# Learning from Hometown and Current City: Cross-city POI Recommendation via Interest Drift and Transfer Learning

JINGTAO DING*, GUANGHUI YU*, YONG LI, and DEPENG JIN, Beijing National Research Center for Information Science and Technology (BNRist), Department of Electronic Engineering, Tsinghua University, Beijing
HUI GAO, Tencent Inc., Beijing

With more and more frequent population movement between different cities, like users' travel or business trip, recommending personalized cross-city Point-of-Interests (POIs) for these users has become an important scenario of POI recommendation tasks. However, traditional models degrade significantly due to sparsity problem because travelers only have limited visiting behaviors. Through a detailed analysis of real-world check-data, we observe 1) the phenomenon of travelers' interest drift and transfer co-exist between hometown and current city; 2) differences between popular POIs among locals and travelers. Motivated by this, we propose a POI Recommendation framework with User Interest Drift and Transfer (PR-UIDT), which jointly considers above two factors when designing user and POI latent vector. In this framework, user vector is divided into a city-independent part and another city-dependent part, and POI is represented as two independent vectors for locals and travelers, respectively. To evaluate the proposed framework, we implement it with a square error based matrix factorization model and a ranking error based matrix factorization model, respectively, and conduct extensive experiments on three real-world datasets. The experiment results demonstrate the superiority of PR-UIDT framework, with a relative improvement of 0.4% ~ 20.5% over several state-of-the-art baselines, as well as the practicality of applying this framework to real-world applications and multi-city scenarios. Further qualitative analysis confirms both the plausibility and validity of combining user interest transfer and drift into cross-city POI recommendation.

CCS Concepts: • **Information systems** → **Recommender systems**; *Location based services*; *Collaborative filtering*; • **Human-centered computing** → Ubiquitous and mobile computing systems and tools.

Additional Key Words and Phrases: Cross-city POI recommendation, interest drift and transfer, matrix factorization

**ACM Reference Format:**
Jingtao Ding, Guanghui Yu, Yong Li, Depeng Jin, and Hui Gao. 2019. Learning from Hometown and Current City: Cross-city POI Recommendation via Interest Drift and Transfer Learning. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 4, Article 131 (December 2019), 28 pages. https://doi.org/10.1145/3369822

## 1 INTRODUCTION

With the widespread popularity of Location-based social networks (LBSNs) such as Yelp and Gowalla, people are more and more willing to share their visited and interested locations, i.e., point of interests (POIs), to the

*Both authors contributed equally to this research.

Authors' addresses: Jingtao Ding, dingjt15@mails.tsinghua.edu.cn; Guanghui Yu, ygh690181479@gmail.com; Yong Li, liyong07@tsinghua.edu.cn; Depeng Jin, jindp@tsinghua.edu.cn, Beijing National Research Center for Information Science and Technology (BNRist), Department of Electronic Engineering, Tsinghua University, Beijing; Hui Gao, Tencent Inc., Beijing, huigaogao@tencent.com.


2474-9567/2019/12-ART131 $15.00
https://doi.org/10.1145/3369822

public. This provides a huge opportunity for LBSN service providers to recommend new POIs to users based on their visiting behaviours, which can not only improve the user engagement, but also bring potential profits in commercial perspective. As a specific scenario of POI recommendation, cross-city recommendation, that is to recommend a new POI for users who have limited records in a strange city, is especially difficult but quite practical. When users are in their familiar city or region, like their hometown or working city, they know this city so well that they don't rely on a POI recommender system to guess their interests and help make a choice. On the contrary, while in unfamiliar cities, for example where users are travelling or on business, they have visited few places in a strange city and thus can not explore every region to find their favourites. Under this circumstance, they are in bad need of a smart recommender system that can help find a suitable POI. Therefore, in this paper, we focus on cross-city POI recommendation that targets for the travelers in unfamiliar cities.

One of the most challenging problems for cross-city POI recommendation is the sparsity of interactions between travelers and POIs in current unfamiliar city. Our statistics on real-world datasets reveals that user-POI interactions in an unfamiliar city are about one less order of magnitude than those in home city on average. Therefore, the sparse user data in the unfamiliar city are far from enough to train a powerful recommender system. To relieve this sparsity problem, three types of methods have been considered. First, a direct solution is to transfer knowledge from the howntown into current city, as developed in previous works [9, 35, 38]. However, for cross-city POI recommendation, it is the knowledge about user interest that should be transferred, while above works focus on context-based knowledge such as the semantic features of locations. Second, with a close relation to cold-start problems, the sparsity issue of user-POI interactions in current city can also be alleviated by introducing the content information of POIs (categories or tags) [30, 34, 43] or borrowing the POI preference from friends [6, 31]. However, users tend to have different interests when they travel in different cities that have different urban compositions (*i.e.* the phenomenon of *user interest drift*), while these methods assume that user interest towards POIs will not change between hometown and current city, which is incorrect and thus can degrade the model performance. Therefore, the third type of methods based on probabilistic generative models are proposed to consider the *user interest drift*. Yin *et al.* [42] assume that a traveler's decision on visiting which POI is dependent on the target spatial region, indicating a different interest compared to that in hometown, and the preference of other travelers in this region is also leveraged to alleviate the data sparsity. Li *et al.* [16] further enhance the learning of traveler preference by separating the city-specific topics of each city from the common topics shared by all cities. Although the *user interest drift* is common and reasonable, it should also be noted that users still have some interests that are invariant among different cities. Therefore, cross-city POI recommendation should also consider this phenomenon of *user interest transfer*, which requires joint learning of users' previous visits in both two cities and has not been captured by above methods.

In this work, we design a novel embedding-based cross-city POI recommendation framework named **P**OI **R**ecommendation enhanced with **U**ser **I**nterest **D**rift and **T**ransfer (PR-UIDT) that learns the user preference from both hometown and current city. Different from previous works, we propose to combine *interest drift* and *interest transfer* together, utilizing users' visiting behaviors across two cities to help improve POI recommendation in the current city. Here we define the *interest drift* as the phenomenon that user preference over POIs varies among different cities, and, in contrast, *interest transfer* as the phenomenon that the preference remains unchanged. In the PR-UIDT framework, each user embedding, i.e., a latent vector, is divided into a city-independent part and city-dependent part, corresponding to the inherent interest and drifted interest respectively. On the other hand, the embeddings for each POI are represented by two independent vectors, which are designed for the travelers and locals in this city, respectively. Moreover, as the visitors from both locals and travelers should contribute in learning the better POI feature, we further add an incentive that motivates these two POI vectors to become similar. Finally, above novel designs of embeddings are integrated into matrix factorization (MF) model using two different loss functions (*i.e.*, the square error based loss [13] and the ranking error based loss [27]). With the recent advance of deep learning based interaction models in recommender systems [11, 14, 39], our proposed PR-UIDT framework can easily integrate them by replacing the basic MF module.

The main contributions of this paper are summarized below:

- To the best of our knowledge, we are the first to solve cross-city POI recommendation problem by combining user interest drift and transfer together, which enables learning the traverlers' preference from both hometown and current city, and thus overcome the sparsity issue of their visit records in current city.
- Our proposed POI recommendation framework jointly leverages interest drift and interest transfer through the novel designs of user and POI embeddings, where the transferable knowledge, from travelers' visit records in hometown and locals' visit records in current city, are incorporated to complement the insufficient information about the drifted interest, provided by limited travel visits in current city.
- Experiments on three real-world datasets are conducted to prove the performance superiority of our proposed framework in POI recommendation for travelers. A further ablation study and qualitative analysis reveals that the effective combination of interest drift and interest transfer leads to performance improvements. Time efficiency test and extension version with geographical effect demonstrate the utility of PR-UIDT for large-scale real-time application. Moreover, the applicability of PR-UIDT in multi-city scenarios has also been evaluated on real data.

## 2 DATASET AND OBSERVATION

### 2.1 Dataset Collection

In this work, we aim to solve the cross-city POI recommendation problem by modeling user interest drift and transfer. In order to validate the feasibility of this idea, we use three real-world POI check-in datasets which contain users' visits in both hometown and non-home city. Now we introduce them in details.

**Tencent dataset:** Tencent Wechat[1] is the biggest online social network service in China. Users can check-in with Wechat mobile App, which is known as Moment. We collect users' check-in records in three Chinese cities, *i.e.*, Beijing, Shanghai and Tianjin, within the time period from Jul. 2017 to Jul. 2018. Each check-in record includes an anonymous user-ID, a POI-ID, current city, hometown city and timestamp. We further obtain POI category via querying these POI-IDs through Wechat Map API[2]. To investigate users' cross-city POI visiting behaviors, we select two cross-city pairs, *i.e.*, *Beijing-Shanghai* and *Beijing-Tianjin*. As Beijing and Shanghai are the two largest cities in China, and Tianjin is a secondary city geographically close to Beijing, these two cross-city pairs are the representatives of typical cross-city scenarios. As for the *Shanghai-Tianjin*, it is similar to the *Beijing-Tianjin* case in terms of city scale. Thus we leave out it due to space limit and plan to investigate more diverse city-pairs in future work. For each cross-city pair, we first select the users that live in either one of the two cities and also have once traveled to another one. Each user is a local in the hometown city and a traveler in another city. Then we extract their visit records in corresponding cross-city pairs from the raw data for further study. Next we merge records of the same user and POI into a single record with the earliest timestamp, as we aim to recommend a new POI to the user. Finally we filter out inactive users and POIs with less than 10 interactions. Here we denote the records from *Beijing-Shanghai* and *Beijing-Tianjin* as Tencent-BS and Tencent-BT, respectively.

**Yelp dataset (public):** The datasets is Yelp's Challenge Dataset[3] that contains users' POI check-in records in more than 100 cities within the time period from Jan. 2006 to Jan. 2015. Each check-in record is stored as user-ID, POI-ID, POI location, POI category, check-in date. Since hometown information is not provided in this dataset, we choose the city where a user has the highest number of check-ins as his/her hometown. As for the cross-city pair, as this dataset is much sparser than Tencent dataset, we select the *Las Vegas-Phoenix* that contains highest

[1]https://weixin.qq.com
[2]https://lbs.qq.com
[3]https://www.yelp.com/dataset

Table 1. Major statistics of datasets.

| Datasets & Metrics | Tencent-BS | Tencent-BT | Yelp-LP |
|---|---|---|---|
| **City A-B** | Beijing - Shanghai | Beijing - Tianjin | Las Vegas - Phoenix |
| **Time Duration** | Jul. 2017 - Jul. 2018 | Jul. 2017 - Jul. 2018 | Jan. 2006 - Jan. 2015 |
| **Users of City A** | 42,113 | 71,998 | 2,625 |
| **Users of City B** | 34,483 | 38,148 | 2,089 |
| **POIs in City A** | 11,914 | 19,911 | 1,661 |
| **POIs in City B** | 10,934 | 14,396 | 1,481 |
| **A→B*** | 283,855 | 317,245 | 7,722 |
| **A→A** | 1,697,256 | 3,873,549 | 53,814 |
| **B→A** | 256,386 | 266,010 | 8,441 |
| **B→B** | 1,372,865 | 1,905,657 | 46,985 |

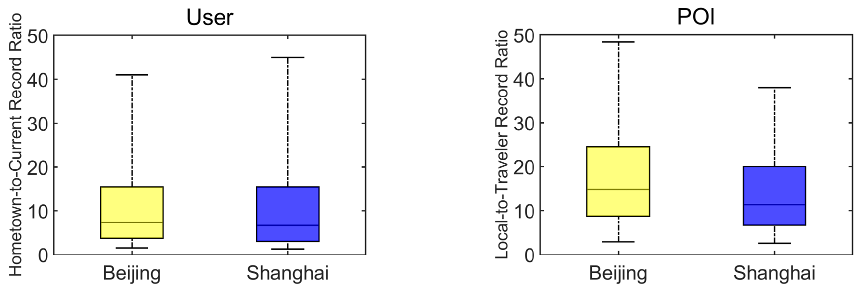
∗: Users of City A travel to City B.

number of cross-city visiting records. After the user-selecting and data preprocessing process that are similar to those in Tencent data, we extract a subset of raw data and denote it as Yelp-LP.

## 2.2 Basic Observation

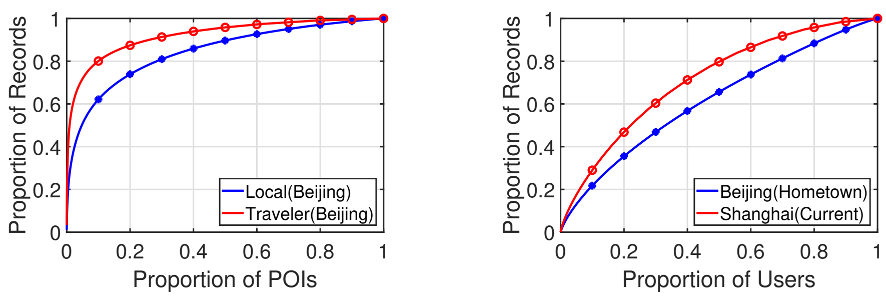
The major statistics of above datasets is shown in Table 1. First, for the difference among them, as Tencent Wechat is more frequently used in daily life, Tencent data is much denser in terms of user number, POI number, record number in total as well as the average per user/POI, which is more beneficial for experiments. More importantly, we can observe that the number of travelers' visits in current city (*e.g.*, "A→B") is about one magnitude smaller than both the number of locals' visits (*e.g.*, "B→B") and that of their own visits in hometown (*e.g.*, "A→A"). If these two parts of information, *i.e.*, the travelers' visiting behaviors in hometown and locals' visiting behaviors in current city, can be handled properly, then the recommendation performance of cross-city travelers can be largely improved. To further demonstrate the differences of above statistics, we take Tencent-BS as an example and show results in terms of each user or POI in Fig. 1. More specifically, it plots the distribution quantiles (5%, 25%, 50%, 75%, 95%) of both hometwon-to-current record ratio and local-to-traveler record ratio, where results of Beijing and Shanghai are presented together. Similar to the aggregation results, cross-city travelers visits more POIs in hometown, as the ratio value between record number in hometown and current city is about six in terms of the median among them (Fig. 1(a)). As for POIs, visit number of locals is over ten times larger than that of travelers among over 50% of them (Fig. 1(b)). Above observations further confirm the necessity of considering the travelers' visiting behaviors in hometown and locals' visiting behaviors in current city, which is more abundant and valuable for learning the user preference.

