Expert Systems with Applications 42 (2015) 4274–4283

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Learning recency based comparative choice towards point-of-interest recommendation

Xin Li [a], Guandong Xu [b,*], Enhong Chen [a], Yu Zong [c]

[a] University of Science and Technology of China, China
[b] University of Technology Sydney, Australia
[c] West Anhui University, China

ARTICLE INFO

Article history:
Available online 4 February 2015

Keywords:
Points-of-Interest
Recommendation
Choice model
Partial order
Learning algorithm

ABSTRACT

With the prevalence of GPS-enabled smart phones, *Location Based Social Network* (LBSN) has emerged and become a hot research topic during the past few years. As one of the most important components in LBSN, *Points-of-Interests* (POIs) has been extensively studied by both academia and industry, yielding POI recommendations to enhance user experience in exploring the city. In conventional methods, rating vectors for both users and POIs are utilized for similarity calculation, which might yield inaccuracy due to the differences of user biases. In our opinion, the rating values themselves do not give exact preferences of users, however the numeric order of ratings given by a user within a certain period provides a hint of preference order of POIs by such user. Firstly, we propose an approach to model users preference by employing utility theory. Secondly, We devise a collection-wise learning method over partial orders through an effective stochastic gradient descent algorithm. We test our model on two real world datasets, i.e., Yelp and TripAdvisor, by comparing with some state-of-the-art approaches including PMF and several user preference modeling methods. In terms of MAP and Recall, we averagely achieve 15% improvement with regard to the baseline methods. The results show the significance of comparative choice in a certain time window and show its superiority to the existing methods.

1. Introduction

Recent years have witnessed the booming of GPS-enabled smart phones, thus the gap between physical and real world has been blurred and location-based services emerge. Along with online social media, Location Based Social Networks (LBSN), such as Foursquare, Facebook Places, and Yelp, become prevalent in recent years. These LBSNs allow users to explore Points-of-Interests (POIs), such as restaurants and entertainment clubs for better services by sharing their experiences and opinions about the places they visited. For example, on Yelp website and with mobile apps, users can (1) check-in at POIs; (2) give ratings to such places; and (3) write reviews and tips for shops or restaurants to show their likes or dislikes about the places. Generally, "A Point-of-interest is a specific point of location that someone may find useful or interesting. Most consumers use the term when referring to hotels, campsites, the stations or any other categories used in modern navigation system" (extracted from Wikipedia entry). In this paper we treat POI as a Business or Merchant at a specific location, and utilize ratings and time stamps which are two common and easily accessible resources in LBSNs to model user preference in turn for recommendation, i.e., POI and merchant are used interchangeably. In a common sense, our proposed approach can be deployed in broader recommendation applications.

Both LBSN users and POI merchants can benefit from the recommendations. On one hand, LBSN users can gain better user experience and satisfaction in terms of quality and service by utilizing the recommendations of POI made. On the other hand, POI merchants will attract more customer visits and increase the business turnover given the appraisal voted by customers. Moreover, POI merchants can get additional benefit through the analysis of user check-in and review data in LBSNs, e.g., understanding their reputations or concerns in customers. Thus POI-recommendation becomes a hot topic in LBSNs research and application.

Traditionally, we can adopt conventional recommendation methods by simply treating POIs as ordinary items (Ye, Yin, & Lee, 2010). Thus, lots of models, such as *neighborhood-based* (Linden, Smith, & York, 2003) or *model-based* (Koren, Bell, & Volinsky, 2009; Hu, Koren, & Volinsky, 2008) approaches can be utilized seamlessly, such as *collaborative filtering* (CF) based POI

* Corresponding author.
E-mail addresses: leexin@ustc.edu.cn (X. Li), guandong.xu@uts.edu.au (G. Xu), cheneh@ustc.edu.cn (E. Chen), nick.zongy@gmail.com (Y. Zong).

http://dx.doi.org/10.1016/j.eswa.2015.01.054
0957-4174/

recommendation. Likewise, such approaches mainly rely on the user-POI rating matrix, where each element represents a rating given by a user to a POI, to calculate similarity between users or POIs. Apparently, the rating value indicates a preference score of a user on specific POI, however, simply utilizing user-rating or POI-rating vectors alone to make the similarity calculation might yield inaccuracy due to the differences of user biases. Though some studies attempt to incorporate user bias into the models to alleviate the inappropriateness of modeling user preference, we argue that the user rating behavior is not sufficiently investigated and modeled. In contrast, we think although the rating values of users don't give the exact preference degrees of users to various POIs, the numeric order of ratings given by a user within a certain period of time at least provides a hint of preference order of POIs by this user, i.e., the higher rating denotes a more preferable judgment, and vice versa. With such observed preference orders by all users, we can train a POI preference prediction model by a learning process, upon which our POI recommendation model is initialized.

The aforementioned summaries shed light on us to give the following assumptions: (1) **Relativization**: By individually viewing the rating a user gives to a particular POI, we sometimes can not tell the actual user preference to this POI due to the user biases. For example, a user gives 4 star to a POI out of average rating 4.5, which indicates the preference to such POI is below the average. However, supposing that the same POI was rated 4 star by another user, whose average score is 3.5, we can say that this user may be more critical and prefers to this POI more than others. In such case, using the absolute ratings alone sometimes could not capture the accurate user preference, yielding unsatisfactory recommendation results POIs, which motivate us to take a relative view to address this problem. Our idea in this paper is to leverage the rating order rather than rating value from user rating history to accurately learn user preference. The intuition is quite straightforward - the order of ratings by users implies the underlying user preference information. Take a toy example shown in Fig. 1, we can see that the user prefers POI p_5 more than p_4 due to the higher rating on p_5 than p_4. (2) **Comparative Choice**: The user rating behavior is indeed a comparative process, where a rating given by a user is actually a choice making after the user compared the target POI with other visited POIs rather than a random determination. (3) **Recency**: Although the user rating is a comparative decision, not all rating history of the user should be considered due to the short-term memory effect, i.e., the user is more likely to compare the target POI with recently visited POIs rather than those visited a long time ago, and finally gives a rating. In other words, recency is the key factor in comparison and decision. Also known from the toy example in Fig. 1, the comparative relationship between p_5 and p_4 is more reliable than that between p_5 and p_3. Thus setting an appropriate time window to form the comparative POI set becomes another research question we need to tackle.

In this paper, we aim to address the problems based on the aforementioned assumptions and propose a new model to learn user preference from user rating behaviors through employing choice model from utility theory. Our idea is originated from the assumptions that user rating is mainly determined by recent comparative experiences of users, i.e., comparative choice and recency, the rating given to a POI by a user is a result of comparisons to other POIs recently visited by this user. The collection of all observed users' comparative choices forms a training set for us to learn the user preferences over various POIs in an aggregated view. More specifically, from the rating history of each user, we count in all POIs within a defined time window, e.g., previous K visits before the current POI, to form a POI collection. Naturally, in each collection, POIs can be sorted according to their ratings in a partial order, where through the view of economics, a higher rating indicates the more satisfaction of user to the POI, meanwhile implies the higher utility of a particular POI to a certain user. Hence, by employing choice model deduced from utility theory, we could simulate the user rating behavior from such partial order relationship, learn the user preference more precisely, and in turn, provide better recommendations.

