

# 一种基于矩阵分解的上下文感知 POI 推荐算法

彭宏伟 靳远远 吕晓强 王晓玲

(华东师范大学教育部可信软件国际合作联合实验室 上海 200062)

**摘要** 近年来,随着移动设备的普及,基于位置的社交网络(Location-Based Social Network, LBSN)逐渐被人们广泛使用并成为一种新型的社交媒体。LBSN 能够记录丰富的上下文信息,例如用户社交网络、POI 地理位置、POI 类别信息等,这无疑为个性化的 POI(Point-of-Interest)推荐系统带来了巨大的发展机遇。但是如何建模这些上下文信息对 POI 推荐的影响并将它们有效地融合成为了一大难点,另外用户签到数据的稀疏性也为 POI 推荐带来巨大的挑战。为了克服上述挑战,该文提出了一个基于矩阵分解的上下文感知 POI 推荐模型。具体地,该文从多个方面建模用户的签到行为,除了利用用户的签到数据,还考虑了 POI 的地理位置对用户签到行为的影响,用户更愿意访问那些距离近并且符合自身偏好的 POI。另外,为了进一步缓解签到数据的稀疏性,该文还利用了用户社交网络数据和 POI 类别信息。最后,该文提出了一个通用的矩阵分解模型,它能有效地融合上述上下文信息,并且具有良好的可扩展性和较低的时间复杂度。在两个真实的 LBSN 数据集上的实验结果表明,该文提出的方法在推荐的准确性上远优于当前流行的 POI 推荐算法。

**关键词** 基于位置的社交网络;兴趣点;推荐系统;矩阵分解;上下文感知

**中图法分类号** TP311 **DOI号** 10.11897/SP.J.1016.2019.01797

## Context-Aware POI Recommendation Based on Matrix Factorization

PENG Hong-Wei JIN Yuan-Yuan LV Xiao-Qiang WANG Xiao-Ling

(MOE International Joint Lab of Trustworthy Software, East China Normal University, Shanghai 200062)

**Abstract** With the popularity of mobile devices, Location-based Social Network (LBSN) has been widely used and becomes a new form of social media in recent years. LBSN can record rich context information, such as social networks, geographical information, POI category information, etc. The context information provides a great opportunity to build personalized POI (Point-of-Interest) recommender systems. In POI recommendation, users' check-in behavior is not only affected by the users' own preferences, but also influenced by various context information in the surrounding environments. Therefore, how to model the influence of the context information on the users' check-in behavior and effectively integrate them with the users' preferences becomes a major difficulty. In addition, the number of users and POIs in the LBSN often reaches millions, which makes the user-POI check-in matrix very large and sparse. The sparsity of user's check-in data also poses a huge challenge for POI recommendation. In this paper, we propose a context-aware POI recommendation model based on matrix factorization, which addresses the above challenges at some extent. Specifically, we attempt to model the users' check-in behavior from multiple aspects. Firstly, we use matrix factorization technology to learn users' own preferences from the users' check-in data ( $U$ ), and consider the influence of POI category information ( $C$ ) on the

收稿日期:2017-09-27;在线出版日期:2018-04-23。本课题得到国家“八六三”高技术研究发展计划项目(2017YFC0803700)、国家自然科学基金(61532021,61472141)、上海市可信物联网软件协同创新中心(中心代号:ZF1213)、上海市科技兴农重点攻关项目(沪农科攻字(2016)第2-1号)资助。彭宏伟,硕士,主要研究方向为数据挖掘、推荐系统。E-mail: penghongwei\_phw@163.com。靳远远,博士,主要研究方向为推荐系统、机器学习。吕晓强,硕士,主要研究方向为数据挖掘、推荐系统。王晓玲(通信作者),博士,教授,博士生导师,主要研究领域为数据管理技术、数据服务。E-mail: xlwang@sei.ecnu.edu.cn。