The popularity skewness exists in many types of human-item interaction behaviors inside a large population and it impacts the ability of capturing user interest for recommender systems [2, 4]. With higher number of interactions on those popular items, it becomes much harder for recommender systems to satisfy those users with niche interests. Here we investigate this effect in the cross-city POI recommendation scenario and show the results in Fig. 2(a). The y-axis represents the proportion of records for a given proportion of users or POIs on the x-axis, sorted by decreasing popularity, where the POI popularity is measured by number of visits from users. Clearly, travelers' visiting behaviors are more skewed in terms of popularity, where the top-10% of the POIs accounts for 80% of the visits, much larger than less 60% in locals. This finding is reasonable, as most travelers tend to visit some popular POIs like attractions, restaurants and so on. Moreover, it also means that there exist a certain proportion of travelers with niche interests, which are hard to learn with such sparse and skewed data only. Besides the popularity skewness in terms of POIs, we also investigate the activeness skewness in terms of




(a) $\frac{\text{\# of visits in hometown}}{\text{\# of visits in current city}}$ of each user (b) $\frac{\text{\# of visits by locals}}{\text{\# of visits by travelers}}$ of each POI

Fig. 1. Data sparsity of travelers' visits in Tencent-BS, in terms of quantiles (5%, 25%, 50%, 75%, 95%) of the ratio values.




(a) POIs in Beijing (b) users in Beijing

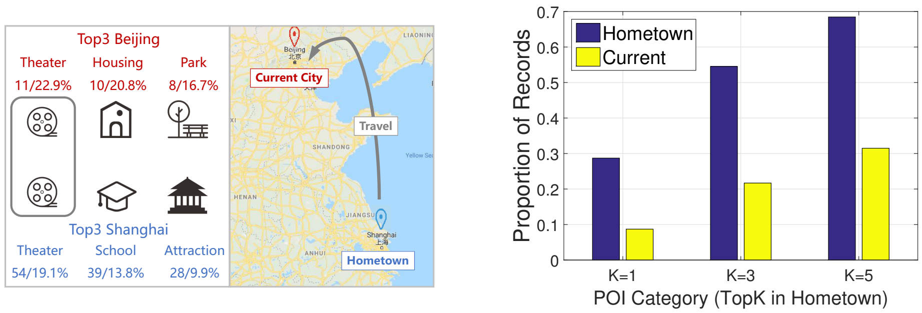
Fig. 2. (a) Popularity skewness in Tencent-BS, in terms of the proportions value of records contributed by a certain proportion of POIs. (b) Activeness skewness in Tencent-BS, in terms of the proportions value of records contributed by a certain proportion of users.

users in Fig. 2(b), where the user activeness is measure by visit number of each user. Similarly we observe that user activeness is more skewed among traverlers, indicating a higher difficulty of capturing their interest.

According to above analysis, travelers' visiting behaviors in current city are not only sparse, but also skewed in terms of POI popularity and user activeness. Therefore, it is not enough in cross-city POI recommendation to learn traveler interest solely from these limited information. With Tencent and Yelp datasets that contain the visiting behaviors in both hometown and current city, we are able to investigate this problem from a new perspective of combining user interest drift and transfer together.

## 3 MOTIVATION

In order to solve the major challenge of data sparsity existed in cross-city POI recommendation problems, both the travelers' visiting behaviors in hometown and locals' visiting behaviors in current city should be leveraged, which are much denser according to our basic statistic analysis. To successfully utilize these two parts of information, we conduct further analysis so as to guide the design of our proposed framework. On the one hand, for travelers themselves, we reveal the phenomenon of user interest drift and transfer between hometown and current city.




(a) toy example (b) mean value among all Shanghai users

Fig. 3. (a) A toy example showing a traveler's interests in hometown and current city. (b) Proportion values of each user's visits to $K$ POI categories in hometown and current city, respectively, where the $K$ categories are those ranked top $K$ by the number of his/her records in hometown.

On the other hand, for POIs in current city, we analyze the differences between popular POIs among locals and travelers.

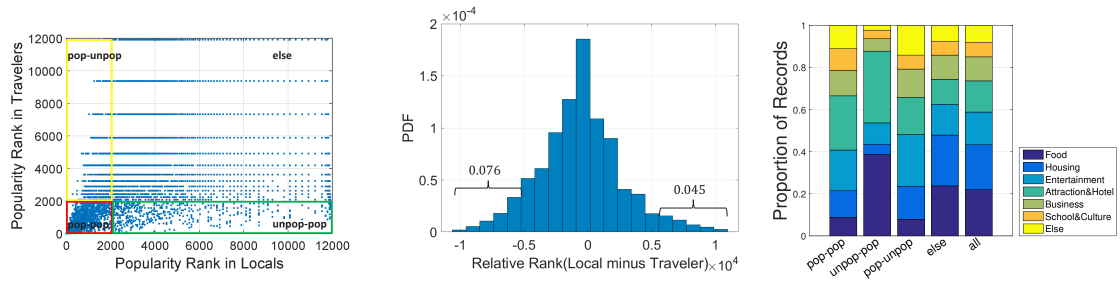
## 3.1 User Interest Drift and Transfer

Fig. 3(a) illustrates a toy example where a user living in Shanghai travels to Beijing. For both hometown and current city, the top 3 POI categories ordered by the visit number of this user are provided, as well as the specific number and proportion of visits corresponding to each category. It can be clearly observed that this user visits the highest number of theater POIs in both two cities, indicating that this interest is transferred from hometown to current city. However, the rest two categories change from school and attraction into housing and park, which demonstrate that this user also has the drifted interest varying among cities.

To verify above findings, we consider each Shanghai user and compare the proportion of his/her visits to $K$ POI categories in hometown and current city, where the $K$ categories are those ranked top $K$ by the visit number in hometown. Fig. 3(b) presents this comparison in terms of the average proportion values among all Shanghai users. By comparing results in current city (yellow) to those in hometown (blue), we observe that, for each user, the mostly visited POI categories in hometown do not count equally in current city. For example, top 5 categories in hometown count for near 70% of the visits on average, while the share of these categories reduces to about 30% in current city. Above observation again indicates the co-existence of user interest drift and transfer, which should be jointly considered when designing our proposed framework.

## 3.2 Different Characteristics of Popular POIs among Locals and Travelers

To investigate whether locals and travelers have different user interests on POIs, we first compare POI popularity within these two groups of users in Fig. 4(a). From this figure, we can observe a large number of POIs located at the top-left or bottom-right corner, indicating these POIs are only popular in one group of users, either locals or travelers. Then, for better illustration, we set a popularity rank of 2000 as the threshold deciding whether the POI is popular or not, and divide the POIs into four groups based on the popularity in locals and travelers. By setting 2000 as the threshold, about 15% of POIs are classified as popular. More detailedly, popular POIs in locals' account for 70.9% of total check-ins, with an average records of 602 per POI, in sharp contrast to unpopular POIs (50 on average). Similarly, popular POIs among travelers contribute to 85.4% of total check-ins, with an




(a) local vesus traveler (b) local vesus traveler (c) all shanghai users

Fig. 4. (a) Popularity ranks of POIs in Shanghai, in terms of locals and travelers, respectively. (b) Empirical distribution of the difference between rank in locals and rank in travelers. Rank of each POI is illustrated in (a). (c) Proportion of POIs belonging to different POI categories in each POI group. The group is divided in terms of the popularity ranks of a POI in both locals and travelers.

even larger gap in terms of average number (112 v.s. 4). Without loss of generality, we adopt this fixed number of 2000 as the threshold. To further represent the difference between the POI ranks in locals and travelers, we illustrate the empirical distribution of this relative rank in Fig. 4(b). It can be easily observed that most POIs have similar ranks in locals and travelers, while the ratio of "unpop-pop" and "pop-unpop" POIs cannot be overlooked either. The ratios of POIs with large positive value ($> 5000$) and negative value ($< -5000$) are about 4.5% and 7.6%, respectively. In Fig. 4(c) we investigate the proportion of POIs belonging to different categories in each POI group. The differences of proportions among groups indicate different user interests between locals and travelers, especially for "unpop-pop" group and "pop-unpop" group, *i.e.*, POIs popular in travelers only and POIs popular in locals only. Compared with "pop-unpop" group, "unpop-pop" group has more food, attraction and hotel POIs, which is reasonable for travelers. Considering the different composition of popular POIs among locals and travelers, we are motivated to characterize a POI with different representations for locals and travelers.

## 4 POI RECOMMENDATION FRAMEWORK WITH USER INTEREST DRIFT AND TRANSFER

In this paper, we build our cross-city POI recommendation framework PR-UIDT by leveraging the widely-used MF techniques [12, 13, 15, 26, 29], where both user and POI are mapped into latent low-dimensional spaces. Therefore, the core part of this framework is the user and POI latent vectors that characterises the user interest towards POIs and the features of POIs, respectively. Motivated by previous analysis, we first propose a novel design of user and POI latent vectors by considering both user interest drift and transfer. Then we integrate these design into a square error based MF model and a ranking error based MF model, respectively. Finally, we discuss the extension of PR-UIDT for locals and multi-city scenarios.

In the following, we represent matrices, vectors, and scalars as bold capital letters (*e.g.*, $\mathbf{X}$), bold lowercase letters (*e.g.*, $\mathbf{x}$), and normal lowercase letters (*e.g.*, $x$), respectively. If not otherwise specified, all vectors are in a column form. $\mathbf{X}^{\mathrm{T}}$ denotes the transpose of $\mathbf{X}$. We denote user latent factor matrix as $\mathbf{P}^{M\times F}$, and POI matrix as $\mathbf{Q}^{N\times F}$, where $M, N$ is the number of users and POIs, and $F$ is the vector dimension. Preference score of user $u$ on POI $i$ is denoted as $r_{ui}$. For readability, major notations used throughout this paper are listed in Table 2.

### 4.1 Framework Design

*4.1.1 Problem Definition.* Our proposed POI recommendation framework targets a practical scenario where the content provider is required to recommend high quality POIs to those cross-city travelers after they visit a

Table 2. List of commonly used notations.

| Notation | Description |
|---|---|
| $M, N, R$ | The number of users, POIs and records |
| $F, K$ | The number of user vector factors and inherent factors |
| $\mathcal{U}, \mathcal{U}_t, \mathcal{U}_l$ | The sets of all users, travelers and locals in current city |
| $\mathcal{V}, \mathcal{V}_c, \mathcal{V}_h$ | The sets of all POIs, POIs in current city and ones in hometown city |
| $\mathbf{P}, \mathbf{Q}$ | The latent factor matrix for users and POIs |
| $\mathbf{p}_u^0, \mathbf{p}_u^c, \mathbf{p}_u^h$ | User latent vector for inherent, current-city-related and hometown-city-related preference |
| $\mathbf{q}_i^t, \mathbf{q}_i^l$ | POI latent vector for traveler and local preference |
| $\hat{r}_{ui}$ | Prediction score of user $u$ over POI $i$ |
| $\alpha, \beta$ | Hyper parameter |
| $\lambda$ | Regularization parameter |

small number of locations in current city. Different from cold-start recommendation, this problem requires the consideration of user interest drift among hometown and current city, *i.e.*, the traveler visits data in both two cities. Here, we briefly define the problem as follows: given user $u$'s check-in records in both hometown and current city, *i.e.*, $R_u^h$ and $R_u^c$, respectively, the recommender needs to predict which POI in current city will be visited by $u$.

*4.1.2 User Vectors.* As we observed in both single user example and overall data analysis (Fig. 3), both user interest drift and transfer exists when users travel from hometown to a new city. In traditional MF models, each user is represented as a latent vector containing all the preference information of this user no matter where the user is, which is not sufficient in our case considering that the user interests between hometown and current city should only overlap to some extent. Therefore, we split the user vector $\mathbf{p}_u$ into a city-independent part, denoted as $\mathbf{p}_u^0$, and another city-dependent part, denoted as $\mathbf{p}_u^h$ in hometown and $\mathbf{p}_u^c$ in current city, which correspond to user's inherent interest and drifted interest, respectively. To learn the inherent interest that is consistent among different cities, $\mathbf{p}_u^0$ is trained with help of check-in records in both hometown and current city. Since hometown records are much denser than those in current city, $\mathbf{p}_u^0$ is able to achieve user interest transfer from hometown to current city. On the contrary, city-dependent user vector, *i.e.*, $\mathbf{p}_u^h$ or $\mathbf{p}_u^c$ is only learned from check-in records in hometown and current city, respectively, which represent the user interest drift between cities. With above two separated vector representations of user interests, we have

$$\mathbf{p}_u = [p_{u1}^h, p_{u2}^h, \cdots p_{uF}^h]. \tag{1}$$

for user $u$ in hometown and

$$\mathbf{p}_u = [p_{u1}^c, p_{u2}^c, \cdots p_{uF}^c] \tag{2}$$

for $u$ in current city, respectively. The first $K$ dimensions of embedding vectors of travelers should be the same between current and hometown city as shown in

$$p_{uk}^h = p_{uk}^c = p_{uk}^0, \forall 1 \leq k \leq K, u \in \mathcal{U}_t, \tag{3}$$

since they represent inherent preference of users which is irrelevant to cities.