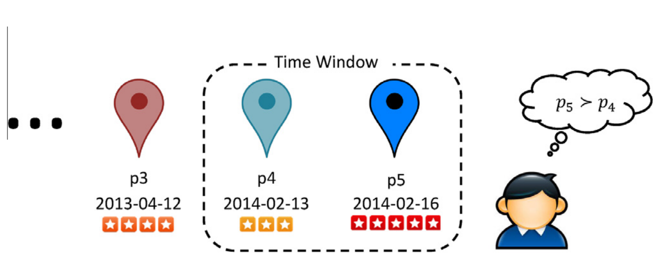


Fig. 1. Toy example.

In summary, we made the following contributions:

- We propose a novel approach to model user rating behavior by exploring comparative relationship between ratings within a certain time window.
- We design a choice model and employ collection-wised learning over partial orders through an effective and efficient stochastic gradient descent algorithm.
- We conduct extensive experiments to evaluate the performance of our model on two large-scale real datasets. The results show that our approach outperforms other state-of-the-art methods.

The remainder of the paper is organized as follows.Section 2 provides an overview of the related work. Section 3 gives the formulation for the problem we study. We detail our model in Section 4 and report the learning and inference in Section 5. The result of the experiments are presented in Section 6, followed by the conclusions and future work in Section 7.

2. Related work

In this section, we review a number of works on recommendations especially in POI recommendation, which are related to our approach.

2.1. Traditional recommendations

Traditional recommender systems mainly focus on user-item rating matrix by employing memory-based (Linden et al., 2003) or model-based collaborative filtering(CF) (Koren et al., 2009; Hu et al., 2008). The premise behind memory-based CF is to recommend items by like-minded user to a given user and the intuition of model-based CF like matrix factorization is that only a few latent factors are in control. Moreover, contextual information like social or trust network are embedded into models for further improvements of the prediction accuracy (Jamali & Ester, 2009; Xiang et al., 2010; Deng, Huang, & Xu, 2014). Text, Tags and temporal information are utilized for recommendation of music (Hyung, Lee, & Lee, 2014), news (Li, Zheng, Yang, & Li, 2014), product (Hong, Li, & Li, 2012) and tagging system (Zheng & Li, 2011). Our research topic is different from traditional recommendations since traditional methods rely much on the entire rating vectors to calculate similarities for estimating user preference while ignoring the differences of user biases. By using numeric order of ratings to depict user preference over POIs is more closer to the real situation and elaborating such mechanism is significant for understanding user behaviors, which needs to be explicitly modeled in our proposed approach.

2.2. POI recommendations

Recently, due to the prevalence of smart portable devices, POI recommendation attracts much research interest and has many applications. Here we use location and POI interchangeably. Typically, conventional CF methods can be applied into this field by considering POI as an item (Ye et al., 2010). Besides, a wide range of properties of LBSN have been extensively explored, such as geographical location (Cheng, Yang, King, & Lyu, 2012; Ye, Yin, Lee, & Lee, 2011b; Liu, Fu, Yao, & Xiong, 2013), social connections (Cho, Myers, & Leskovec, 2011; Liu & Xiong, 2013), temporal information (Gao, Tang, Hu, & Liu, 2013; Yuan, Cong, Ma, Sun, & Magnenat-Thalmann, 2013; Ye, Janowicz, Mülligann, & Lee, 2011a) and POI descriptions (Yin, Sun, Cui, Hu, & Chen, 2013).

Under the observation that strong geographical and social ties exist among users and their favorite locations, Ye etc., (Ye et al., 2010) extends the conventional neighborhood-based CF method with different similarity measurements derived from geographical and social information. As discussed in (Ye et al., 2011b), Ye proposed a unified POI recommendation framework by fusing both social and geographical information based on naive Bayesian, where geographical influence is modeled by power law distribution. However, multi-center Gaussian model is proposed in Cheng et al. (2012) to model the probability of a user's check-in history, and finally been fused into MF for POI recommendation. Liu developed LDA embedded Topic-Location PMF and geographical probabilistic factor model respectively to explore location preference in LBSN (Liu et al., 2013; Liu & Xiong, 2013). In Yin et al. (2013), Yin etc., also employed LDA and further extended it to LCA-LDA to simultaneously consider user interest and local preference.

2.3. Time-aware & preference modeling

Time is also an crucial contextual information in LBSN and has always been segmented into several time slots. Yuan etc., in Yuan et al. (2013) incorporated temporal influence with user-based CF to recommend locations. Gao in Gao et al. (2013) utilized MF to model temporal information based on two important properties, i.e., non-uniformness and consecutiveness. Yehuda modeled time changing user biases (Koren, 2009) while He incorporated time-context into traditional CF for recommendation (He & Wu, 2009). Our study differentiates itself from all these existing studies with the following point: Given the ratings and their corresponding time stamps, by utilizing numeric order of ratings within a certain period, we employ choice model to learn user preferences, which is a real annotation of user rating behavior.

Yang (Yang, Long, Smola, Zha, & Zheng, 2011) explored the competitive process of user behavior when facing with recommendations by adopting utility theory, which is most relevant to our work. However, the author took missing values in rating matrix as implicit feedbacks while we model rating behavior by adopting time window across recent visits to depict user short-term memory precisely. To deem recommendation as a personalized ranking problem, Steffen (Rendle, Freudenthaler, Gantner, & Schmidt-Thieme, 2009) integrated implicit feedback into the model to propose a generic optimization criterion derived from the maximum posterior estimator. To be different from other studies, we consider time-window in representation of user's short-term memory in order to better depict user preference.

2.4. Main difference & novelty

In reviewing a number of aforementioned works, our paper mainly differentiates itself in three aspects.

Firstly, based on the fact that user biases do exist, we take numeric order of ratings to better depict user preference over POIs while traditional methods mainly rely on the entire rating vectors. Secondly, although abundant information are included in LBSN and lots of works make efforts on that, in some occasion some of them are not accessible. We only leverage rating along with its time stamp which are the two most common and easily accessible data resources in LBSN for modeling user preference, thus making our model more general. Thirdly, we integrate time window into our user preference model. To the best of our knowledge, no existing works focusing on preference modeling consider time in this way and no time-aware recommendations utilize partial order of ratings to depict user preference.

3. Problem formulation

Supposing that we have N users and M POIs and let $\mathbb{U} = \{u_1, u_2, \ldots, u_N\}$ and $\mathbb{V} = \{p_1, p_2, \ldots, p_M\}$ be the set of users and POIs respectively. Generally, a user u_i can give a rating r_{ij} to a POI p_j. All ratings could form a user-POI rating matrix R with its entry denoted as r_{ij}.