users' preferences, since users often prefer a certain type of POI rather than a specific POI. Secondly, POI's geographical location (G) has a great impact on user's check-in behavior. Users prefer to visit POIs that are closer and meet their own preferences. We use kernel functions to model the distance distribution between any POI pair, and then use item based collaborative filtering to calculate the users' preferences for POIs in terms of geographical location. Thirdly, taking into account that user's check-in behavior may be influenced by friends, we use user-based collaborative filtering to model the user social network (S) on the users' check-in behavior, and further relieve the sparsity of the check-in data. Finally, we propose a general matrix factorization model UCGSMF. When modeling users' check-in behavior with the influence of context information and their own preferences, different context information strategies are applied to visited POIs and non-visited POIs. In this way, users' own preferences can be better fitted. What's more, the model has good extensibility, and it is very flexible in the modeling of context information. At the same time, by adopting an improved alternating least-squares algorithm, the model has a lower time complexity. In this paper, a large number of experiments are conducted on Dianping and Foursquare datasets. First, we analyze the recommendation performance under different algorithms. Precision and recall are used to evaluate the performance of the algorithms. Experimental results show that the recommendation performance of our model is much better than the state of the art POI recommendation algorithms. Then, considering the important influence of context information on POI recommendation, we analyze the performance of different context information. The experimental results show that the POI category information factor can indeed improve the POI recommendation performance when a suitable value is obtained. Compared with social network information, geographical location information has higher impact on POI recommendation. Finally, we compare the training time of different algorithms. The experimental results verify that our model has a lower time complexity.

**Keywords** location-based social network; point-of-interest; recommender system; matrix factorization; context-aware

## 1 引言

近年来,随着智能手机和无线网络的普及,人们能够更加便捷地获取自己的实时位置,这极大地方便了出行,同时也让基于位置的社交网络(Location-Based Social Network, LBSN)得到广泛的应用,例如大众点评、Foursquare、Gawalla等。图1展示了常

见的基于位置的社交网络,用户利用 LBSN 去发现自己感兴趣的 POI (Point-of-Interest) (如餐厅、电影院等),同时可以对这些 POI 进行签到、评论等操作,还能分享给自己的朋友。可以看到,除了用户的签到数据,LBSN 还包含着许多上下文信息,包括用户社交网络以及 POI 的各种信息,例如经纬度、类别信息等。这些丰富的上下文信息无疑为个性化的 POI 推荐提供了巨大的发展机遇。

POI 推荐旨在给用户推荐那些符合他们兴趣爱好但其还未访问过的 POI。对用户来说,POI 推荐系统可以帮助用户发现自己潜在感兴趣的 POI;另一方面,对商家而言,他们可以通过 POI 推荐系统发现那些对特定 POI 感兴趣的潜在用户群,从而制定相应策略来增加盈利。

尽管当前存在很多针对 POI 推荐的研究,但是个性化的 POI 推荐系统仍然存在许多问题。首先,LBSN 中用户的签到数据是非常稀疏的。在现实的 LBSN 中,POI 的数量十分庞大而且分布广泛,用户

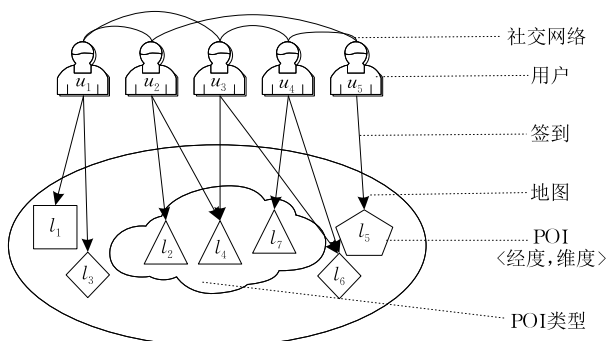


图1 常见的基于位置的社交网络<sup>[1]</sup>

访问过并进行签到的 POI 往往非常少,这就使得用户-POI 签到矩阵十分稀疏,签到数据的极度稀疏无疑给 POI 推荐带来了很大的困难。

另一方面,用户的签到行为背后是一个复杂的决策过程.除了用户自身的兴趣偏好,它还可能受其他因素影响,不同因素的影响程度也是不一样的,如何去建模用户的决策过程是件非常困难的事.首先,根据地理学第一定律(Tobler's First Law of Geography):所有事物都和其他事物相关,相邻的事物更加相似,远离的事物更为相异<sup>[2]</sup>.因此,用户的决策行为往往受到 POI 的地理位置影响,用户更偏向于访问那些距离较近的 POI.其次,最近的研究表明朋友之间往往有着共同的兴趣爱好,因此用户的社交网络也会影响用户的签到行为<sup>[3-4]</sup>.另外,POI 的一些其他上下文信息,例如 POI 的类别信息,也会对用户的决策行为产生影响.比如一个口味偏辣的用户在选择餐厅时可能更偏好川菜馆或湘菜馆.因此,如何去建模这些上下文信息对用户决策的影响以及将这些上下文信息有效地融合是一个研究重点。