*4.1.3 POI Vectors.* When investigating the phenomenon of user interest drift in terms of the POIs, we observed that locals' visiting behaviors, *i.e.*, the user interest in the hometown, and travelers' visiting behaviors, *i.e.*, the user interest in the current city, are different in the same city (Fig. 4). This motivates us to consider the differences between locals and travelers when designing POI vectors. However, as we already model the user vector with

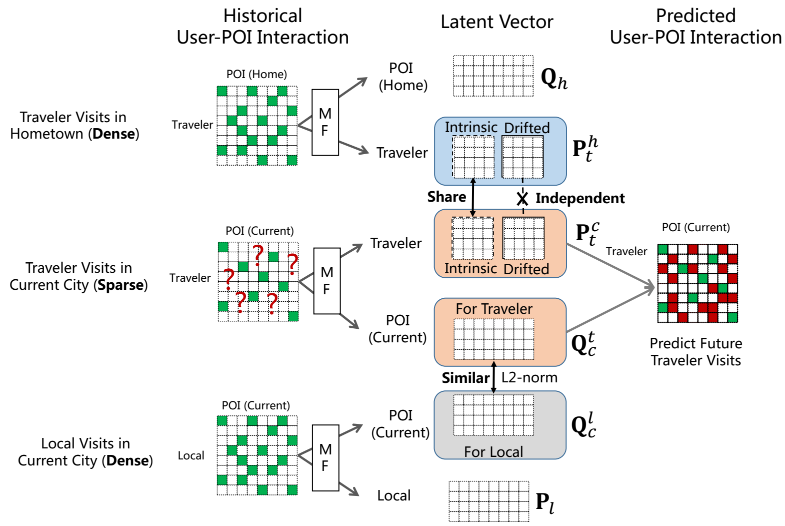



Fig. 5. Framework of PR-UIDT.

two separated parts, *i.e.*, $\mathbf{p}_u^0$ and $\{\mathbf{p}_u^h, \mathbf{p}_u^c\}$, the user-POI interaction function defined by MF technique implicitly divides POI vector into two corresponding parts similarly, *i.e.*, $\mathbf{q}_i^0$ and $\mathbf{q}_i'$ that only interact with $\mathbf{p}_u^0$ and $\{\mathbf{p}_u^h, \mathbf{p}_u^c\}$, respectively. Therefore, instead of separating POI vectors into two parts, we use two independent vectors so as to model the different interests among two user groups, *i.e.*, $\mathbf{q}_i^l$ for locals and $\mathbf{q}_i^t$ for travelers, which further demonstrates the user interest drift between hometown and current city. At the same time, travelers have limited check-in records in the current city, making it hard to learn a high quality POI vector. Therefore, besides the design of two independent vectors for the same POI that corresponds to interest drift, we further propose to enhance the learning of POI vectors designed for travelers by transferring the information from those designed for locals. More specifically, a $l^2$-norm constraint on distance between $\mathbf{q}_i^l$ and $\mathbf{q}_i^t$ is considered, which can be formulated as

$$d(\mathbf{q}_i^l, \mathbf{q}_i^t) = \sum_{k=1}^{F} (q_{ik}^l - q_{ik}^t)^2 \tag{4}$$

Consistent with user vectors, $\mathbf{q}_i^l$ is trained only with locals' records and $\mathbf{q}_i^t$ with travelers' ones.

*4.1.4 Overall Framework.* In this paper, we propose a cross-city POI recommendation framework that combines both user interest drift and interest transfer between hometown and current city. In order to learn from both two cities, besides travelers' check-in records in current city, their previous records in hometown as well as the records of other locals are also leveraged. More specifically, the sum of prediction errors on above three parts of user-POI interactions are all considered in the loss function. Here we denote the prediction errors on interactions between travelers and POIs in current city as $L_1$, those between these travelers and POIs in their hometown as $L_2$ and those between other locals and POIs in current city as $L_3$, respectively. For $L_1$ and $L_2$, the set of users is the same, though they play different roles, *i.e.*, travelers in current city and locals in hometown. According to our above designs of user vectors, for a user $u$, the corresponding user vector $\mathbf{p}_u$ is defined as (2) and (1) when used in $L_1$ and $L_2$, respectively. As for POI vectors, in $L_2$ and $L_3$ that relate to the same set of POIs $\{i\}$ in current city, two independent sets of POI vectors, *i.e.*, $\mathbf{q}_i^t$ and $\mathbf{q}_i^l$, are used, respectively. Moreover, a $l^2$-norm constraint on distance between $\mathbf{q}_i^t$ and $\mathbf{q}_i^l$ is integrated into the loss function. Mathematically, the final loss function is formulated as

$$L_{PR} = \min_{\mathbf{P}_t^c, \mathbf{Q}_c^t, \mathbf{P}_t^h, \mathbf{Q}_h, \mathbf{P}_l, \mathbf{Q}_c^l} L_1(\mathbf{P}_t^c, \mathbf{Q}_c^t) + \alpha L_2(\mathbf{P}_t^h, \mathbf{Q}_h) + L_3(\mathbf{P}_l, \mathbf{Q}_c^l) + \beta\big(||\mathbf{Q}_c^t - \mathbf{Q}_c^l||_F^2\big), \tag{5}$$

where $\alpha$ controls the extent to which we transfer hometown records to current city for a single user, and $\beta$ represents similarity constraint on the same POI.

The overview of our proposed PR-UIDT framework is illustrated in Fig. 5. The left three matrixes are historical records, which turn into latent vectors by matrix factorization. User vector is composed of intrinsic and drifted parts, and POI vector is separate for traveler and local. Then by multiplying traveler vector and current POI vector, we get predict matrix where all unrecorded user and POI pairs will have a preference score, based on which a recommender can build. With the abundant user interest knowledge learned from dense records of traveler check-in in hometown and local check-in in current city, PR-UIDT is able to achieve more accurate predictions on travelers' future check-ins in current city. We develop two different types of models which use different loss functions for $\{L_1, L_2, L_3\}$, *i.e.*, the square error based and the ranking error based loss functions, and will be described in the next two sections, respectively.

## 4.2 The Square Error based Model

Next, we present the Square error based Matrix Factorization model based on PR-UIDT framework (SMF-UIDT) and its optimization method.

*4.2.1 SMF-UIDT.* To learn user/item latent vectors, we introduce a square error based weighted regression function, which assigns a zero $r_{ui}$ value to missing user-POI entries with a confidence variable. Mathematically, it has a general form as follows:

$$L_{SMF}(\mathbf{P}, \mathbf{Q}) = \sum_{u \in \mathcal{U}, i \in \mathcal{V}} \omega_{ui}(\hat{r}_{ui} - r_{ui})^2 + \lambda\big(||\mathbf{P}||_F^2 + ||\mathbf{Q}||_F^2\big), \tag{6}$$

where $r_{ui}$ is the indicator function of whether user $u$ has visited POI $i$ and $\omega_{ui}$ is the weight function. In accordance with previous practice, we set $r_{ui} = 1, \omega_{ui} = 1$ when user $u$ has visited POI $i$, $r_{ui} = 0, \omega_{ui} = \omega_0$ otherwise. $\hat{r}_{ui}$ is the predicting score, which is defined as $\hat{r}_{ui} = \mathbf{p}_u^{\mathrm{T}} \cdot \mathbf{q}_i$. Corresponding to different types of user-POI interactions in $\{L_1, L_2, L_3\}$ depending on whether a user is a traveler or a local, $\hat{r}_{ui}$ can be further reformulated as

$$\hat{r}_{ui} = \begin{cases} [\ \mathbf{p}_u^0\ ;\ \mathbf{p}_u^c\ ]^{\mathrm{T}} \cdot \mathbf{q}_i^t & u \in \mathcal{U}_t, i \in \mathcal{V}_c \\ [\ \mathbf{p}_u^0\ ;\ \mathbf{p}_u^h\ ]^{\mathrm{T}} \cdot \mathbf{q}_i & u \in \mathcal{U}_t, i \in \mathcal{V}_h \\ \mathbf{p}_u^{\mathrm{T}} \cdot \mathbf{q}_i^l & u \in \mathcal{U}_l, i \in \mathcal{V}_c \end{cases}. \tag{7}$$

After integrating into the PR-UIDT framework, the final opjective In SMF-UIDT model, based on the (6) and (7), the matrices $\{\mathbf{P}_t^c, \mathbf{Q}_c^t, \mathbf{P}_t^h, \mathbf{Q}_h, \mathbf{P}_l, \mathbf{Q}_c^l\}$ are learned by minimizing the following regularized optimization problem:

$$\begin{aligned} L = &\min_{\mathbf{P}_t^c, \mathbf{Q}_c^t, \mathbf{P}_t^h, \mathbf{Q}_h, \mathbf{P}_l, \mathbf{Q}_c^l} L_1(\mathbf{P}_t^c, \mathbf{Q}_c^t) + \alpha L_2(\mathbf{P}_t^h, \mathbf{Q}_h) + L_3(\mathbf{P}_l, \mathbf{Q}_c^l) + \beta\big(||\mathbf{Q}_c^t - \mathbf{Q}_c^l||_F^2\big), \\ \text{where } L_1(\mathbf{P}_t^c, \mathbf{Q}_c^t) &= \sum_{u \in \mathcal{U}_t, i \in \mathcal{V}_c} \omega_{ui}(\hat{r}_{ui} - r_{ui})^2 + \lambda\big(||\mathbf{P}_t^c||_F^2 + ||\mathbf{Q}_c^t||_F^2\big), \\ L_2(\mathbf{P}_t^h, \mathbf{Q}_h) &= \sum_{u \in \mathcal{U}_t, i \in \mathcal{V}_h} \omega_{ui}(\hat{r}_{ui} - r_{ui})^2 + \lambda\big(||\mathbf{P}_t^h||_F^2 + ||\mathbf{Q}_h||_F^2\big), \\ L_3(\mathbf{P}_l, \mathbf{Q}_c^l) &= \sum_{u \in \mathcal{U}_l, i \in \mathcal{V}_c} \omega_{ui}(\hat{r}_{ui} - r_{ui})^2 + \lambda\big(||\mathbf{P}_l||_F^2 + ||\mathbf{Q}_c^l||_F^2\big). \end{aligned} \tag{8}$$

*4.2.2 The Parameter Estimation.* For SMF-UIDT, we minimize the loss function in (8) by element-wisely updating user vectors and POI vectors alternatively [5, 12]. In each iteration, the optimized value of $p_{uf}$ and $q_{if}$ is determined by solving $\frac{\partial L}{\partial p_{uf}} = 0$ or $\frac{\partial L}{\partial q_{if}} = 0$. The basic expression of update function can be found in (9), where $\hat{r}^f_{ui} = \hat{r}_{ui} - p_{uf} \cdot q_{if}$ represents the predicting score without dimension $f$. For a certain $p_{uf}$, the update function is

$$p_{uf} = \frac{\sum_i (r_{ui} - \hat{r}^f_{ui})\omega_{ui} q_{if}}{\sum_i \omega_{ui} q^2_{if} + \lambda}. \tag{9}$$

Depending on whether $u$ is a traveler or a local, and whether $f$ belongs to $\mathbf{p}^0_u$ representing the inherent interest or $\{\mathbf{p}^h_u, \mathbf{p}^c_u\}$ representing the drifted interest, a detailed update function is as follows:

$$\begin{cases} p^0_{uf} = \frac{\sum_{i\in\mathcal{V}_c} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} q^t_{if} + \alpha \sum_{i\in\mathcal{V}_h} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} q_{if}}{\sum_{i\in\mathcal{V}_c} \omega_{ui} q^{t2}_{if} + \alpha \sum_{i\in\mathcal{V}_h} \omega_{ui} q^2_{if} + \lambda} & 1 \le f \le K, u \in \mathcal{U}_t \\ p^c_{uf} = \frac{\sum_{i\in\mathcal{V}_c} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} q^t_{if}}{\sum_{i\in\mathcal{V}_c} \omega_{ui} q^{t2}_{if} + \lambda} & K \le f \le F, u \in \mathcal{U}_t, \hat{r}_{ui} = \mathbf{p}^{c\,\mathrm{T}}_u \cdot \mathbf{q}^t_i \\ p^h_{uf} = \frac{\sum_{i\in\mathcal{V}_h} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} q_{if}}{\sum_{i\in\mathcal{V}_h} \omega_{ui} q^2_{if} + \lambda} & K \le f \le F, u \in \mathcal{U}_t, \hat{r}_{ui} = \mathbf{p}^{h\,\mathrm{T}}_u \cdot \mathbf{q}_i \\ p_{uf} = \frac{\sum_{i\in\mathcal{V}_c} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} q^l_{if}}{\sum_{i\in\mathcal{V}_c} \omega_{ui} q^{l2}_{if} + \lambda} & 1 \le f \le F, u \in \mathcal{U}_l \end{cases}. \tag{10}$$

Similarly, POI vector is updated, according to whether this vector is used for locals or travelers, by (11),

$$\begin{cases} q^t_{if} = \frac{\sum_{u\in\mathcal{U}_t} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} p^c_{uf} + \beta q^l_{if}}{\sum_{u\in\mathcal{U}_t} \omega_{ui} p^{c2}_{uf} + \lambda + \beta} & 1 \le f \le F, i \in \mathcal{V}_c, \hat{r}_{ui} = \mathbf{p}^{c\,\mathrm{T}}_u \cdot \mathbf{q}^l_i \\ q^l_{if} = \frac{\sum_{u\in\mathcal{U}_l} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} p_{uf} + \beta q^t_{if}}{\sum_{u\in\mathcal{U}_l} \omega_{ui} p^2_{uf} + \lambda + \beta} & 1 \le f \le F, i \in \mathcal{V}_c, \hat{r}_{ui} = \mathbf{p}^{\mathrm{T}}_u \cdot \mathbf{q}^l_i \\ q_{if} = \frac{\sum_{u\in\mathcal{U}_t} (r_{ui}-\hat{r}^f_{ui})\omega_{ui} p^h_{uf}}{\sum_{u\in\mathcal{U}_t} \omega_{ui} p^{h2}_{uf} + \lambda} & 1 \le f \le F, i \in \mathcal{V}_h \end{cases}. \tag{11}$$

In each iteration, the $f^{\text{th}}$ latent factor of $u$'s vector, namely $p_{uf}$, is updated in sequence. After computing all the user factors, POI factors can be similarly updated. Overall, one iteration takes $O((M+N)K^2 + RK)$ time, which only depends on the number of observed interactions. Due to space limit, we leave out the detailed algorithm.