As mentioned in Section 1, ratings of a certain user can be ranged according to the time when the user gives the rating. Fig. 2 presents us the time stamps of rating histories for u_1. With the preset time window, for each user we count in a number of the previous POIs to form a collection, which we term as recent visit collection:

Definition 1 (*Recent Visit Collection*). A recent visit collection of the current POI for a particular user u_i is a set of POIs, where all the POIs satisfy two conditions: (1) the time stamp of each POI k in this collection is within a time window T before that of the current POI j (2) user u_i have visited p_k in history, thus:

$$C_{ij}^T = \{p_j\} \bigcup \{p_k | \forall i, \exists r_{ik}, \quad \text{st. } t(i,p_j) - t(i,p_k) \leqslant T\} \tag{1}$$

where $t(i,*)$ is the visit time of a POI $*$ for a particular user u_i.

Actually, T refers to the period users often take into consideration when giving the current rating, which could be weekly, monthly, quarterly or even yearly. Alternatively in our experiments, T should not necessarily be a time, We adopt the previous K visits to simulate the time window. For now, in order to simplify the discussion, we use the definition shown in Eq. (1). In Fig. 2, supposing that u_1 visits a POI p_6 at *2014/02/17*, $\{p_4, p_5, p_6\}$ forms a recent visit collection when we set T to a *week* or set K = 2, representing the previous 2 visits. In real data set, we obtain a large amount of collections for each POI visit, illustrated in Fig. 3.

Each of the collections we obtained is a reflection of rating process of comparison, with higher rating indicating more satisfaction of a user to a particular POI. Thus all POIs in the collection can be ranged in a partial order relationship with regard to their ratings.

Definition 2 (*POI Partial Order*). Each POIs in C_{ij}^T has a rating, i.e., 1 to 5. Naturally a partial order in C_{ij}^T is shown below:

$$\succeq_{ij}^T = \left\{ \widehat{p}_1 \succeq \widehat{p}_2 \succeq \ldots | r_{i\widehat{p}_k} \geqslant r_{i\widehat{p}_{k+1}}, \widehat{p}_k \in C_{ij}^T \right\} \tag{2}$$

POI	Rating	Date
p_1	5	2012-06-12
p_3	4	2013-04-12
p_4	3	2014-02-13
p_5	5	2014-02-16

Fig. 2. Example of rating and date for user 1.

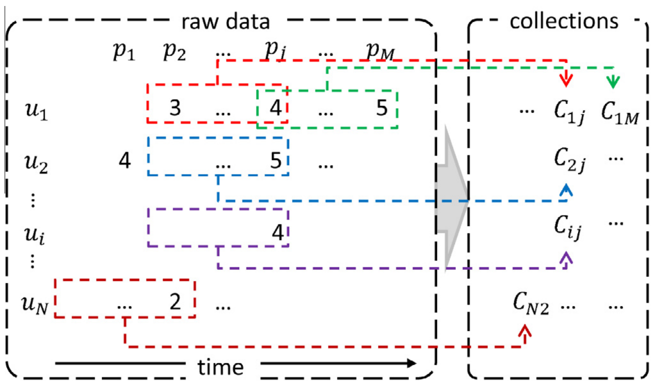


Fig. 3. The formulation of recent visit collections, with time window marked as colored dashed line.

where $r_{i\widehat{p}_k}$ denotes the rating user i give to POI $\widehat{p}_k$.

In the above equation, symbol $\geqslant$ compares two numeric values while symbol $\succeq$ denotes more preferable between two POIs. For example, $\widehat{p}_1 \succeq \widehat{p}_2$ means user i prefers $\widehat{p}_1$ than $\widehat{p}_2$.

Intuitively, why user give a rating to a POI higher than the others is mainly because his recent comparative experiences, i.e., he/she rates a POI by referring to the recently visited POIs and their corresponding ratings. Using the same example, once the rating of p_6 by u_1 is granted, for Example 4 star, the collection $\{p_4, p_5, p_6\}$ can be represented in the way $\succeq_{1,6}^{week} = (p_5 \succeq p_6 \succeq p_4)$.

By the definition above, the purpose in our paper is to recommend personalized top-k POIs for users given the observation of partial order collections in Fig. 3.

4. Model

As aforementioned, user rating is indeed a process of comparative choice. We try to formally model the rating behaviors in this section.

Choice analysis attempts to model the decision process of an individual under a particular context. It was widely adopted in economics and psychophysics (Greene & Hensher, 2008; Train, 2009), which can describe, explain and predict choices between two more multiple alternatives. In this section, we design a comparative choice model to better analyze the rating behavior.

4.1. Rating behavior of comparison

In general, discrete choice models are usually derived from utility theory (Neumann & Morgenstern, 2007). Utility is a representation of preference over a set of alternatives. This is consistent with the situation of user's rating decision making. In our case, given a user, the alternatives refer to POIs he visited. The rating a user gives to a POI indicates his/her satisfaction to such place, which is depicted as the utility of a certain POI to the user in choice model.

In order to make it formalized and easily interpretable in utility theory. Following the definition from previous section, we introduce the utility function $\mathcal{U}(u, p) : p \to \mathbb{R}$, which means user u_i assigns each candidate POI in collection C_{ij}^T with a real value. Naturally, a larger utility value shows more satisfaction and thus a higher rating should be granted to the POI. Hence the partial order of POIs in C_{ij}^T could be easily explained and become meaningful. To achieve the coincidence between the intuitive idea of user's preference among POIs which is defined in Eq. (2) and formalization of utility function we introduced above, below is a formula to show how our idea on user's partial preference is transferred to utility theory. We gives the partial order for POIs as follows:

$$p_j \succeq p_{j'} \text{ IIF } \mathcal{U}(u_i, p_j) \geqslant \mathcal{U}(u_i, p_{j'}) \tag{3}$$

where $\mathcal{U}(u_i, p_j)$ can be abbreviated to $\mathcal{U}_{ij}$. Eq. (3) shows how user's preference on POI j and j' are expressed in utility function. By this way, utility function can be deemed as the quantitative representation of the qualitative judge of user's preference. Recall the rationale we proposed in the previous sections, we suppose that two ratings of a user within a certain time window satisfy partial order, which means user i prefer POI j to j' if the rating of POI j is larger than that of POI j'. However, according to utility theory, the reason why a user gives a lower rating to a particular POI is due to the utility of such POI is of less use to the user. Thus we obtain the Eq. (3) which directly reflects our ideas.

From now on, employing derivation on utility theory to solve our real problem is straightforward. Usually, utility is decomposed into two parts (Train, 2009) based on random utility model (RUM). The formulation is as follows:

$$\mathcal{U}(u_i, p_j) = \mathcal{V}(u_i, p_j) + \varepsilon_{ij} \tag{4}$$

where the first part is what we observed and the second part is some unobserved factors like emotion, weather or even some occurrent events. In our case, the observed part is depicted as ratings, i.e., $\mathcal{V}(u_i, p_j) = r_{ij}$. For simplicity, we can use the latent factor based predicted rating $U_i V_j^T$ to qualify the observed utility. $U_i V_j^T$ is borrowed from matrix factorization techniques (Koren et al., 2009). The premise behind the matrix factorization is that both user and item are affected by a number of factors, among which only a few of them termed as latent factors are determinable. The number of latent factors is called dimension as well.

