathcal{D}$;
3 Rank $\mathcal{D}$ in decreasing order of $R(l_i)$;
4 Prune the POI set $L_{pru} = \{l_i|\sigma(l_i) < (1-\beta) \times R(l_K)\}$ where l_K is the K^{th} POI ranked by the relevance score;
5 **for** $j = 1$ **to** K **do**
6 $\quad l_j \leftarrow argmax_{l_j}\ [\sigma(L \bigcup l_j) - \sigma(L)]$;
7 $\quad L \leftarrow L \bigcup l_j$;
8 **return** L;

Proposition 5. *Pruning Optimization preserves the solution quality of the greedy algorithm.*

Proof. Assume that there exists a POI $l_a \in L_{pru}$ belonging to top-K LC-POIs recommendation $L_{K-1} \bigcup l_a$, where L_{K-1} is top-$(K-1)$ LC-POIs recommendation. Let L_{rel} be top-K relevant POIs set (containing the first K POIs of sorted $\mathcal{D}$ in line 3 of Alg. 1), l_b be the POI meeting $l_b \in L_{rel}$ and $l_b \notin L_{K-1} \bigcup l_a$. As $(1-\beta) \times R(l_a) + \beta \times I(l_a) < (1-\beta) \times R(l_b)$, $\sigma(L_{K-1} \bigcup l_a) < \sigma(L_{K-1}) + (1-\beta) \times R(l_a) + \beta \times I(l_a) < \sigma(L_{K-1}) + (1-\beta) \times R(l_b) < \sigma(L_{K-1} \bigcup l_b)$. Thus, we get a better K POIs recommendation $L_{K-1} \bigcup l_b$, it contradicts the assumption. This implies that each POI in L_{pru} is not a candidate of top-K LC-POIs. Hence it is safe for Pruning Optimization to prune the POIs. ■

Experimental Study

We conducted a series of experiments to study the top-K LC-POIs recommendation problem and demonstrated performance of the proposed approaches compared to state-of-the-art recommendation techniques.

Experimental Settings

Datasets. We used two real-world check-in datasets. One was collected from $Foursquare$ which was made in Singapore between Aug. 2010 and Jul. 2011 (Yuan et al. 2013). The other one is from $Gowalla$ which was made in Austin between Nov. 2009 and Oct. 2010 (Cho, Myers, and Leskovec 2011). Each check-in contains the aforementioned attributes in Table 1. To fill in the missing values of $Category$, we implemented the tool based on the $Foursquare$ APIs. For both datasets, we removed users who have checked in fewer than 5 POIs, and then removed POIs that were checked by fewer than 5 users. After preprocessing, the $Foursquare$ dataset has 189,306 check-ins made by 2,321 users at 5,412 POIs, and the $Gowalla$ dataset contains 201,525 check-ins made by 4,630 users at 6,176 POIs. For each user, we randomly mark off 20% of his/her visited POIs as the testing data to evaluate the recommendation methods. The recommendation is done for a user given a specific time Day. To compute Eq. 6, we extracted $Gowalla$ check-ins made in 162 cities from the dataset provided by *Cho et. al* (Cho, Myers, and Leskovec 2011).

Comparative Recommendation Techniques. We implemented both the greedy algorithm (GA) and its improved version with the pruning optimization (GA+PO) to find top-K LC-POIs. Additionally, we adopt the CELF optimization (Leskovec et al. 2007), which is widely used to improve the efficiency of GA, in the implementation of GA+PO. For the comparison purpose, we implemented a random algorithm (Random) which selects K POIs from the dataset randomly and repeats the procedure for a sufficient number of times (10,000). Note that all of the four algorithms recommend top-K LC-POIs.

To demonstrate the quality of top-K LC-POIs, we implemented the state-of-the-art method (UST) that recommends conventional top-K POIs solely based on the relevance score. All methods are implemented in JAVA, and experiments are conducted on a Windows PC with a 4-core Intel i7-3770 3.4GHz CPU and 8 GB memory.