## 4.3 The Ranking Error based Model

Besides square error based model, we also propose a Ranking error based Matrix Factorization model based on PR-UIDT framework (RMF-UIDT).

*4.3.1 RMF-UIDT.* In POI dataset, we only have a user's check-in record and an unvisited POI does not necessarily indicate the user dislikes it. The unobserved data actually is a mixture of negative preference for POIs and missing values. This motivates us to consider a ranking error based loss function for modeling the ranking order of user's preference for observed POIs and unobserved POIs. By assuming that the user prefers an observed POIs over all other unobserved POIs, we introduce following pairwise loss function:

$$L_{RMF}(D,P,Q) = -\sum_{(u,i,j)\in D} \log \sigma(\hat{x}_{uij}) + \lambda(||\mathbf{P}||^2_F + ||\mathbf{Q}||^2_F), \tag{12}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ is the logistic sigmoid function. $(u,i,j)$ is the updating triple and $D$ represents universal set of all possible triples. $\hat{x}_{uij} = \hat{r}_{ui} - \hat{r}_{uj}$ should be as large as possible because of the intuition that visited POIs should be ranked before unvisited ones.

Applying PR-UIDT framework to (12), RMF-UIDT is to minimize following loss function:

$$
\begin{aligned}
L = &\min_{\mathbf{P}_t^c,\mathbf{Q}_c^t,\mathbf{P}_t^h,\mathbf{Q}_h,\mathbf{P}_l,\mathbf{Q}_c^l} L_1(\mathbf{P}_t^c,\mathbf{Q}_c^t) + \alpha L_2(\mathbf{P}_t^h,\mathbf{Q}_h) + L_3(\mathbf{P}_l,\mathbf{Q}_c^l) + \beta\big(||\mathbf{Q}_c^t - \mathbf{Q}_c^l||_F^2\big),\\
\text{where } &L_1(\mathbf{P}_t^c,\mathbf{Q}_c^t) + L_3(\mathbf{P}_l,\mathbf{Q}_c^l) = -\sum_{(u,i,j)\in D*} \log\sigma(\hat{x}_{uij}) + \lambda\big(||\mathbf{P}_t^c||_F^2 + ||\mathbf{Q}_c^t||_F^2 + ||\mathbf{P}_l||_F^2 + ||\mathbf{Q}_c^l||_F^2\big),\\
&L_2(\mathbf{P}_t^h,\mathbf{Q}_h) = -\sum_{(u,i,j)\in D**} \log\sigma(\hat{x}_{uij}) + \lambda\big(||\mathbf{P}_t^h||_F^2 + ||\mathbf{Q}_h||_F^2\big).
\end{aligned}
\tag{13}
$$

Note that $D^*$ and $D^{**}$ are the triple sets corresponding to POI visits in current city and hometown, respectively, which is defined as:

$$
\begin{aligned}
D^* &= \{(u,i,j) \mid r_{ui} = 1 \cap r_{uj} = 0 \cap u \in \mathcal{U} \cap (i,j) \in \mathcal{V}_c\},\\
D^{**} &= \{(u,i,j) \mid r_{ui} = 1 \cap r_{uj} = 0 \cap u \in \mathcal{U}_t \cap (i,j) \in \mathcal{V}_h\}.
\end{aligned}
\tag{14}
$$

*4.3.2 The Parameter Estimation.* Stochastic gradient descent (SGD) is widely used to update above functions. The updating formula for user or POI vectors among $\{\mathbf{P}_t^c,\mathbf{Q}_c^t,\mathbf{P}_t^h,\mathbf{Q}_h,\mathbf{P}_l,\mathbf{Q}_c^l\}$ can be derived via computing their corresponding gradients according to (13). As for the training process, the basic idea is to sample training instances from $D^*$ and $D^{**}$ and then update parameters iteratively, with time complexity of $O(RK)$.

## 4.4 Extension

*4.4.1 POI Recommendation for Locals.* Though our proposed PR-UIDT framework focuses on improving recommendation performance for travelers, it can simultaneously make recommendation for locals without any additional module. More specifically, for a local user $u$, the predication score $\hat{r}_{ui}$ for visiting a local POI $i$ is calculated by multiplying user vector $[\mathbf{p}_u^0;\mathbf{p}_u^h]$ and POI vector $\mathbf{q}_i^l$ together. Note that $\mathbf{p}_u^h$ refers to $u$'s unique interest in hometown and $\mathbf{q}_i^l$ is designed for $i$'s local visitors. Both $[\mathbf{p}_u^0;\mathbf{p}_u^h]$ and $\mathbf{q}_i^l$ are trained via $L_3$-term in (5). Since the learning of POI vectors are enhanced by transferring the information between those designed for travelers, *i.e.*, $\mathbf{q}_i^t$, and those designed for locals, *i.e.*, $\mathbf{q}_i^l$, $\mathbf{q}_i^l$ should embed more powerfully information than a POI vector that is only trained with hometown check-ins, implying that PR-UIDT is likely to perform better than a recommender designed solely for locals. Therefore, the proposed PR-UIDT framework has the ability to serve both cross-city travelers and locals, while in this paper we focus on the former and do not evaluate the recommendation performance for locals.

*4.4.2 Cross-city POI Recommendation in Multi-city scenarios.* Users may travel among multiple cities instead of just two cities. Next we will demonstrate how our proposed PR-UIDT framework can be extended into multi-city scenarios.

As illustrated in Fig. 6, user latent vector is still separated into two parts corresponding to inherent interest and drifted interest, respectively. The inherent interest is assumed to be invariant among cities, while the drifted interest is city-dependent. However, assigning an independent vector to user's drifted interest for each city brings $n$ times of memory usage, where $n$ is the number of cities, making it impossible for a nationwide location-based service like Wechat. Therefore, in our design, the drifted interest only depends on the role user played in each city, *i.e.*, locals or travelers, denoted as $\mathbf{P}^h$ and $\mathbf{P}^t$ respectively, rather than on the city itself. Corresponding to users' roles, the same POI has different latent vectors, $\mathbf{Q}^h$ for locals and $\mathbf{Q}^t$ for travelers, with a $l^2$-norm constraint as well. Similar to the objective function we defined in (5) for two-city scenario, here the PR-UIDT is optimized

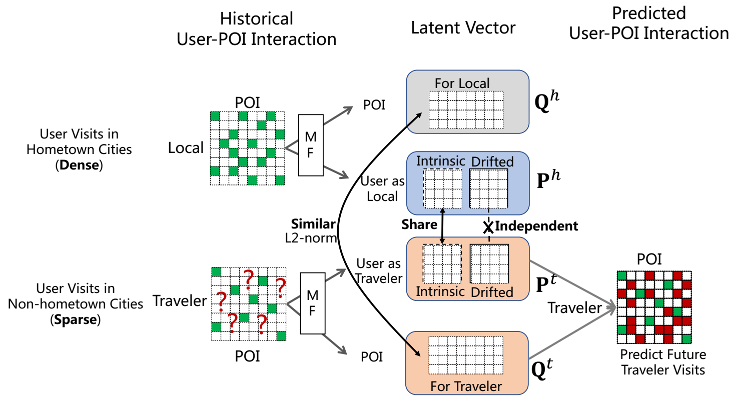



Fig. 6. Framework of PR-UIDT in Multi-city Scenarios.

based on follows:

$$L_{PR} = \min_{\mathbf{P}^h, \mathbf{Q}^h, \mathbf{P}^t, \mathbf{Q}^t} L_{\text{local}}(\mathbf{P}^h, \mathbf{Q}^h) + \alpha L_{\text{traveler}}(\mathbf{P}^t, \mathbf{Q}^t) + \beta\big(||\mathbf{Q}^t - \mathbf{Q}^h||_F^2\big), \tag{15}$$

where the former $\{L_2, L_3\}$ related to the prediction errors on locals' check-ins are merged into $L_{\text{local}}$, while the former $L_1$ is extended into $L_{\text{traveler}}$ that considers travelers' check-ins in multiple cities instead of one fixed current city. By this means, the extended PR-UIDT is able to recommend future check-ins for both locals and travelers among multiple cities. Moreover, compared to the preliminary version for *hometown-current city* setting, it does not increase the memory usage for storing user and POI vectors as long as the numbers of users and POIs are fixed. The overall time complexity is still linear to the size of check-in data and independent of the number of cities.

Based on above framework, the user interest transfer is enabled not only between hometown and current city, but also among multiple non-home cities. First, the user inherent vector is learned from all check-ins of a user. Second, as a user plays the same role of traveler in all non-home cities, the user drifted interest part is jointly learned from check-ins in these cities. To be more practicable, the recommender system can label several cities as the hometowns of a user if she frequently check-ins these cities. Also, PR-UIDT allows a user to have no hometown if she never check-ins in the hometown.

*4.4.3 Discussion.* Our proposed PR-UIDT framework considers user interest drift and transfer among different cities, *i.e.*, in spatial domain. As users' check-in behaviors are generally affected by both spatial and temporal effects that cannot be decoupled, the PR-UIDT can also be extended so as to consider these two simultaneously. More specifically, some time-specific interest can be transferred from one city to another, while the others may drift in different cities. For example, a user may go to restaurant at dinnertime regardless of the visited city. However, the other parts of her daily routine may change a lot if she is on vacation. To distinguish this, a direct solution can follow the similar idea of MF and model the context of users' check-ins with the latent factors. Based on these learnable factors, user interest related to a specific context can transfer or drift among different cities. In above example, such context is the pair of POI type and check-in time. Mathematically, we denote the context vector as $\mathbf{s}_c$, then the overall prediction score $\hat{r}_{uic}$ for user $u$ visiting POI $i$ in context $c$ is calculated as $\hat{r}_{uic} = \mathbf{p}_u^{\mathrm{T}} \cdot \mathbf{q}_i + \mathbf{p}_u^{\mathrm{T}} \cdot \mathbf{s}_c$. A more advanced solution is to combine with factorization machine based models [28].

Furthermore, in order to model user interest more precisely, we can consider separate users in more fine-grained groups based on ages or gender. In general, these fine-grained user groups can be helpful in POI recommendation, since extra information and detailed user profiles contribute to uncovering similarity and dissimilarity of user groups. However, for cross-city POI recommendation, these information have fewer impact compared with the role of users (local or traveler) and they are not easily accessible. Thus we do not consider it in PR-UIDT framework.

## 5 EXPERIMENTS

To evaluate the performance of our proposed cross-city POI recommendation framework, *i.e.*, PR-UIDT including two models SMF-UIDT and RMF-UIDT, we conduct a series of experiments on three real-world datasets to answer the following five key research questions:

- **RQ1**: How do SMF-UIDT and RMF-UIDT perform compared with other state-of-the-art models?
- **RQ2**: What are the effects of characterizing user interest drift and transfer in PR-UIDT?
- **RQ3**: How do hyper-parameters (i.e., $\alpha$, $\beta$) impact the performance of proposed models?
- **RQ4**: Is the PR-UIDT framework practical enough for real-world POI recommendation applications?
- **RQ5**: Is this proposed PR-UIDT framework general enough for multi-city POI recommendation scenarios?

### 5.1 Experiment Settings

Since each dataset contains a pair of two cities, we can make recommendation for travelers in each city respectively, thus two-way experiments (six recommendation tasks) are conducted as shown below.

- **Tencent-BS (SB)**: Recommend POIs in Shanghai (Beijing) to travelers from Beijing (Shanghai).
- **Tencent-BT (TB)**: Recommend POIs in Tianjin (Beijing) to travelers from Beijing (Tianjin).
- **Yelp-LP (PL)**: Recommend POIs in Phoenix (Las Vegas) to travelers from Las Vegas (Phoenix).