According to Thurstone (1927), the probability of user preference over alternatives can be defined in terms of the utility of choice:

$$Pr(p_j \succeq p_{j'}) = Pr(\mathcal{U}(u_i, p_j) \geqslant \mathcal{U}(u_i, p_{j'})) \tag{5}$$

Further, we replace the $\mathcal{U}$ with the Eq. (4), we obtain:

$$\begin{aligned} Pr(p_j \succeq p_{j'}) &= Pr(\mathcal{V}(u_i, p_j) + \varepsilon_{ij} \geqslant \mathcal{V}(u_i, p_{j'}) + \varepsilon_{ij'}) \\ &= Pr(\varepsilon_{ij'} \leqslant \varepsilon_{ij} + \mathcal{V}(u_i, p_j) - \mathcal{V}(u_i, p_{j'})) \\ &= CDF(\varepsilon_{ij} + \mathcal{V}(u_i, p_j) - \mathcal{V}(u_i, p_{j'})) \end{aligned} \tag{6}$$

where CDF is cumulative density function. Normally, we assume that the error term $\varepsilon_{ij'}$ ~*iid extreme value*, that is the double exponential format as $exp(-e^{-\varepsilon})$. Finally, the probability for user preference over POIs in a collection is deduced as follows.

$$Pr(p_j \succeq p_{j'}) = \frac{e^{\mathcal{V}(u_i, p_j)}}{\sum_{j'}^{|C^T|} e^{\mathcal{V}(u_i, p'_j)}} \tag{7}$$

Note that the equation we derived above has the same formulation as *multinomial logit model*.

4.2. Choice model

We model the process for a certain visit with recent visits involved. Since for each visit we obtain a collection, we can easily get the probability for the whole observations:

$$Pr(\succeq^T) = \prod_{u \in \mathbb{U}} \prod_{p_j \in \mathbb{V}(u)} Pr(p_j \succeq p_{j'}) \tag{8}$$

where $\mathbb{V}(u)$ denotes the POIs visited by user u. As aforementioned, we use latent factor based predicted rating to qualify observed utility, thus we can place priors on both U and V. Generally, a spherical multivariate Gaussian prior is assumed. That is

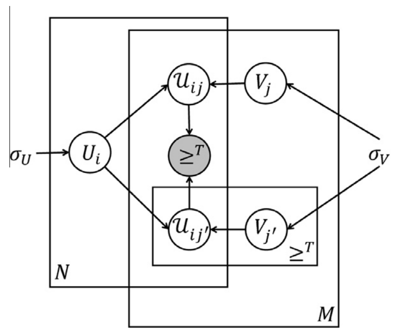


Fig. 4. Model for user comparative rating process.

$$Pr(\Omega|\Theta) = \mathcal{N}(\Omega|\mathbf{0},\sigma^2\mathbf{I}) = \lambda e^{-\frac{\Sigma_l^L\Omega_l^2}{2\sigma^2}} \tag{9}$$

where $\Omega = \{U, V\}$, and Θ denotes some hyper-parameters. Ω_l is a component of Ω.

The graphical model is shown in Fig. 4. The dark circle is the observation variable which denotes all the partial order collections we have and are also taken as training set. The blank circle such as vector U_i, V_j are unknown parameters needed to be learnt. As soon as U_i and V_j are gained, according to matrix factorization theory, a rating or utility can be approximately calculated, which is formally shown in Section 5.2. Two parameters out of the box, i.e., σ_U, σ_V, are hyper-parameters where priors should be placed on. N and M refer to the number of users and POIs respectively.

4.3. Optimization problem

With the Eqs. (8) and (9), we can immediately obtain the posterior according to the Bayesian theorem.

$$Pr(U,V|\succeq^T) \propto Pr(\succeq^T|U,V)Pr(U|\Theta)Pr(V|\Theta) \tag{10}$$

Straightforwardly, we can maximize the posterior to learn the latent factor. Equivalently, we can minimize its negative log form. Therefore, the objective function is given by:

$$\Omega = \arg\min_{\Omega} - \left[logPr(\succeq^T|\Omega) + logPr(\Omega|\Theta)\right] \tag{11}$$

where Ω is defined above and the second part $logPr(\Omega|\Theta)$ can be deemed as regularizer $R(\Omega)$ to alleviate overfitting problem. Here we place gaussian prior on $\Omega, R(\Omega)$ corresponds to L2 norm regularization.

5. Learning and inference

As presented in previous section, we need to learn the latent factor U and V of the model. In this section we design a stochastic gradient descent (SGD) algorithm to efficiently learn the latent factor. After that, we show how to make recommendations with the latent factor matrices.

5.1. Learning algorithm

For the Eq. (11), the gradient of Ω can be operated on both two parts respectively. That is:

$$\nabla\Omega_l(\succeq^T) = -\frac{\partial logPr(\succeq^T|\Omega)}{\partial\Omega_l} - \frac{\partial logPr(\Omega|\Theta)}{\partial\Omega_l} \tag{12}$$

for the second part, i.e., the regularizer induces L2 norm, aka:

$$-\frac{\partial logPr(\Omega|\Theta)}{\partial\Omega_l} = \frac{R(\Omega)}{\partial\Omega_l} = \lambda\Omega_l \tag{13}$$

for first part, we present the form as:

$$-\sum_{u\in\mathbb{U}}\sum_{p_j\in\mathbb{V}(u)}\frac{\partial logPr(p_j \succeq p_{j'})}{\partial\Omega_l} = \sum_{u\in\mathbb{U}}\sum_{p_j\in\mathbb{V}(u)}\frac{\partial\left[log\left(\sum_{j'}^{|C^T|}e^{\mathcal{V}(u_i,p_j')}\right) - \mathcal{V}(u_i,p_j)\right]}{\partial\Omega_l} \tag{14}$$

By using the latent factor based predicted rating UV^T to qualify observed utility, we show the derivation for each point respectively below:

$$\nabla U_i = \frac{\sum_{j'}^{|C^T|}\left(e^{U_iV_{j'}^T}\cdot V_{j'}\right)}{\sum_{j'}^{|C^T|}e^{U_iV_{j'}^T}} - V_j + \lambda U_i \tag{15}$$

$$\nabla V_j = \frac{e^{U_iV_j^T}\cdot U_i}{\sum_{j'}^{|C^T|}e^{U_iV_{j'}^T}} - U_i + \lambda V_j \tag{16}$$

Given a data point, i.e., the collection C^T at time t, the stochastic gradient update rule can be given by:

$$\Omega_l \leftarrow \Omega_l - \eta\nabla\Omega_l \tag{17}$$

where η is the learning rate. Nevertheless, we do not need to update every point at each iteration. Thus we randomly draw a batch of $\succeq^T$ to embed into our learning algorithm. Accordingly, the algorithm is shown in Algorithm 1.