Performance Metrics. The evaluated recommendation methods aim to find top-K LC-POIs to the targeted user by computing the scoring function and then rank candidate POIs accordingly. We employ the Shannon entropy (Cover and Thomas 1991) as a measurement (div@K) to show the category diversity of the recommended POIs in Eq. 8.

$$div@K = -\frac{\sum_{a_q} \frac{|l_j^{a_q}|}{K} ln(\frac{|l_j^{a_q}|}{K})}{lnK} \quad (8)$$

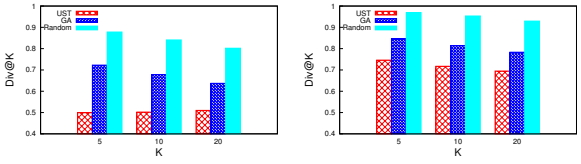
where $|l_j^{a_q}|$ is the number of POIs labelled by the category a_q.

Meanwhile we use pre@K and rec@K to measure how many POIs in the recommended POIs correspond to the hold-off POIs in the testing data and how many POIs in the hold-off POIs in the testing set are returned as the recommended POIs respectively (Yuan et al. 2013).

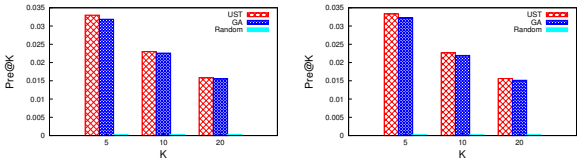
Performance of Methods

Effectiveness of Methods. We compare the recommendation quality of three methods (GA, UST and Random) with the settings of β=0.3 and 0.2 respectively in Foursquare and Gowalla. Note that GA and its enhanced versions (GA+PO and GA+PO+CELF) recommend the same top-K LC-POIs.

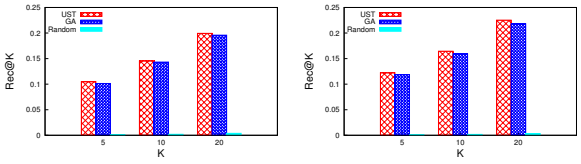
Fig. 3 shows that GA improves the diversity of UST by around 35% in $Foursquare$ and 13% in $Gowalla$. Meanwhile, it keeps around 97% precision and recall of UST on both datasets. Although Random achieves the best diversity, it leads to an extremely low recommendation precision and recall. Overall the results demonstrate that Random can not offer a good recommendation while GA improves the category diversity of UST significantly and achieve similar precision and recall as UST does.



(a) Div@K-Foursquare (b) Div@K-Gowalla



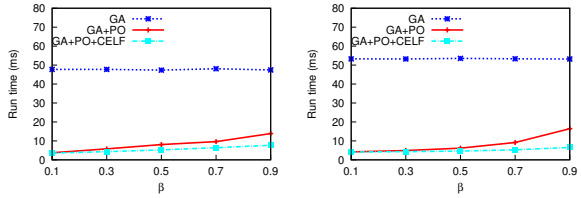
(c) Pre@K-Foursquare (d) Pre@K-Gowalla



(e) Rec@K-Foursquare (f) Rec@K-Gowalla

Figure 3: Comparison of the diversity, precision and recall of top-K POIs recommended by the methods.

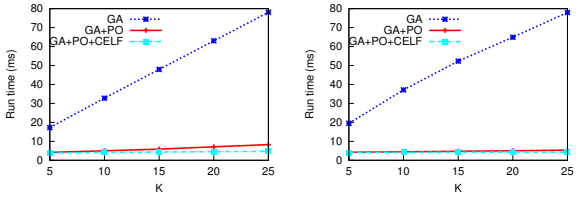
Efficiency of Methods. We compare the methods (GA, GA+PO and GA+PO+CELF) based on run time each takes to identify top-K LC-POIs in two datasets. Fig. 4 shows the run time of the methods varying β with $K = 15$. GA consumes much more time than other methods. A large β shrinks the pruned POI set, which results in the increase of the run time for GA+PO. Fig. 5 exhibits the run time of the methods varying K with a fixed β (0.3 on Foursquare and 0.2 on Gowalla). The run time of GA+PO and GA+PO+CELF is small and stable while the run time of GA grows quickly as K increases.