**Evaluation methodology**. For evaluation, we divide the cross-city travelers' records in current city based on 80/20 principle to generate training and testing sets. Note that the travelers' records in hometown and locals' records in current city are considered as training data. To avoid information leaking, we also remove a part of each traveler's records in hometown from the training data if these records are temporally after the earliest record in the testing data of this user. For evaluation measures, we predict the top-$k$ ranking among all non-visited POIs for each user and employ two metrics including *Hit Ratio* (HR) and *Normalized Discounted Cumulative Gain* (NDCG). Mathematically, with testing set $\mathcal{T}$, the $HR@k$ is defined as:

$$HR_u@k = \frac{\sum_{(u,i)\in\mathcal{T}} HR_{u,i}@k}{|\mathcal{T}|}, \text{ where } HR_{u,i}@K = \begin{cases} 1\,, & \text{hit in top-k recommendation} \\ 0\,, & \text{else} \end{cases}. \tag{16}$$

As for $NDCG@k$, considering each user $u$ with her own testing set $\mathcal{T}_u$, it is defined as the average of $NDCG_u@k$, which is calculated as:

$$NDCG_u@k = \frac{\sum_{p=1}^{k} \frac{2^{R(u,p)}-1}{\log(p+1)}}{\sum_{p=1}^{|\mathcal{T}_u|} \frac{1}{\log(p+1)}}, \tag{17}$$

where $R(u,p)$ is the rating assigned by $u$ to the POI at the $p^{th}$ position on the ranked list produced for $u$. Here $R(u,p)$ equals 1 if hit and 0 otherwise. Among these two metrics, HR is more relevant to the prediction recall, *i.e.*, measuring how many groundtruth POIs are recalled into the top-$k$ list, while NDCG is very sensitive to the ratings of the highest ranked POIs. Therefore, it is a common practice to evaluating both two metrics in top-$k$ recommendation tasks [11, 12, 33]

**Baselines**. We compare our proposed SMF-UIDT and RMF-UIDT models with five baselines, which can be divided into three groups based on whether the user interest drift or transfer is considered or not.

For the first group, we consider two state-of-the-art MF approaches that only leverage the check-in data in current city (named as vanilla).

- **WRMF** [12]. WRMF minimizes the square error loss by assigning both observed and unobserved check-ins with different confidential values based on MF. We use the implementation released by the authors[4].

- **BPR** [27]. BPR optimizes the MF model with a pairwise ranking loss and learns model parameters with Stochastic Gradient Descent (SGD) method. It is a highly competitive approach for item recommendation and we use the implementation released in [12].

Since the sequence characteristics among users' visiting behaviors may also benefit the prediction of future visits, we also investigate this temporal effect without considering interest drift.

- **SASRec** [14]. SASRec uses a Transformer language model to capture users' sequential behaviors, and achieves state-of-the-art performance on sequential recommendation.

Then we consider to transfer users' interest from hometown to current city so as to improve the cross-city POI recommendation performance.

- **LCE** [30]. LCE is a MF model that exploits the POI property (category) and user preference to solve cold-start problems. When leveraging the travelers' visiting behaviors in hometown, recommending POIs in current city becomes a cold-start POI problem. Thus in our cross-city POI recommendation problem we adapt the LCE model by learning user interest in hometown and transfer this information to help predicting their check-ins in current city. We use the implementation released by the authors[5].

Finally, we compare with a cross-city POI recommendation method that characterizing the user interest drift among different cities.

- **STLDA** [42]. STLDA is a state-of-the-art probabilistic generative model that learns region-dependent user preference based on the POI content, social relationship, temporal and spatial correlation information [6]. Since our proposed PR-UIDT framework does not take social relation and temporal effect into consideration, we implemented two versions, *i.e.*, a degenerative one that removes these latent variables (denotted as $\text{STLDA}_d$) and a complete one (denotted as $\text{STLDA}_c$), for fair comparisons. Note that social relationship is unavailable in Tencent data, thus we infer this information via an open-sourced algorithm[7] based on co-located check-ins [1].

**Parameter setting**. For above baselines, we explore hyper-parameters similarly as the original paper. More specifically, for WRMF, we tune the weight parameters of negative instances $\omega$ in [0.001, 0.005, 0.01, 0.05, 0.1, 0.5]. For SASRec, following the original paper, we set the historical sequence length as the average length of users' POI sequence in the dataset. The numbers of layers and attention heads are also carefully tuned. For LCE, the parameter that controls the relative weight of user-content (POI) matrix and user-POI matrix is tuned in [0.1, 0.2, ..., 0.8, 0.9]. For STLDA, the number of topics is tuned in [30, 40, ..., 70, 80], and region number in [10, 20, ..., 90, 100]. Other hyper-parameters of STLDA, prior hyper-parameters $\alpha, \beta, \gamma, \eta$ are set to the fixed value as proposed in [42]. For PR-UIDT framework that is implemented based on corresponding SMF and RMF models, the weight parameters of negative instances $\omega$ in RMF are set to the same value as WRMF. As for $\alpha$ and $\beta$ that decide the tradeoff of interest transfer and drift, we conduct grid search over these two parameters to find the peak performance, that is $\alpha$ in [0 0.1, 0.2, ..., 0.9, 1.0] and $\beta$ in [0.01, 0.1, 1, 10, 100, 1000]. Finally for learning rate and regularization term $\lambda$, we tuned them in range of [0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1]. Since the findings are consistent across the number of latent factors $F$, we report the results of $F = 32$ only.

Table 3. Performance comparison.

| Task | Metric@100 | Vanilla | | Sequence | Transfer | Drift | | Proposed | | Gain* |
|---|---|---|---|---|---|---|---|---|---|---|
| | | WRMF | BPR | SASRec | LCE | STLDA$_d$ | STLDA$_c$ | SMF-UIDT | RMF-UIDT | |
| Tencent-BS | NDCG | 0.2770 | 0.2831 | 0.3049 | 0.1799 | 0.2402 | 0.2436 | 0.3136 | **0.3139** | +3.0% |
| | HR | 0.5060 | 0.5360 | 0.5679 | 0.3994 | 0.3563 | 0.3612 | **0.6027** | 0.5589 | +6.1% |
| Tencent-SB | NDCG | 0.2264 | 0.1817 | 0.2348 | 0.1923 | 0.2245 | 0.1983 | 0.2715 | **0.2774** | +18.1% |
| | HR | 0.4094 | 0.2015 | 0.4458 | 0.3429 | 0.4952 | 0.4317 | **0.5127** | 0.4547 | +3.5% |
| Tencent-BT | NDCG | 0.1941 | 0.1673 | 0.1665 | 0.1357 | 0.1828 | 0.1869 | **0.2161** | 0.2153 | +11.3% |
| | HR | 0.3125 | 0.2193 | 0.3679 | 0.1964 | 0.4236 | **0.4396** | 0.3908 | 0.3539 | -11.1% |
| Tencent-TB | NDCG | 0.1722 | 0.1445 | 0.1277 | 0.0939 | 0.1793 | 0.1808 | **0.1815** | 0.1635 | +0.4% |
| | HR | 0.2555 | 0.2107 | 0.3087 | 0.1448 | 0.3095 | 0.3226 | **0.3316** | 0.2615 | +2.8% |
| Yelp-LP | NDCG | 0.1742 | 0.1601 | 0.1834 | 0.1439 | 0.1678 | 0.1695 | 0.1966 | **0.1984** | +8.2% |
| | HR | 0.2881 | 0.2450 | 0.3238 | 0.1948 | 0.3531 | **0.3902** | 0.3891 | 0.3780 | -0.3% |
| Yelp-PL | NDCG | 0.1572 | 0.1504 | 0.1510 | 0.1417 | 0.1661 | 0.1688 | **0.1860** | 0.1793 | +10.2% |
| | HR | 0.2433 | 0.2386 | 0.3045 | 0.1858 | 0.3134 | 0.3333 | **0.4017** | 0.3546 | +20.5% |

$*$: The performance gain is by the best one among SMF-UIDT and RMF-UIDT, compared with the best one among other baselines (underlined). The best results among all methods are bolded.

## 5.2 Performance Comparison (RQ1)

Detailed experiment results of all methods on NDCG@100 and HR@100 are presented in Table 3, where the reported metric value is obtained by calculating the average of results from 5 repeated experiments.

From above results, we have the following observations:

- Overall, our proposed SMF-UIDT and RMF-UIDT reach the best performance among all 6 tasks in terms of NDCG and 4 out of 6 tasks in terms of HR. More specifically, they achieve a significant improvement of 0.4%~20.5% compared with best performance of other state-of-the-art models, except for HR@100 in Tencent-BT and Yelp-LP tasks. Compared with first two vanilla MF methods that do not consider users' previous visiting behaviors in their hometown, *i.e.*, WRMF and BPR, SMF-UIDT has an improvement of 5.4%~65.1% and BPR-UIDT gets 4.3%~125.7% better. Such significant performance gain comes from both data augmentation and our proposed PR-UIDT framework that consider both user interest drift and transfer, which we will investigate detailedly in RQ2.
- With fine-grained modeling of region-dependent user interest and other auxiliary information like users' social relationship, STLDA$_c$ performs fairly good in terms of HR, being the best among all baselines. Moreover, in Tencent-BT task, STLDA$_c$ is 11.1% higher in HR than our proposed SMF-UIDT while 11.3% lower in NDCG. With the number of spatial region set to 100, compared to SMF-UIDT that only distinguishes between hometown and current city, STLDA$_d$ and STLDA$_c$ utilize more precise information of users' drifted interest, resulting in high performance in terms of HR. However,they do not achieve good performance in terms of NDCG that focuses more on the specific order among POIs, which is caused by following two reasons. First, they do not leverage latent vector to modeling user preference order among different POIs. Second, they do not consider the user interest transfer from hometown to current city. As the correct ranking order of recommended POIs is more valuable in practical applications, the better performance in

[4]https://github.com/hexiangnan/sigir16-eals
[5]https://github.com/msaveski/LCE
[6]We use the implementation released by the authors https://sites.google.com/site/dbhongzhi/.
[7]https://github.com/yangzhangalmo/walk2friends

NDCG proves the superiority of our proposed PR-UIDT, and also the insufficiency of leveraging interest drift only. Moreover, by comparing STLDA$_c$ with STLDA$_d$, we observe a performance improvement among 5 out of 6 tasks, indicating the value of leveraging social relations and validity of the social inference algorithm that we apply on Tencent data as well. This insightful knowledge will motivate our future extension on incorporating social relation for better recommendation quality.

- Surprisingly, LCE performs poor in almost every task. Although LCE model utilizes user interest transfer to solve clod-start POI problem, no drifted interest is specifically modelled. This finding implies that without sufficient utilization of travelers' visiting behavior in current city, recommender fails to predict users' future check-ins even with the transferred interest from hometown.
- Last but not least, we observe that the sequencial recommender, *i.e.*, SASRec, performs fairly competitive on Tencent-BS/SB and Yelp-LP. This demonstrate the positive effect brought by modeling sequential nature existed in users' POI visiting behaviors, which can be further integrated into our proposed PR-UIDT framework.

Practical recommender systems typically have two stages [37]: 1) candidate selection that selects hundreds of items that might be of interest to a user, and 2) ranking that re-ranks the candidates to show top a few results. If we want to apply the PR-UIDT in candidate selection stage that requires a high recall, evaluation with a large K of hundreds is suitable. As for the ranking stage, we should evaluate the performance with a small K (10∼20). Therefore, we further compare the performance *w.r.t.* HR@K ($K \in \{1, 2, 5, 10, 20, 50, 100\}$) in Fig. 7. Similar to the previous result, our proposed PR-UIDT outperforms other state-of-the-art baselines in most cases. STLDA$_c$ performs better in large $K$ cases on Tencent-BT, while it does not provide more accurate predictions in small $K$ cases, *i.e.*, not suitable for ranking stage. SASRec shows fairly competitive performance on Tecent-TB, and outperforms the PR-UIDT in small $K$ cases.

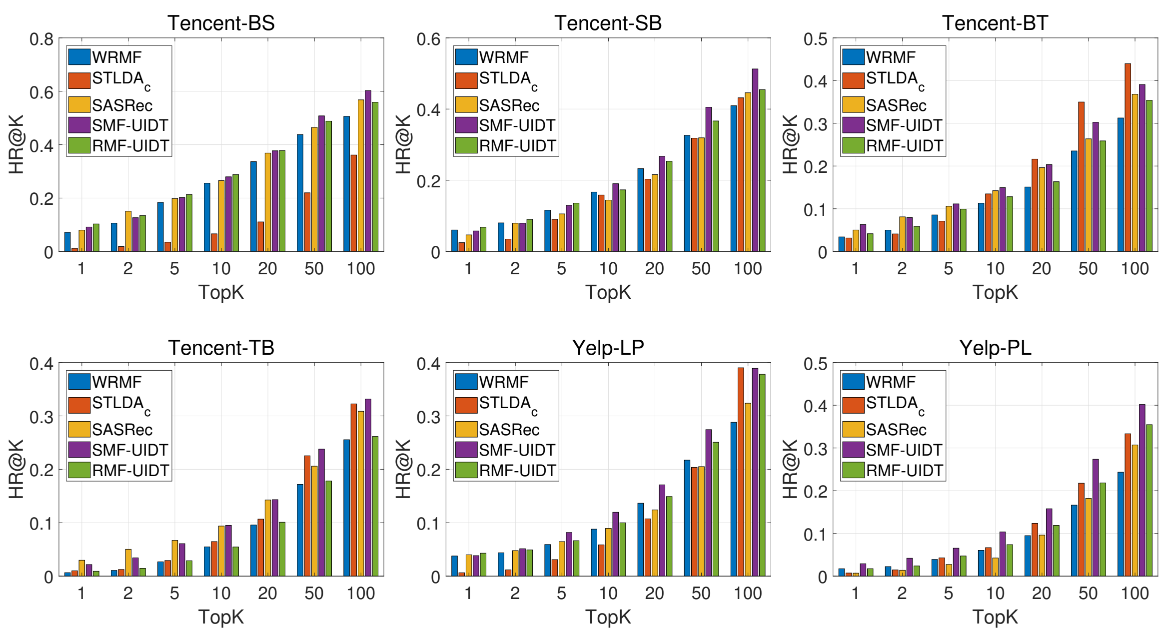



Fig. 7. Performance comparison in terms of HR evaluated by different values of $K$.