Algorithm 1. Time-aware Comparative Learning.

```
input: ⪰^T, η_0
output : U,V
randomU, V  η ← η_0
for step = 1 to MAX STEP do
    random pick u_i from 𝕌
    random pick p_j from 𝕍(u_i)
    select POI collection C_ij^T
    p_j = max(C_ij^T)
    U_i ← U_i − η∇U_i
    V_j ← V_j − η∇V_j
    if converge then
        return U,V
    η ← η / (1+step/MAXSTEP)
```

5.2. Recommendations

Up till now, after we complete the optimization in Algorithm 1 we have U and V at hand, thus we can estimate the utility $\mathcal{U}_{ij}$ in the same way as those works which borrowing the idea from matrix factorization techniques. Recall the definition of utility, it indicates the satisfaction for a particular user over a POI. We can sort the utility values of POIs for a certain user in an order and recommend the top-K POIs to him. The estimation for the utility of a POI is denoted as:

$$\begin{aligned}\mathbb{E}[\mathcal{U}(ij)] &= \mathbb{E}[\mathcal{V}(ij) + \varepsilon_{ij}] \\ &= \mathcal{V}(ij) + \mathbb{E}(\varepsilon_{ij}) \\ &= UV^T + c\end{aligned} \tag{18}$$

where c is a constant.

5.3. Complexity analysis

The overall complexity is determined by both utility computation and gradient descent process cost.

For the first part, the non-missing values contribute much to the algorithm and the recent POI collection size is dominant for the second part. As shown in Eqs. (8), (14) and Algorithm 1, only the observations R will be involved into the computation. Since we devise the algorithm based on stochastic gradient descent, thus for each observation R_{ij}, the gradient descent process is strongly relevant to the collection size $|C_{ij}^T|$. Thus the integrated computation cost is $\mathcal{O}(|R_{ij}| \cdot |C_{ij}^T|)$. Generally, T is relatively small and the collection size can be deemed as a constant. In all, our approach is computationally efficient and linear with the data set size.

6. Experiment analysis

In this section, we conduct several experiments to compare our model with several methods over IR metrics and further analyze both effectiveness and efficiency in our POI recommendations. We will address the following questions in our experiments:

1. How are our approaches compared with existing methods?
2. What result may be achieved if we vary the size of time window?

6.1. Datasets & setting

Our model is quite general and requires time stamp of the rating as auxiliary information. Hence in this paper, we evaluate our method on two datasets, which are Yelp and TripAdvisor respectively.

The first data source we choose is Yelp.[1] Yelp is a well-known website for business ratings and reviews which have a profound effect on the success of business (Anderson & Magruder, 2012) - an extra half-star rating causes restaurants to sell out 19 percentage more frequently, thus making it an ideal source for our experiments. By filtering out users who have less than 10 ratings from the raw data, we finally obtain 20166 unique users and 39104 merchants with 586274 ratings.

The second dataset we employ for evaluation is TripAdvisor.[2] TripAdvisor[3] is a travel website providing reviews of travel-related content. Likewise to Yelp, we filter out users with less than 8 ratings, thus making the final data set containing 13410 users and 9149 merchants with 152721 ratings.

The statistics of the both data sets are shown in Table 1. Note that in this paper, we only exploit ratings and their corresponding time stamps. The study of other auxiliary information will be investigated in future work.

As discussed in Section 4, the iteration rate η and regularization parameter λ will be tuned through cross validation, which will be detailed in the following section. In order to choose a proper dimension, we vary the dimensionality to watch the performance of Recall on both datasets and finally get Fig. 5. Curves of both datasets demonstrate that with the increasement of dimension the Recall increase as well at first. When it surpasses a threshold, the metric Recall decays. Thus we choose the dimensionality at the peak for both datasets. Our experiments are conducted on a personal computer equipped with an Intel Core2 Duo CPU(2.67 GHz) and 2 Giga byte memory.

6.2. Metrics

POI recommendation aims to recommend personalized top-K POIs to users, which naturally coincides with the process of ranking. Hence we adopt the following metrics, which are commonly used in *Information Retrieval* and *Search Engine*, to evaluate the performance of all models. In our experiments, top 100 POIs are recommended due to the maximum webpage display in real applications.

- Recall@K: the recall of top K recommended POIs for a particular user is defined by:

$$Recall@K = \frac{|rec@K \cap rel|}{|rel|}$$

where $rec@K$ denotes the top K recommended POIs and rel is the true visited POIs in the testing set.

- MAP: Mean Average Precision (MAP), which for a test collection is the arithmetic mean of average precision values for individual user. The definition is as follows:

$$MAP = \frac{1}{N}\sum_{u=1}^{N}\left(\frac{1}{|rel|}\sum_{l=1}^{|rel|} Precision_u@l\right)$$

where $Precision_u@l$ denotes the precision for user u when l relevant POIs are retrieved and N is the total number of users.

6.3. Baseline and comparison

In experiments, the following state-of-the-art methods are evaluated for comparison, including our model.

- **TBCF**: It is a time-based collaborative filtering method, proposed in He and Wu (2009). It incorporates the time-context into the traditional CF algorithm by considering the impact of the user's activities regressively, which means the longer of the activity happens, the weaker impact it has on the latest one.
- **SPLINE**: We adopt this method from Koren (2009), which considers time changing user bias. It is a more flexible parameterization method than linear model to fuse time distance into time changing user bias.

Table 1
Statistics of dataset Yelp and TripAdvisor.

Dataset	Statistics	User	POI
Yelp	Max. Num of Ratings	1234	1189
	Avg. Num of Ratings	29.1	15.0
TripAdvisor	Max. Num of Ratings	96	708
	Avg. Num of Ratings	11.4	16.7

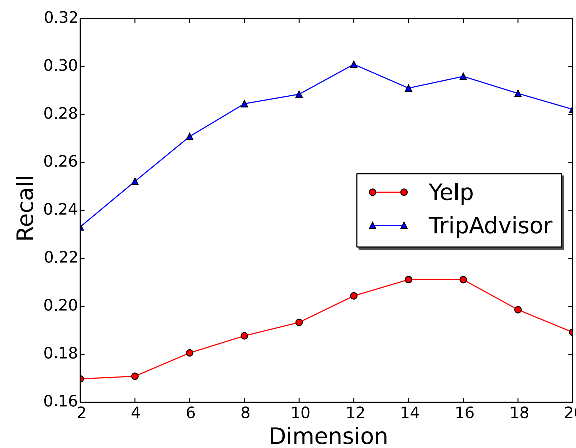
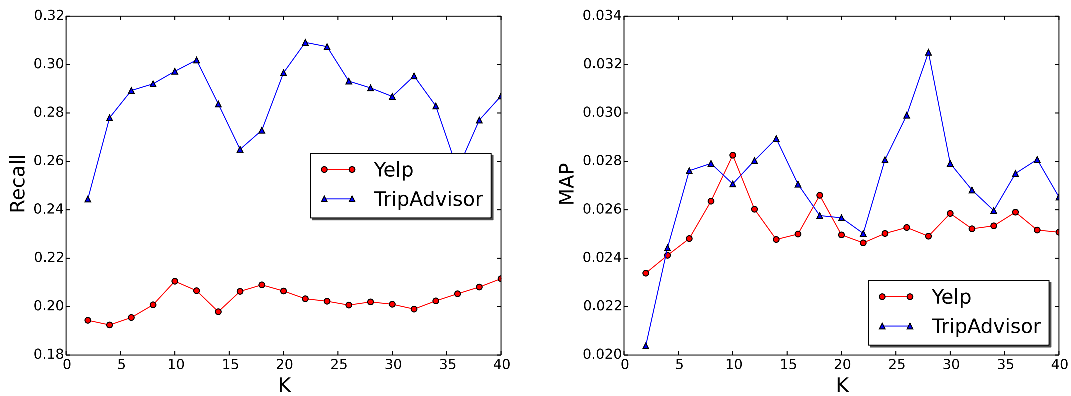


Fig. 5. Dimension choose.