(a) Foursquare and $K = 15$ (b) Gowalla and $K = 15$

Figure 4: Comparison of methods varying β on run time

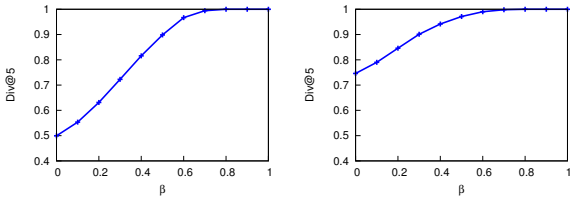
Effect of Parameter β. The parameter β balances the relevance and information coverage factors in the POI recommendation. If $\beta = 0$, top-K LC-POIs become conventional



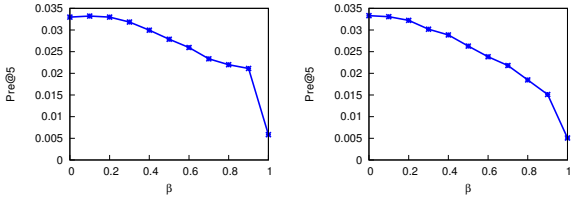
(a) Foursquare and $\beta = 0.3$ (b) Gowalla and $\beta = 0.2$

Figure 5: Comparison of methods varying K on run time.

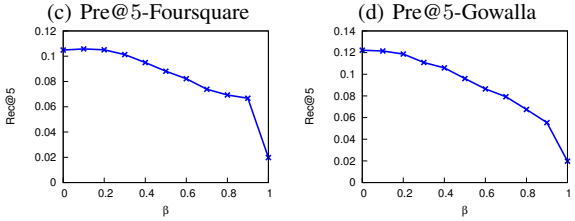
top-K POIs, and if $\beta = 1$, top-K LC-POIs maximize the information coverage. Fig. 6 shows the effect of β with $K = 5$ on the recommendation. With the increase of β, the diversity grows while the precision and recall decrease. Tuning β in this fashion allows top-K LC-POIs to be customized and optimized for needs of different users or communities.



(a) Div@5-Foursquare (b) Div@5-Gowalla



(c) Pre@5-Foursquare (d) Pre@5-Gowalla



(e) Rec@5-Foursquare (f) Rec@5-Gowalla

Figure 6: Effect of parameter β with $K = 5$.

Conclusion

With the observation of users' interests in exploring new location categories, we improve the POI recommendation by introducing information coverage into the recommendation measurement. We formulate the top-K LC-POIs recommendation problem and prove its monotone submodularity property. To solve the new problem, we propose a greedy algorithm with (1-$\frac{1}{e}$) theoretical bound and improve it using one pruning optimization. We empirically demonstrate the quality of top-K LC-POIs recommendation as well as the performance of our proposed methods. Further research can be conducted on improving POI recommendation through users' explicit feedback in LBSNs.

Acknowledgements

Yifeng Zeng would thank the DFI support in Teesside University. Yanping Xiang is the corresponding author. Gao Cong was supported in part by a grant awarded by a Singapore MOE AcRF Tier 2 Grant (ARC30/12).

References

Bao, J.; Zheng, Y.; and Mokbel, M. F. 2012. Location-based and preference-aware recommendation using sparse geo-social networking data. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, 199–208. ACM.

Cheng, C.; Yang, H.; King, I.; and Lyu, M. R. 2012. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, volume 12, 1.

Cho, E.; Myers, S. A.; and Leskovec, J. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1082–1090. ACM.

Cover, T. M., and Thomas, J. A. 1991. *Elements of Information Theory*. New York, USA: Wiley-Interscience.

El-Arini, K.; Veda, G.; Shahaf, D.; and Guestrin, C. 2009. Turning down the noise in the blogosphere. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, 289–298.

Gao, H.; Tang, J.; and Liu, H. 2012. gscorr: Modeling geo-social correlations for new check-ins on location-based social networks. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*, 1582–1586.