In a word, from above comparison, the PR-UIDT framework does outperform other state-of-the-art models, implying that both user interest drift and interest transfer need to be considered in cross-city POI recommendation. In the mean time, we observe the competitive performance of baselines after considering social relations or sequential behaviors, which motivates us to enhance PR-UIDT with social regularization [40] and transformer structure [14].

### 5.3 Studies of User Interest Drift and Transfer (RQ2)

We first conduct an ablation study to investigate whether user interest drift and transfer are necessary for cross-city POI recommendation. Then our qualitative analysis further reveal how PR-UIDT helps us to comprehend user check-in behaviors between different cities and generate more convincing recommendation.

*5.3.1 Ablation Study.* The performance comparison between PR-UIDT and its degenerative versions in terms of NDCG@100 is presented in Fig. 8, where "+User" and "+POI", these refer to leveraging the travelers' check-ins in hometown and locals' check-ins in current city, respectively, and "Without UIDT" refers to not using our design of user and POI vectors proposed in Sec. 4.1.

First we focus on the performance of SMF and RMF without UIDT when leveraging different user check-in data. It is an intuition that SMF and RMF should perform better after adding either travelers' check-in records in hometown or locals' check-in records in current city, and best when adding both. However, SMF degrades in Tencent-BT and Tencent-TB with "+User" (-9.8%, -3.6%) or "+POI" (-11.3%, -6.1%) separately, and only gets slight improvement after adding both information (+3.3%, +1.5%). This abnormal phenomenon, that more information about users and POIs leads to worse recommendation results, shows an inconsistence between users' visiting

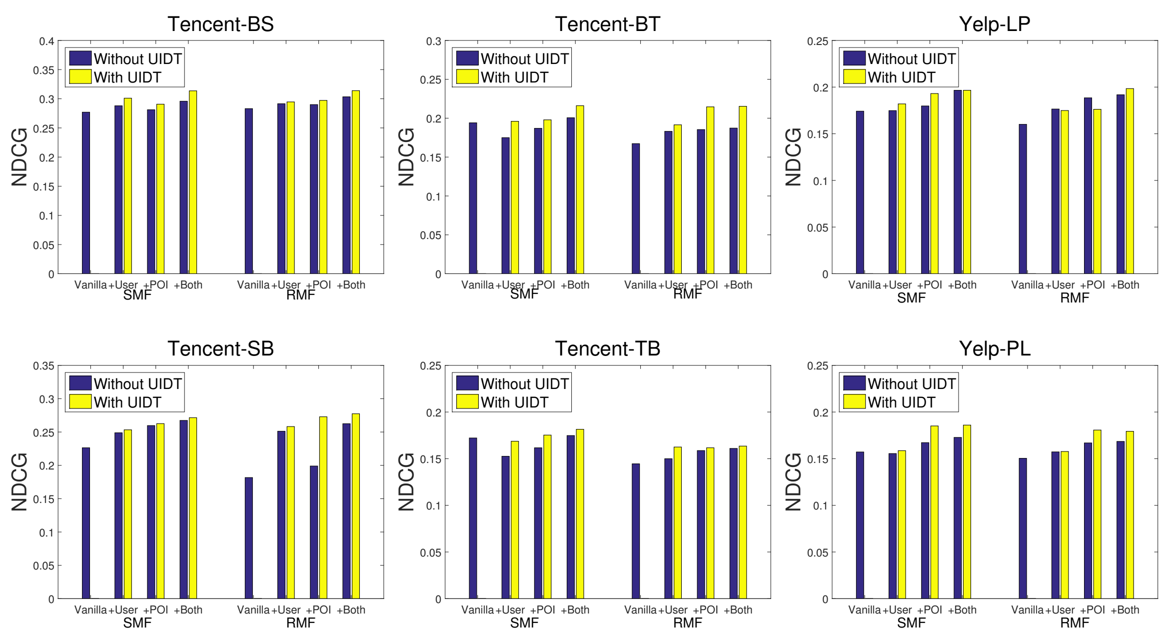



Fig. 8. Performance comparison between PR-UIDT and its degenerative versions, in terms of NDCG@100.

behaviors in current city and hometown, and also between those of travelers and locals, confirming our observation about the necessity of considering user interest drift in cross-city POI recommendation problem.

Then we compare the performance improvements on NDCG before and after applying PR-UIDT framework with the same training data. SMF-UIDT outperforms its degenerative version in every tasks, with average improvements of 5.8% with “+User”, 6.1% with “+POI” and 4.5% with “+Both”. Except for Yelp-LP, RMF-UIDT reaches average improvements of 3.4% with “+User”, 13.1% with “+POI” and 5.9% with “+Both” in the other five tasks. But in Yelp-LP task, it degrades with “+User” (-0.9%) and “+User” (-6.5%) separately, but outperforms with “+Both” (+3.4%). It is interesting that at the same time RMF-UIDT is able to achieve performance improvements in terms of HR@100 in all three cases (*ex.*, +6.8% with “+User” and +4.5% with “+POI”). Compared with those without UIDT, SMF-UIDT and RMF-UIDT perform better after adding either travelers’ check-in records in hometown or locals’ check-in records in current city, and best when adding both, implying that user interest drift can effectively solve the data augmentation problem, and our proposed PR-UIDT framework is capable of handling this complex challenge.

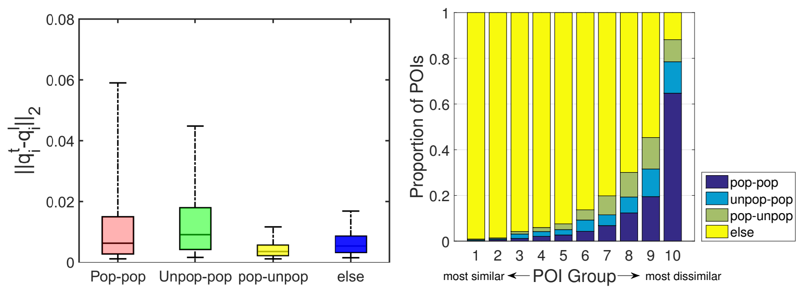
Above ablation study makes it clear that the performance improvements of SMF-UIDT and RMF-UIDT not only come from the data augmentation, but also the result of combining both user interest drift and transfer in model design.

*5.3.2 Qualitative Analysis.* To further reveal the plausibility of PR-UIDT in modeling user interest drift and transfer between different cities, our following analysis is divided into two parts, focusing on two designs of user and POI vectors, respectively.

**Individual user case study.** In PR-UIDT framework we represent each user vector with a city-independent part, representing the inherent user interest that is invariant from hometown to current city, and another city-dependent part, representing the drifted user interest drift in different cities. Therefore, we can recommend two lists of POI candidates, denoted as $l_i$ and $l_d$, to each user based on above two types of user interests, *i.e.*, two parts of user vectors, respectively. Here we choose the recommendation results of SMF-UIDT on Tencent-SB task, where POIs in Beijing are recommended to travelers from Shanghai. In order to investigate how SMF-UIDT make recommendation, we randomly sample a user #76474 that has groundtruth POIs hit in $l_i$ and $l_d$, respectively. Table 4 lists two of the testing POIs and the top 5 similar POIs in the training data, evaluated by cosine similarity of POI vectors, where the informations of POI name, POI-ID, category, located city (“B” for Beijing and “S” for Shanghai) and similarity value are all presented. As shown in this table, user #76474 has a complexed preference

Table 4. User case study.

| User-ID #76474 | POI | POI Similarity | City | POI-ID | Category |
|---|---|---|---|---|---|
| Test POI Hit in $l_i$ | Renaissance Beijing Capital Hotel | - | B | #772 | Hotel: Star Hotel |
| Top 5 Similar POIs in Train Data | Clove International Business Center | 0.762 | S | #5388 | Entertainment: General Shopping Mall |
| | TLScontact Centre | 0.721 | S | #3501 | Else: Living Services |
| | Chong Hing Financial Center | 0.690 | S | #18817 | Housing: commercial buildings |
| | Shanghai Longemont Hotel | 0.683 | S | #1472 | Hotel: Star Hotel |
| | Shangri-La Hotel | 0.635 | B | #726 | Hotel: Star Hotel |
| Test POI Hit in $l_d$ | National Aquatics Center | - | B | #1280 | Entertainment: Swimming Hall |
| Top 5 Similar POIs in Train Data | Capital International Airport | 0.958 | B | #55 | Transportation Facilities: Airport |
| | Olympic Park | 0.977 | B | #62 | Tourist attractions: national attractions |
| | Beijing South Railway Station | 0.965 | B | #561 | Transportation Facilities: Railway Station |
| | Wangfujing | 0.960 | B | #79 | Else: Hot spot area: business circle |
| | National Library | 0.741 | B | #577 | School&Culture: Library |




(a) POIs grouped by popularity (b) POIs grouped by vector distance

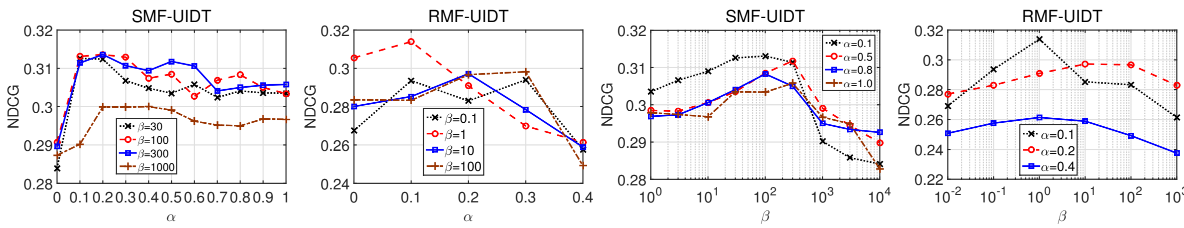
Fig. 9. The relationship between POI Popularity and POI vector distance $||q_i^t - q_i^l||_2$.

for Beijing POIs. In testing records, he visits a star hotel and National Aquatics Center. The former one is hit in $l_i$, indicating the user interest transferred from hometown, as this user has visited two star hotels in Shanghai. As for the later one, indeed National Aquatics Center looks more like a tourist attraction for travelers in Beijing, thus PR-UIDT make this recommendation based on the fact that this user has visited similar hot spots, like Olympic Park, Capital International Airport and Beijing South Railway Station, indicating the drifted user interest that largely depends on the city of Beijing. Above user case study shows how user interest drift and transfer work in cross-city POI recommendation.

**POIs as a whole.** In PR-UIDT framework, POI vectors are set different for locals and travelers, following the intuition that two user groups should have different interest. Meanwhile, $l^2$-norm for two vectors of the same POI is integrated to transfer POI features from that designed for locals to that designed for travelers. To demonstrate the effectiveness of above design, Fig. 9(a) shows the distance between two vectors, *i.e.*, $||q_i^t - q_i^l||_2$, in terms of the distribution quantiles (5%, 25%, 50%, 75%, 95%), where POIs are divided into four groups based on the popularity in locals and travelers (the same as we did in Sec. 3.2). From this figure, we observe the difference of $||q_i^t - q_i^l||_2$ among these four POI groups. When a POI is popular among travelers, *i.e.*, “pop-pop” and “unpop-pop ”groups, the POI vector distance tend to be larger, implying that less transferred information and more independent information are considered in learning this POI feature. Vice versa, when learning an unpopular POI among travelers, much information are transferred from locals' visiting behaviors to solve the sparsity issue. Moreover, we show the relationship between the POI popularity and POI vector distance by plotting the POI popularity distribution of different POI groups in Fig. 9(b), where POIs are equally divided into ten groups based on $||q_i^t - q_i^l||_2$, *i.e.*, the similarity between POI vector designed for locals and the other designed for tourists. It can be clearly observed that those with larger values of $||q_i^t - q_i^l||_2$ tend to be popular POIs. POI that is popular among either locals or travelers tend to have less similar vectors, revealing that PR-UIDT framework successfully capture user interest drift in different cities. The vectors of unpopular POIs are similar, which can effectively relieve sparsity problem for unpopular POIs by interest transfer between locals and travelers.

## 5.4 Hyperparameter Investigation (RQ3)

Figure 10 shows the impact of $\alpha$ and $\beta$ on SMF-UIDT and RMF-UIDT in Tencent-BS task, evaluated by NDCG. $\alpha$ controls the relative weight of hometown records on current city records, in other words, $\alpha$ measures to what extent we can rely on interest transfer for users in different cities, thus it should be a positive number no greater than 1, considering that our goal is to recommend POIs in current city. $\alpha = 0$ means hometown records are not used at all, while $\alpha = 1$ implies hometown records are utilized with the same importance as current city records.




(a) SMF-UIDT: $\alpha$ vs. NDCG (b) RMF-UIDT: $\alpha$ vs. NDCG (c) SMF-UIDT: $\beta$ vs. NDCG (d) RMF-UIDT: $\beta$ vs. NDCG

Fig. 10. Impact of hyper-parameters $\alpha$ and $\beta$ on SMF-UIDT and RMF-UIDT performance, in Tencent-BS task.