[1] <http://www.yelp.com.au/dataset_challenge>.
[2] <http://sifaka.cs.uiuc.edu/wang296/Data/index.html>.
[3] <http://www.tripadvisor.com/>.

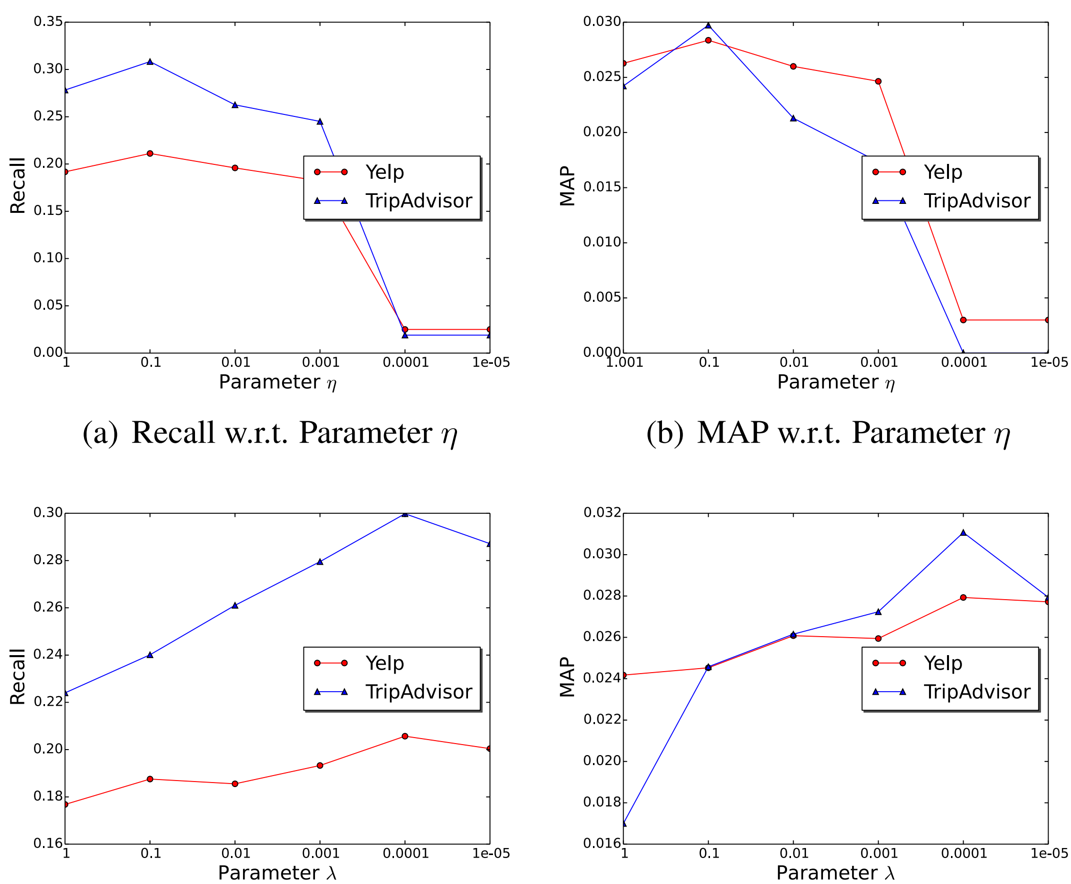
Table 2
Prediction accuracy comparison on both datasets.

Dataset	Metrics	TBCF	SPLINE	PMF	BPR	TCL-K
Yelp	Recall	0.17512	0.18454	0.18755	0.19797	**0.21281**
	improve	21.52%	15.32%	13.47%	7.50%	
	MAP	0.02217	0.02471	0.02559	0.02648	**0.02831**
	improve	27.70%	14.57%	10.63%	6.91%	
TripAdvisor	Recall	0.23941	0.25449	0.27302	0.28462	**0.30701**
	improve	28.24%	20.64%	12.45%	7.23%	
	MAP	0.02681	0.02772	0.02811	0.03072	**0.03254**
	improve	21.37%	17.39%	15.76%	5.92%	



(a) Recall w.r.t. Time Window (b) MAP w.r.t. Time Window

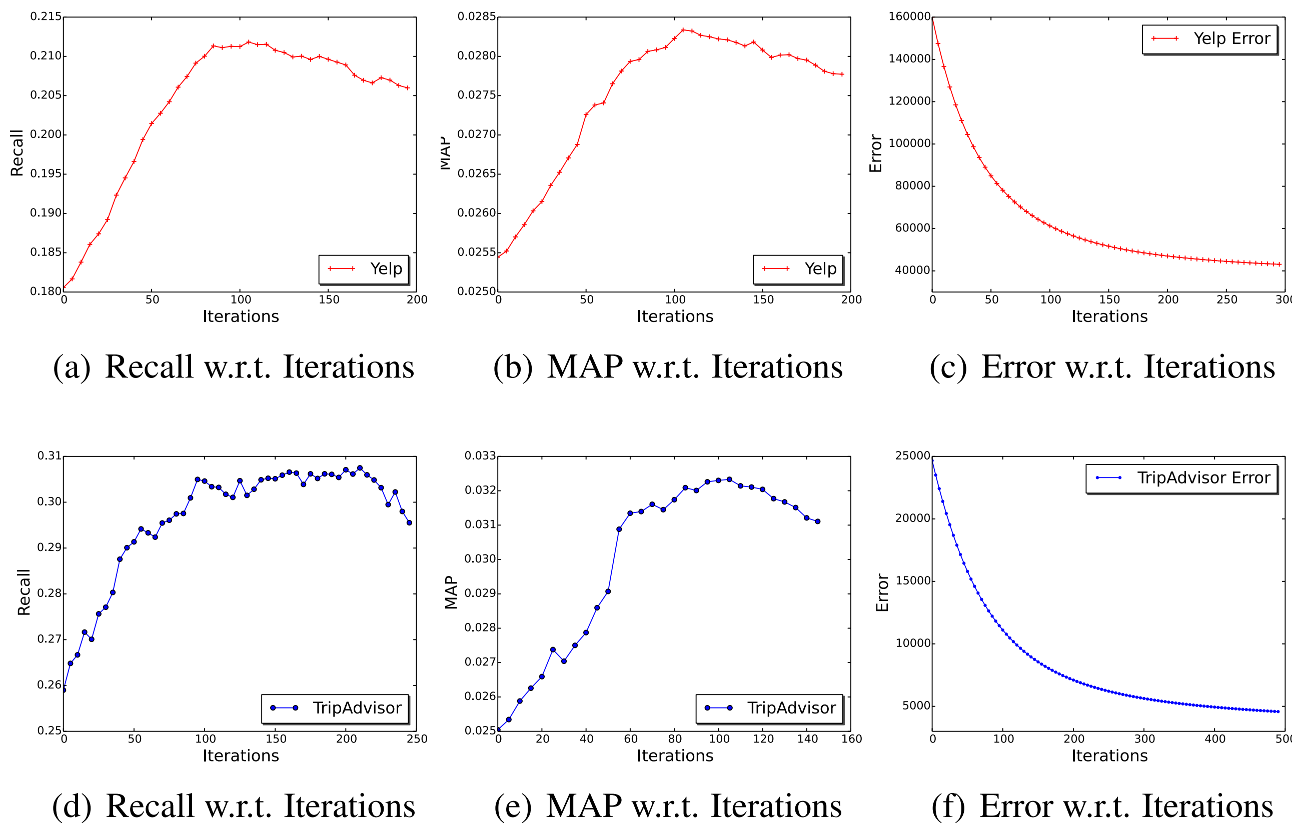
Fig. 6. Impact of time window on both metrics.