Khuller, S.; Moss, A.; and Naor, J. S. 1999. The budgeted maximum coverage problem. *Information Processing Letters* 70(1):39–45.

Konstas, I.; Stathopoulos, V.; and Jose, J. M. 2009. On social networks and collaborative recommendation. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 195–202.

Lathia, N.; Hailes, S.; Capra, L.; and Amatriain, X. 2010. Temporal diversity in recommender systems. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 210–217.

Liu, B., and Xiong, H. 2013. Point-of-interest recommendation in location based social networks with topic and location awareness. In *SDM*, 396–404. SIAM.

Liu, B.; Fu, Y.; Yao, Z.; and Xiong, H. 2013. Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1043–1051. ACM.

Nemhauser, G.; Wolsey, L.; and Fisher, M. 1978. An analysis of the approximations for maximizing submodular set functions. *Mathematical Programming* 14:265–294.

Noulas, A.; Scellato, S.; Mascolo, C.; and Pontil, M. 2011. Exploiting semantic annotations for clustering geographic areas and users in location-based social networks. In *Proceedings of the Social Mobile Web*.

Qin, L., and Zhu, X. 2013. Promoting diversity in recommendation by entropy regularizer. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, 2698–2704.

Ye, M.; Shou, D.; Lee, W.-C.; Yin, P.; and Janowicz, K. 2011a. On the semantic annotation of places in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 520–528.

Ye, M.; Yin, P.; Lee, W.-C.; and Lee, D.-L. 2011b. Exploiting geographical influence for collaborative point-of-interest recommendation. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 325–334.

Ye, M.; Liu, X.; and Lee, W.-C. 2012. Exploring social influence for recommendation: A generative model approach. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 671–680.

Ye, M.; Yin, P.; and Lee, W.-C. 2010. Location recommendation for location-based social networks. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS)*, 458–461.

Ye, J.; Zhu, Z.; and Cheng, H. 2013. What's your next move: User activity prediction in location-based social networks. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 171–179.

Yin, Z.; Cao, L.; Han, J.; Luo, J.; and Huang, T. S. 2011. Diversified trajectory pattern ranking in geo-tagged social media. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 980–991.

Yin, H.; Sun, Y.; Cui, B.; Hu, Z.; and Chen, L. 2013. Lcars: A location-content-aware recommender system. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 221–229.

Yu, C.; Lakshmanan, L.; and Amer-Yahia, S. 2009. It takes variety to make a world: Diversification in recommender systems. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology (EDBT)*, 368–378.

Yuan, Q.; Cong, G.; Ma, Z.; Sun, A.; and Thalmann, N. M. 2013. Time-aware point-of-interest recommendation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 363–372.

Zhang, M., and Hurley, N. 2008. Avoiding monotony: Improving the diversity of recommendation lists. In *Proceedings of the ACM Conference on Recommender Systems (RecSys)*, 123–130.

Zhang, C.; Zhang, Y.; Zhang, W.; Lin, X.; Cheema, M. A.; and Wang, X. 2014. Diversified spatial keyword search on road networks. In *Proceedings of the 17th International Conference on Extending Database Technology (EDBT)*, 367–378.

Zheng, Y.; Zhang, L.; Xie, X.; and Ma, W.-Y. 2009. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th International Conference on World Wide Web (WWW)*, 791–800.

Zheng, V. W.; Zheng, Y.; Xie, X.; and Yang, Q. 2010a. Collaborative location and activity recommendations with gps history data. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, 1029–1038.

Zheng, V. W.; Cao, B.; Zheng, Y.; Xie, X.; and Yang, Q. 2010b. Collaborative filtering meets mobile recommendation: A user-centered approach. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, 236–241.

Zheng, Y. 2011. Location-based social networks: Users. *Computing with Spatial Trajectories* 243–276.

Zhou, T.; Kuscsik, Z.; Liu, J.-G.; Medoa, M.; Wakeling, J. R.; and Zhang, Y. C. 2010. Solving the apparent diversity-accuracy dilemma of recommender systems. In *Proceedings of the National Academy of Sciences (PNAS)*, 4511–4515.