The peak performance of SMF-UIDT lies at $\alpha = 0.1$ or $0.2$, shown in Fig. 10(a), which approximately equals the ratio value between current city records and hometown records. The same pattern holds true for RMF-UIDT as in Fig. 10(b), since $\alpha$ has the same implication on both loss function. Such a relative low value of $\alpha$ indicates the limited impact of hometown records in cross-city POI recommendation problem, which make it necessary to distinguish interactions in different cities.

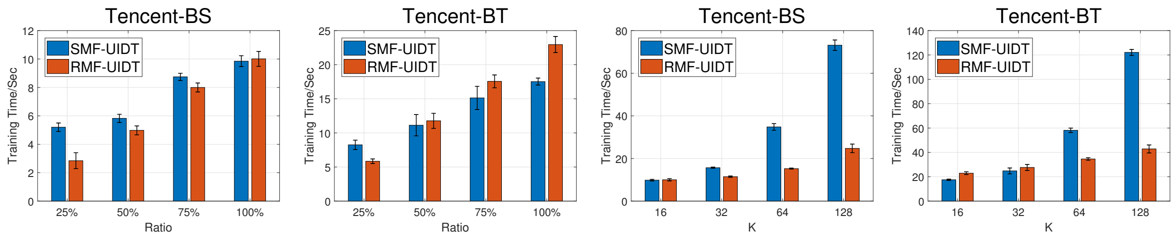
The other parameter $\beta$ constrains the relative difference of POI vectors for locals and travelers, and with the higher $\beta$, the difference between locals' visiting behaviors and travelers' visiting behaviors becomes the smaller. A middle value of $\beta$ reaches the peak of performances indicating the need to deliberately characterize constraints (Fig. 10(c) and (d)). Either strong or loose constraints lead to the degradation of performance. But for SMF and RMF that have a different loss function, the absolute value of $\beta$ at peak performance changes significantly, that is, 100 for SMF -UIDT and 1~10 for RMF-UIDT, while the variation trend remains the same.

From the comparison of different baselines, we find hometown records do provide valuable inference for user preference in target city. But through further grid search of the impact of hyper parameters $\alpha$ and $\beta$, these hometown records should be treated unequally with current city records, considering the limited impact and potential misleading inference. In such a cross-city situation, it is worthwhile to distinguish travelers and locals as different user groups, thus both STLDA and PR-UIDT, which follow such principle, achieve relative higher performance. However, PR-UIDT can still learn useful information from locals while STLDA cannot, so the fact PR-UIDT outperforms STLDA highlights the importance of considering different user group behaviors on the same POI.

## 5.5 Practical Analysis (RQ4)

*5.5.1 Efficiency.* First we evaluate the efficiency of we proposed PR-UIDT framework. SMF-UIDT has time complexity of $O((M+N)K^2 + RK)$, and RMF-UIDT has time complexity of $O(RK)$. We compare their efficiency on the same machine (Intel Xeon 2.10 GHz CPU), as shown in Fig. 11, in terms of record data size (Ratio) and the number of latent vectors ($K$). All results of single iteration time are measured in terms of both mean and standard deviation over 5 repeated experiments. In each experiment, we completely run the algorithm and divide the whole time spent on training with number of iterations to calculate the iteration time.

In Fig. 11(a) and (b), as data size increases, RMF-UIDT running time increase linearly, which is consistent with $O(RK)$ complexity. The change of SMF-UIDT time consumption is also almost the same, like in Tencent-BT, the model consumes 2.88, 4.00, 2.40 (sec) for every 25% records added, proving a linear related time complexity to $R$. In Fig. 11(c) and (d), while factor number $K$ is increasing exponentially, running time of SMF-UIDT multiplies as well. The original time complexity of SMF-UIDT is $O((M+N)K^2 + RK)$, but in our dataset $M+N$ is far less than $R$, so in experiments it can be seen as $O(RK)$ complexity which matches our experiment results. For RMF-UIDT, if




(a) data size (b) data size (c) latent factors (d) latent factors

Fig. 11. Training time (sec) per iteration of different model with varying data seize (Ratio) or number of factors ($K$).

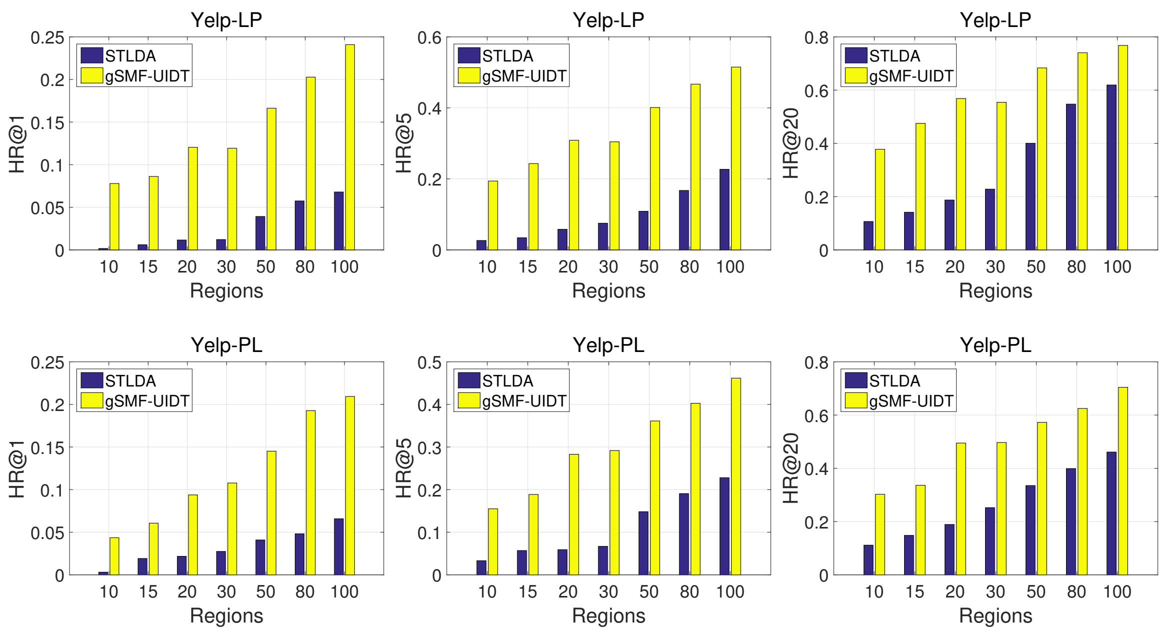



Fig. 12. The performance comparison between STLDA and SMF-UIDT that considers geographical effect (gSMF-UIDT).

we look into the difference time as $K$ increases, for example, in Tencent-BT task, it consumes 4.70, 6.95, 8.26 (sec) when $K$ increase to 16, 32, 64, the linear complexity still holds true.

Therefore, we demonstrate that both SMF-UIDT and RMF-UIDT integrate the additional information in an efficient way, making these two methods scalable and practicable for large-scale real-world data.

*5.5.2 Geographical effect.* As users tend to visit POIs close to their current location, this geographical effect is an important factor in real-world POI recommendation applications. To incorporate this effect into our PR-UIDT framework, we borrow the idea from STLDA, which considers geographical effect by multiplying the predicting score by a coefficient decided by user location and POI region. The geographical coefficient imposed by STLDA is calculated by assuming a Gaussian distribution of each region, and shown below as

$$p(l_u|\mu_r, \Sigma_r) = \frac{1}{2\pi\sqrt{|\Sigma_r|}} exp(\frac{-(l_u - \mu_r)^T \Sigma_r^{-1}(l_u - \mu_r)}{2}), \quad (18)$$

where $l_u$ is user's current location, which we use the location of her test POI, and $\mu_r, \Sigma_r$ is the gaussian distribution parameters of region $r$. We denote the SMF-UIDT that multiplies its predicting scores with the coefficient $p(l_u|\mu_r, \Sigma_r)$ as gSMF-UIDT. Fig. 12 compares the performance of gSMF-UIDT and STLDA in Yelp dataset. Hit Ratio at top 1/5/20 are used to evaluate model performances under different region scales. gSMF-UIDT has obvious advantage compared with STLDA, especially when region number is smaller than 30. With region number set to 100, gSMF-UIDT is able to achieve the HR@20 approaching 0.80, which indicates a high practical value in real-world POI recommendation applications. STLDA model benefits a lot form region split, as its performance increases significantly with region number increasing. However, this region information in training of STLDA cannot ensure that testing POIs are ranked in the top of recommendation list. In PR-UIDT model, we extract the valuable information of user preference by learning from both hometown and current city, thus we can not only recommend interesting POIs which would not appear in recommendation list before, but also make the right POI ranked higher than geographically closer POI, even if geographical effect is included in this model.

PR-UIDT are flexible to integrate geographical effect to make more precise recommendation. By following the same means, we believe our proposed PR-UIDT framework is capable of integrating more information, like temporal regularity, but since the main goal of this framework is to balance user interest transfer and drift in cross-city scenario, we ignore these regularities and present the most essential part as in PR-UIDT model.

## 5.6 Applicability in multi-city scenarios. (RQ5)

To evaluate PR-UIDT framework performance in multi-city scenario, we extract two subsets from Yelp datasets, named as Yelp-3city and Yelp-4city. Yelp-3city dataset contains all records in Las Vegas, Phoenix and Toronto, and Yelp-4city additionally contains Charlotte. Users or POIs with less than 10 interactions are filtered out to reduce data sparseness. The basic statistics of Yelp-3city (Yelp-4city) are 5,879 (6,834) users, 4,058 (4,824) POIs, 124,196 (147,490) hometown records and 20,768 (24,402) non-hometown records, respectively.

In multi-city datasets, the hometown of a user is defined as the city where he has most interactions, and all other cities are defined as non-hometown. Similar to what we did in 2-city scenario, the 80/20 principle is applied to non-hometown records to generate training and testing sets. When recommending POI for a user, only unvisited non-hometown POIs are considered as potential choices. We report the final performance of PR-UIDT and other baselines in Table 5, where the baselines are selected based on their competitive performance in 2-city scenario (Table 3). Note that the SASRec is not included because we focus on the user interest drift and transfer. Compared with these baselines, PR-UIDT framework brings 6.3%∼10.7% of performance improvement in multi-city POI recommendation. Even RMF-UIDT does not perform as well as SMF-UIDT, it still gets 11.0% improvements compared with vanilla BPR algorithm in terms of NDCG@100 in Yelp-3city, and 19.3% in Yelp-4city.

Table 5. Performance comparison in multi-city scenarios.

| Algorithm | | Vanilla | | Interest Drift | Proposed | | Gain |
|---|---|---|---|---|---|---|---|
| | | WRMF | BPR | STLDA$_c$ | WMF-UIDT | BPR-UIDT | |
| 3city | NDCG@100 | <u>0.1085</u> | 0.0866 | 0.0707 | **0.1153** | 0.0961 | +6.3% |
| | HR@100 | <u>0.3344</u> | 0.3100 | 0.2570 | **0.3577** | 0.3203 | +7.0% |
| 4city | NDCG@100 | <u>0.1031</u> | 0.0742 | 0.0648 | **0.1127** | 0.0885 | +9.3% |
| | HR@100 | <u>0.3158</u> | 0.2380 | 0.2487 | **0.3496** | 0.2935 | +10.7% |

Overall, these improvements prove that PR-UIDT framework is capable of accurately modeling user interest drift and transfer when users travel among multiple cities, which shows the generality of the proposed method.

# 6 RELATED WORK AND DISCUSSION

## 6.1 Related Work

We first review some related works on improving POI recommendation in a general case. Then, we detail the cross-region POI recommendation methods that is our main focus. Since this problem is related to transfer learning and cold-start problem to some extent, we further discuss related works on these two fields.

**POI Recommendation.** The most frequently utilized side information in POI recommendation is the geo-social influence [3, 41]. In terms of geographical information, [20] incorporated users' preference over spatial regions into a weighted MF framework. [23] exploited neighborhood characteristics to learn the better embedding of users and POIs. As for social context, many works took advantage of social relations to learn user preference on those POIs visited by their potential friends [17, 32]. Others also considered to combine above two parts of information together to improve POI recommendation [25, 47].

The second type of side informationi s temporal effect on user check-in behaviors [45]. [22] separated the motivation of visiting POI into a static interest and a time-relevant interest. [48] fused the sequential influence with geo-social influence into a unified recommendation framework. Recently, a deep recurrent collaborative filtering method was proposed to further model this effect by using recurrent neural network (RNN) unit [24].

The various type of content information on LBSNs can also help characterizing users' preference and thus improve POI recommendation. [8] modeled user-interest content and poi-property content by two overlapping word latent topics. User sentiment information embedded in review texts was also considered as an important complement [34, 50].

The above methods worked well in general cases. However, an important scenario in POI recommendation is for cross-region users who go to an unfamiliar region and thus the visit history is extremely sparse. In this paper, we propose a PR-UIDT framework that tackles this problem through modeling user interest drift and transfer between hometown and current city.

**Cross-region Recommendation.** The most challenging problem for cross-region recommendation is the data sparsity. Most existing works utilized POI content information [21, 46] or social relationship [6]. However, these works ignored a basic fact that users' visiting behaviors should be impacted by the locality, which requires fine-grained modeling [36, 43]. In addition, [52] grouped both users in the home city and those in the target city into communities by inferring their interests from contents, and seek for an optimal match between communities in order to support the cross-city POI recommendations. Recently, a state-of-the-art model [42] assumed that the personal interests are region-dependent and users' decisions are impacted by other users of the same role. Similarly, [16] proposed to separate the city topics into the city-specific topics and the common ones, and users from a source city will match with POIs in a target city if they are distributed in the same common topics.