(a) Recall w.r.t. Parameter η (b) MAP w.r.t. Parameter η

(c) Recall w.r.t. Parameter λ (d) MAP w.r.t. Parameter λ

Fig. 7. Impact of parameter η and λ on both metrics.



(a) Recall w.r.t. Iterations (b) MAP w.r.t. Iterations (c) Error w.r.t. Iterations

(d) Recall w.r.t. Iterations (e) MAP w.r.t. Iterations (f) Error w.r.t. Iterations

Fig. 8. Convergency analysis.

- **PMF**: This method is proposed in Salakhutdinov and Mnih (2007) by factorizing user-POI matrix, only utilizing the rating matrix. It gives the probabilistic explanation for matrix factorization.
- **BPR**: It was proposed in Rendle et al. (2009) by modeling user preference as a ranking problem. It provides a generic learning algorithm based on stochastic gradient descent with bootstrap sampling.
- **TCL-K**[4]: This is our recency based comparative learning algorithm by taking recent K ratings into consideration.

The experimental results are listed in Table 2. From Table 2, we can observe that among the baseline methods, our approach generally achieve the best performance in both metrics on two datasets. Additionally BPR outperforms PMF, which coincides with Rendle et al. (2009). As to Recall, our method improves the result by 7.50% and 7.23% respectively compared with BPR for dataset Yelp and TripAdvisor. For MAP, the improvements compared to BPR are 6.91% and 5.92%. In overall, the improvements of our approach are significant, which justifies our claim of involvement of visit order within a certain time-window.

One key challenge recommender systems always meet is the sparsity of the data and there is no exception for our method. Known in Section 6.1, Yelp is sparser than that of TripAdvisor. When comparing the results across two datasets in Table 2, metrics' value in TripAdvisor outperform that of Yelp, which in turn conclude that the denser data the better results our method will achieve.

The better results we achieved than existing methods owe to the evaluation metrics we adopt. As is known to all, there are two main categories for metrics in recommendation evaluation: one focuses on real value prediction, i.e., MAE and RMSE, another emphasizes ranking problem in information retrieval, i.e., Recall and MAP. The ranking problem coincides with the learning process in our model which leads us to choose the two evaluation metrics. The chosen metrics favor the order as well, thus making our approach superior to state-of-the-art methods.

6.4. Time window analysis

According to the method we proposed, we analyze the effects with regard to different time windows we utilized. Small K denotes the small size of the time window that implies user will consider less POIs in memory, which is so-called short-term property. In contrast, long-term property suggests larger K.

Fig. 6(a) and (b) plot the performance of both metrics with regard to different K values. In consideration of the granularity of K, we observe several peaks on the line charts, which show the periodicity of user's comparative choices. When user consider quite a small number of past POIs, the short-term is in dominance of the comparative process and it decays with the value K increases. For another extreme example, when a user takes the entire historical records into consideration, the results are not satisfactory as well. Thus a trade-off between short-term and long-term property should be taken. In comparison with the baselines, we generally set the K according to our experiments in order to achieve better results.

6.5. Impact of parameter η and λ

In our method, the parameter η and λ play crucial roles. They control how much the regularization terms should be integrated and how quickly the objective function descent. In the extreme

[4] public code in python: <https://www.dropbox.com/sh/f8usnw19yg7werv/AABlR7ipr4uLbrtUHmlMkDwja?dl=0>.

case, if we set η to a tiny value, it means every step of the gradient descent will be performed slowly, however it yields unpredicted results in the experiments. On the other hand, it could possibly miss the minimum point if we set η to a certain large value. From the Fig. 7(a) and (b), we conclude that the results firstly increase with the decrease of η however they decay when η surpasses a certain value, i.e., 0.1 in our experiments according to two datasets we utilized.

Parameter λ has the same property as well, which is shown in 7(c) and (d). By tuning the parameter, we achieve the peak when λ is set to 0.0001, which becomes the global setting for our method.

6.6. Iteration analysis

As theoretically discussed in Section 5.3, our method is computationally efficient. In this part, we plot two results in terms of the total error term by measuring how they vary with the increment of the iteration step. The total error term is calculated as the sum of differences between the real value and the predicted value across all the test dataset.

From the Fig. 8(f) and (c), we can see that the error will converge in about 500 iteration rounds with each round costs no more than 2 s, thus making the total consuming time limited to 10–20 min. For Recall and MAP, both of them increase at first rapidly as the iteration step increase. At around 100–150 iterations, they reach the peak and after that both drop. This is mainly because the overfitting problem, which means that more iteration steps lead to worse ranking prediction. This is obvious when comparing the figures horizontally on the same dataset. In summary, our approach is able to converge in a limited number of iterations.

7. Conclusion and future work

In this paper, we focus on POI recommendation by exploring the ratings and their corresponding time stamps on LBSN. Different from conventional methods, we aim to model user rating behavior for learning user preferences through exploring the comparative choices within a certain period. We make several contributions: A novel approach is proposed by employing choice model deduced from utility theory to model user preference. We devise a collection-wise learning method over partial orders through an effective stochastic gradient descent algorithm. Experiments conducted on two real world datasets have demonstrated that our approach outperform existing methods. Moreover, the computational cost of our algorithm is linear to the data set size, which indicates that it can be applied to a large database.

Our method suffers the same sparsity problem most recommender systems meet. Moreover, it is sensitive to the size of time window, which implies the model needs to tune the time-window for every dataset.

In our current work only the ratings and their corresponding time are utilized, leaving abundant information unexplored like social networks, geographical location. In future work, we aim to take into other contextual information, especially geographical location, into investigation.

As mobile connects both physical and real world, POI recommendation stands out especially in online social networks. By the better annotation of user rating behavior, we can provide better POI recommendations.

Acknowledgement

This research was partially supported by grants from the National Science Foundation for Distinguished Young Scholars of China (Grant No. 61325010), the National High Technology Research and Development Program of China (Grant No. 2014AA015203), and National Center for International Joint Research on E-Business Information Processing (2013B01035).

References

Anderson, M. L., & Magruder, J. R. (2012). Learning from the crowd: Regression discontinuity estimates of the effects of an online review database. *Economic Journal, 122*(563), 957–989.

Cheng, C., Yang, H., King, I., & Lyu, M.R. (2012). Fused matrix factorization with geographical and social influence in location-based social networks. In: *Proceedings of the twenty-sixth AAAI conference on artificial intelligence* (p. 1). July 22-26, 2012, Toronto, Ontario, Canada.

Cho, E., Myers, S.A., & Leskovec, J. (2011). Friendship and mobility: user movement in location-based social networks. In: *Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1082–1090). San Diego, CA, USA, August 21-24, 2011.

Deng, S., Huang, L., & Xu, G. (2014). Social network-based service recommendation with trust enhancement. *Expert Systems Applied, 41*(18), 8075–8084.

Gao, H., Tang, J., Hu, X., & Liu, H. (2013). Exploring temporal effects for location recommendation on location-based social networks. In: *Seventh ACM conference on recommender systems, RecSys '13* (pp. 93–100). Hong Kong, China, October 12-16, 2013.

Greene, W.H., & Hensher, D.A. (2008). Modeling ordered choices: A primer and recent developments. Tech. Rep. 08-26.

He, L., & Wu, F. (2009). A time-context-based collaborative filtering algorithm. In: *The 2009 IEEE international conference on granular computing* (pp. 209–213). GrC 2009, Lushan Mountain, Nanchang, China, 17-19 August 2009.

Hong, W., Li, L., & Li, T. (2012). Product recommendation with temporal dynamics. *Expert Systems Applied, 39*(16), 12398–12406.

Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. In: *Proceedings of the 8th IEEE international conference on data mining (ICDM 2008)* (pp. 263–272). December 15-19, 2008, Pisa, Italy.

Hyung, Z., Lee, K., & Lee, K. (2014). Music recommendation using text analysis on song requests to radio stations. *Expert Systems Applied, 41*(5), 2608–2618.

Jamali, M., & Ester, M. (2009). TrustWalker: a random walk model for combining trust-based and item-based recommendation. In: *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 397–406). Paris, France, June 28–July 1, 2009.

Koren, Y. (2009). Collaborative filtering with temporal dynamics. In: *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 447–456). Paris, France, June 28–July 1, 2009.

Koren, Y., Bell, R. M., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *IEEE Computer, 42*(8), 30–37.

Linden, G., Smith, B., & York, J. (2003). Industry report: Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Distributed Systems Online, 4*(1).

Liu, B., & Xiong, H. (2013). Point-of-interest recommendation in location based social networks with topic and location awareness. In: *Proceedings of the 13th SIAM international conference on data mining* (pp. 396–404). May 2-4, 2013. Austin, Texas, USA.

Liu, B., Fu, Y., Yao, Z., & Xiong, H. (2013). Learning geographical preferences for point-of-interest recommendation. In: *The 19th ACM SIGKDD international conference on knowledge discovery and data mining, KDD 2013* (pp. 1043–1051). Chicago, IL, USA, August 11-14, 2013.

Li, L., Zheng, L., Yang, F., & Li, T. (2014). Modeling and broadening temporal user interest in personalized news recommendation. *Expert Systems Applied, 41*(7), 3168–3177.

Neumann, J., & Morgenstern, O. (2007). *Theory of games and economic behavior*. Princeton University Press.

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: bayesian personalized ranking from implicit feedback. In: *UAI 2009, Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence* (pp. 452–461). Montreal, QC, Canada, June 18-21, 2009.

Salakhutdinov, R., & Mnih, A. (2007). Probabilistic matrix factorization. In: *Advances in neural information processing systems 20, proceedings of the twenty-first annual conference on neural information processing systems* (pp. 1257–1264). Vancouver, British Columbia, Canada, December 3-6, 2007.

Thurstone, L. L. (1927). A law of comparative judgement. *Psychological Review, 34*, 278–286.

Train, K. E. (2009). *Discrete choice methods with simulation* (2nd ed.). Cambridge University Press.

Xiang, L., Yuan, Q., Zhao, S., Chen, L., Zhang, X., Yang, Q., & Sun, J. (2010). Temporal recommendation on graphs via long- and short-term preference fusion. In: *Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 723–732). Washington, DC, USA, July 25–28, 2010.

Yang, S.-H., Long, B., Smola, A.J., Zha, H., & Zheng, Z. (2011). Collaborative competitive filtering: learning recommender using context of user choice. In: *Proceeding of the 34th International ACM SIGIR conference on research and development in information retrieval, SIGIR 2011* (pp. 295–304). Beijing, China, July 25–29, 2011.

Ye, M., Yin, P., & Lee, W.-C., 2010. Location recommendation for location-based social networks. In: *Proceedings 18th ACM SIGSPATIAL international symposium* *on advances in geographic information systems, ACM-GIS 2010* (pp. 458–461). November 3–5, 2010, San Jose, CA, USA.

Ye, M., Janowicz, K., Mülligann, C., & Lee, W.-C., 2011a. What you are is when you are: the temporal dimension of feature types in location-based social networks. In: *Proceedings 19th ACM SIGSPATIAL international symposium on advances in geographic information systems, ACM-GIS 2011* (pp. 102–111). November 1–4, 2011, Chicago, IL, USA.

Ye, M., Yin, P., Lee, W.-C., & Lee, D.L. (2011b). Exploiting geographical influence for collaborative point-of-interest recommendation. In: *Proceeding of the 34th International ACM SIGIR conference on research and development in information retrieval, SIGIR 2011* (pp. 325–334). Beijing, China, July 25–29, 2011.

Yin, H., Sun, Y., Cui, B., Hu, Z., & Chen, L. (2013). Lcars: A location-content-aware recommender system. In: *The 19th ACM SIGKDD international conference on knowledge discovery and data mining, KDD 2013* (pp. 221–229). Chicago, IL, USA, August 11-14, 2013.

Yuan, Q., Cong, G., Ma, Z., Sun, A., & Magnenat-Thalmann, N. (2013). Time-aware point-of-interest recommendation. In: *The 36th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR '13* (pp. 363–372). Dublin, Ireland - July 28 - August 01, 2013.

Zheng, N., & Li, Q. (2011). A recommender system based on tag and time information for social tagging systems. *Expert Systems Applied, 38*(4), 4575–4587.