Different from above existing works, we do not consider user interests to be fully region-dependent as in [42], instead we consider both region-dependent interests and region-independent interests at the same time, which is more accurate and reasonable. Moreover, we choose to model each user's interest drift and transfer between cities, instead of an aggregated view of the city topics as in [16], which is more fine-grained and effective.

**Transfer Learning in Recommendation.** Previous works of leveraging transfer learning for personalized recommendation mainly focused on transferring the aggregation level knowledge, for example, like the semantic features of locations for chain store site recommendation [9, 18], tag-inferred correlation for item recommendation [10] and perturbation-added POI check-in history for privacy preserving cross-domain location recommendation [7]. However, in the case of POI recommendation for cross-city travelers, it is more important to transfer personalized interests from hometown to current city, which is more challenging and not contained in these works.

**Cold-start Recommendation.** In terms of auxiliary information, [51] addressed cold-start product recommendation by using microblogging information. [44] used POI visiting information to improve smartphone app usage prediction. [33] solved cold-start location recommendation by learning user interest from app usage data. In terms of the methodology, [49] proposed a semi-supervised co-training algorithm. [19] proposed a content-aware collaborative filtering for location recommendation. [30] learned a matrix factorization that exploits items' properties and past user preferences while enforcing the manifold structure exhibited by the collective embeddings. However, all above methods are not suitable for cross-city POI recommendation because users' preference over POIs in hometown can not be directly utilized to help predicting their preference over those POIs in current city (which can be considered as cold-start POIs in this circumstance), which is due to the interest drift and also known as negative transfer in transfer learning domain. Instead, our proposed PR-UIDT only leverages the partial information corresponding to city-independent user interests.

### 6.2 Discussion

POI recommendation performance degrades significantly due to sparse interactions, especially for travelers in cross-city scenarios. In sharp contrast to previous work on leveraging cross-domain auxiliary information, we propose to solve the sparsity problem by additionally incorporating travelers' check-in records in hometown and locals' check-in records in current city at the same time. This data augmentation brings a tenfold increase in data volume, but such large amounts of extra records contain both instructive and misleading information. Therefore, it is necessary to take into account both user interest drift and transfer between hometown and current city. Motivated by this, we propose the PR-UIDT framework that leverages such information through the novel designs of user and POI latent vectors. With superiority on recommendation performance, our PR-UIDT also has a large practical value in real applications considering its high efficiency and ability of integrating geographical effect. In terms of limitation, we do not consider other auxiliary information like POI category, temporal effect and so on. Besides, the two models of SMF-UIDT and RMF-UIDT is basic MF models, which seem not powerful enough considering the recent development in neural recommender systems [11, 14, 39]. However, as we demonstrated in experiments, the PR-UIDT can easily integrate other auxiliary information to boost the recommendation performance. As for the neural network (NN) models, the PR-UIDT is a general framework that only put constraints on the latent vectors and loss functions, thus it can also replace the existed MF module with another NN module.

## 7 CONCLUSION

In this work we study the problem of cross-city POI recommendation for the travelers. To accurately characterize user interest, we propose a PR-UIDT framework that enable learning from users' visiting behaviors in both hometown and current city. Through the novel design of user and POI vectors, both user interest drift and transfer among different cities are effectively modelled in PR-UIDT. Experiments on real-world datasets show significant improvements, about $0.4\% \sim 20.5\%$ increase compared with other state-of-the-art models, proving the superiority of PR-UIDT framework, and further studies demonstrate the importance of combining both interest transfer and drift in cross-city POI recommendation problems. Moreover, we also evaluate the efficiency of PR-UIDT and the recommendation performance after integrating geographical effect. The results show great adaptability and applicability of PR-UIDT in real-word applications, with high efficiency and ability of integrating various types of auxiliary information.

For further study, we aim to explore whether it's possible to include cross-domain information like POI category into our PR-UIDT framework to reach better performance. Another possible improvement may lie in differentiated values of hyper-parameters instead of stable for users and POIs to more accurately tradeoff between interest drift and transfer. We believe it is possible to build a smart and pragmatic POI recommender no matter how sparse the data is or how hidden user preference is, by utilizing all possible information for designing recommender models.

## ACKNOWLEDGMENTS

This work was supported in part by The National Key Research and Development Program of China under grant SQ2018YFB180012, the National Nature Science Foundation of China under 61971267, 61972223, 61861136003, and 61621091, Beijing Natural Science Foundation under L182038, Beijing National Research Center for Information Science and Technology under 20031887521, and research fund of Tsinghua University - Tencent Joint Laboratory for Internet Innovation Technology.

## REFERENCES

[1] Michael Backes, Mathias Humbert, Jun Pang, and Yang Zhang. 2017. walk2friends: Inferring social links from mobility profiles. In *CCS*. 1943–1957.

[2] Rocío Cañamares and Pablo Castells. 2018. Should i follow the crowd?: A probabilistic analysis of the effectiveness of popularity in recommender systems. In *SIGIR*. 415–424.

[3] Chen Cheng, Haiqin Yang, Irwin King, and Michael R Lyu. 2012. Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks. In *AAAI*. 17–23.

[4] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Recsys*. 39–46.

[5] Jingtao Ding, Guanghui Yu, Xiangnan He, Yuhan Quan, Yong Li, Tat-Seng Chua, Depeng Jin, and Jiajie Yu. 2018. Improving Implicit Recommender Systems with View Data.. In *IJCAI*. 3343–3349.

[6] Gregory Ference, Mao Ye, and Wang-Chien Lee. 2013. Location recommendation for out-of-town users in location-based social networks. In *CIKM*. 721–726.

[7] Chen Gao, Chao Huang, Yue Yu, Huandong Wang, Yong Li, and Depeng Jin. 2019. Privacy-preserving Cross-domain Location Recommendation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 1 (2019), 11.

[8] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2015. Content-Aware Point of Interest Recommendation on Location-Based Social Networks. In *AAAI*. 1721–1727.

[9] Bin Guo, Jing Li, Vincent W Zheng, Zhu Wang, and Zhiwen Yu. 2018. Citytransfer: Transferring inter-and intra-city knowledge for chain store site recommendation based on multi-source urban data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 135.

[10] Peng Hao, Guangquan Zhang, Luis Martinez, and Jie Lu. 2019. Regularizing knowledge transfer in recommendation with tag-inferred correlation. *IEEE transactions on cybernetics* 49, 1 (2019), 83–196.

[11] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *WWW*. 173–182.

[12] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In *SIGIR*. 549–558.

[13] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *ICDM*. 263–272.

[14] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *ICDM*. 197–206.

[15] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.

[16] Dichao Li, Zhiguo Gong, and Defu Zhang. 2018. A Common Topic Transfer Learning Model for Crossing City POI Recommendations. *IEEE transactions on cybernetics* 99 (2018), 1–14.

[17] Huayu Li, Yong Ge, Richang Hong, and Hengshu Zhu. 2016. Point-of-interest recommendations: Learning potential check-ins from friends. In *KDD*. 975–984.

[18] Jing Li, Bin Guo, Zhu Wang, Mingyang Li, and Zhiwen Yu. 2016. Where to place the next outlet? harnessing cross-space urban data for multi-scale chain store recommendation. In *Ubicomp Adjunct*. 149–152.

[19] Defu Lian, Yong Ge, Fuzheng Zhang, Nicholas Jing Yuan, Xing Xie, Tao Zhou, and Yong Rui. 2018. Scalable Content-Aware Collaborative Filtering for Location Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 30, 6 (2018), 1122–1135.

[20] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 2014. GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *KDD*. 831–840.

[21] Xiao Lin, Min Zhang, and Yongfeng Zhang. 2017. Joint factorizational topic models for cross-city recommendation. In *APWeb/WAIM Joint Conference on Web and Big Data*. 591–609.

[22] Yanchi Liu, Chuanren Liu, Bin Liu, Meng Qu, and Hui Xiong. 2016. Unified point-of-interest recommendation with temporal interval assessment. In *KDD*. 1015–1024.

[23] Yong Liu, Wei Wei, Aixin Sun, and Chunyan Miao. 2014. Exploiting geographical neighborhood characteristics for location recommendation. In *CIKM*. 739–748.

[24] Jarana Manotumruksa, Craig Macdonald, and Iadh Ounis. 2017. A Deep Recurrent Collaborative Filtering Framework for Venue Recommendation. In *CIKM*. 1429–1438.

[25] Jarana Manotumruksa, Craig Macdonald, and Iadh Ounis. 2017. A Personalised Ranking Framework with Multiple Sampling Criteria for Venue Recommendation. In *CIKM*. 1469–1478.

[26] Andriy Mnih and Ruslan R Salakhutdinov. 2008. Probabilistic matrix factorization. In *NIPS*. 1257–1264.

[27] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. 452–461.

[28] Steffen Rendle, Zeno Gantner, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2011. Fast context-aware recommendations with factorization machines. In *SIGIR*. 635–644.

[29] Ruslan Salakhutdinov and Andriy Mnih. 2008. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *ICML*. 880–887.

[30] Martin Saveski and Amin Mantrach. 2014. Item cold-start recommendations: learning local collective embeddings. In *Recsys*. 89–96.

[31] Suvash Sedhain, Scott Sanner, Darius Braziunas, Lexing Xie, and Jordan Christensen. 2014. Social collaborative filtering for cold-start recommendations. In *Recsys*. 345–348.

[32] Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. 2013. Exploiting local and global social context for recommendation. In *IJCAI*. 2712–2718.

[33] Zhen Tu, Yali Fan, Yong Li, Xiang Chen, Li Su, and Depeng Jin. 2019. From Fingerprint to Footprint: Cold-start Location Recommendation by Learning User Interest from App Data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 1 (2019), 26.

[34] Hao Wang, Yanmei Fu, Qinyong Wang, Hongzhi Yin, Changying Du, and Hui Xiong. 2017. A location-sentiment-aware recommender system for both home-town and out-of-town users. In *KDD*. 1135–1143.

[35] Leye Wang, Bin Guo, and Qiang Yang. 2018. Smart City Development With Urban Transfer Learning. *Computer* 51, 12 (2018), 32–41.

[36] Weiqing Wang, Hongzhi Yin, Ling Chen, Yizhou Sun, Shazia Sadiq, and Xiaofang Zhou. 2015. Geo-SAGE: A geographical sparse additive generative model for spatial item recommendation. In *KDD*. 1255–1264.

[37] Zihan Wang, Ziheng Jiang, Zhaochun Ren, Jiliang Tang, and Dawei Yin. 2018. A path-constrained framework for discriminating substitutable and complementary products in e-commerce. In *WSDM*. 619–627.

[38] Ying Wei, Yu Zheng, and Qiang Yang. 2016. Transfer knowledge between cities. In *KDD*. 1905–1914.

[39] Feng Xue, Xiangnan He, Xiang Wang, Jiandong Xu, Kai Liu, and Richang Hong. 2019. Deep Item-based Collaborative Filtering for Top-N Recommendation. *ACM Transactions on Information Systems (TOIS)* 37, 3 (2019), 33.

[40] Lina Yao, Quan Z Sheng, Yongrui Qin, Xianzhi Wang, Ali Shemshadi, and Qi He. 2015. Context-aware point-of-interest recommendation using tensor factorization with social regularization. In *SIGIR*. 1007–1010.

[41] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*. 325–334.

[42] Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, and Quoc Viet Hung Nguyen. 2016. Adapting to user interest drift for poi recommendation. *IEEE Transactions on Knowledge and Data Engineering* 28, 10 (2016), 2566–2581.

[43] Hongzhi Yin, Xiaofang Zhou, Yingxia Shao, Hao Wang, and Shazia Sadiq. 2015. Joint modeling of user check-in behaviors for point-of-interest recommendation. In *CIKM*. 1631–1640.

[44] Donghan Yu, Yong Li, Fengli Xu, Pengyu Zhang, and Vassilis Kostakos. 2018. Smartphone app usage prediction using points of interest. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 174.

[45] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. 2013. Time-aware point-of-interest recommendation. In *SIGIR*. 363–372.

[46] Chenyi Zhang and Ke Wang. 2016. POI recommendation through cross-region collaborative filtering. *Knowledge and Information Systems* 46, 2 (2016), 369–387.

[47] Jia-Dong Zhang and Chi-Yin Chow. 2015. GeoSoCa: Exploiting geographical, social and categorical correlations for point-of-interest recommendations. In *SIGIR*. 443–452.

[48] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. 2014. Lore: Exploiting sequential influence for location recommendations. In *SIGSPATIAL*. 103–112.

[49] Mi Zhang, Jie Tang, Xuchen Zhang, and Xiangyang Xue. 2014. Addressing cold start in recommender systems: A semi-supervised co-training algorithm. In *SIGIR*. 73–82.

[50] Kaiqi Zhao, Gao Cong, Quan Yuan, and Kenny Q Zhu. 2015. SAR: A sentiment-aspect-region model for user preference analysis in geo-tagged reviews. In *ICDE*. 675–686.
[51] Wayne Xin Zhao, Sui Li, Yulan He, Edward Y Chang, Ji-Rong Wen, and Xiaoming Li. 2016. Connecting social media to e-commerce: Cold-start product recommendation using microblogging information. *IEEE Transactions on Knowledge and Data Engineering* 28, 5 (2016), 1147–1159.
[52] Yi-Liang Zhao, Liqiang Nie, Xiangyu Wang, and Tat-Seng Chua. 2014. Personalized recommendations of locally interesting venues to tourists via cross-region community matching. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 3 (2014), 50.