为了解决上述的问题,本文提出了一个基于矩阵分解的上下文感知 POI 推荐模型.本文考虑从以下几个方面去建模用户签到行为:(1)用户自身偏好;(2)地理位置信息;(3)社交网络信息.首先考虑地理位置信息对用户签到行为的影响.本文利用核函数去建模任意 POI 之间的距离分布,然后利用基于物品的协同过滤方法计算地理位置信息对用户签到行为的影响程度.类似地,通过基于用户的协同过滤方法可得到社交网络信息对用户签到行为的影响程度.在计算用户之间的相似度时,本文假设那些具有更多共同签到的 POI 或签到 POI 的类别有着更大重合度的用户的相似度更高.最后,本文提出了一个通用的矩阵分解模型去建模用户自身偏好,该模型能够有效地融合用户自身偏好以及各种上下文信息以更好地拟合用户签到行为,从而有效地提高 POI 推荐的准确率。

本文主要贡献包括以下三点:

(1)本文提出一个通用的融合上下文信息的矩阵分解模型.该模型在融合用户自身偏好和上下文信息以建模用户签到行为时,对用户访问过的 POI 和未访问过的 POI 采用不同的融合上下文信息策略,能更有效地刻画用户自身偏好,并且具有良好的可扩展性。

(2)本文从用户自身偏好、地理位置信息、社交网络信息等多方面建模用户的签到行为,有效地缓

解了签到数据的稀疏性。

(3)本文在两个真实的 LBSN 数据集上进行了充分的实验,实验结果表明本文提出的方法在推荐的准确性上远优于其他 POI 推荐算法,并且具有较低的时间复杂度。

本文第 2 节介绍本文相关工作;第 3 节建模用户签到行为,包括地理位置信息、社交网络信息和用户自身偏好对用户签到行为的影响;第 4 节介绍本文提出的基于矩阵分解的上下文感知 POI 推荐模型;第 5 节通过实验验证本文提出的 POI 推荐方法的性能;第 6 节对本文工作进行总结。

## 2 相关工作

目前,基于 LBSN 的个性化 POI 推荐被广泛研究,大部分的 POI 推荐主要是利用用户的签到数据并结合各种上下文信息来进行推荐。

一类常用的上下文信息就是地理位置信息<sup>[1,5-14]</sup>.例如,Ye 等人<sup>[1]</sup>发现用户访问过的 POI 存在空间聚类现象并使用一个幂律分布去刻画 POI 之间的距离,用户对 POI 的签到行为由用户自身偏好、地理位置信息和社交网络信息三个方面线性组合来刻画.与采用幂律分布假设不同,Zhang 等人<sup>[5]</sup>使用核密度估计(kernel density estimation)去估计任意两个 POI 之间的距离分布,以避免采用特定分布拟合可能带来的误差.Cheng 等人<sup>[6]</sup>则发现用户签到行为具有多个中心,因此采用一个多中心的高斯模型去建模地理位置信息的影响.除此之外,作者还考虑将矩阵分解模型与地理位置信息和社交网络信息相结合来进行 POI 推荐.但是这种模型缺乏对影响用户签到行为的因素的综合考虑,并且容易过拟合<sup>[7]</sup>.Lian 等人<sup>[8]</sup>提出了 GeoMF 模型,将地理位置信息融合到加权矩阵分解中.该模型给非零签到数据分配一个较大的权重,而给为零的签到数据分配一个较小的权重.与 GeoMF 模型不同的是,Li 等人<sup>[9]</sup>提出一个基于逐对排序的 rank-geoFM 模型,该模型根据用户的签到数据对 POI 进行逐对排序,对每个用户来说,有两种偏序关系:签到次数较多的 POI 排在签到次数较少的 POI 前面,观察到的 POI 排在未观察到的 POI 前面,通过最小化模型预测出的偏序关系与真实数据中偏序关系的不一致性来求解模型.而 Liu 等人<sup>[10]</sup>考虑了两种地理位置特征:单个 POI 级别和区域级别.具体来说,对单个 POI,用户对它的偏好受用户自身兴趣及其周边的 POI 影响;

而对区域级别的 POI, 则采用 group lasso 来建模 POI 的区域特征, 即属于同一区域的 POI 对应的隐向量具有相似的稀疏结构. 最近工作中, Yang 等人<sup>[14]</sup> 提出一个融合协同过滤和基于上下文网络图的半监督学习方法的框架 PACE, 其中协同过滤技术用来学习用户偏好, 而半监督学习方法则用来建模 POI 上下文信息(如地理位置信息)和用户上下文信息(如社交网络信息), 最后通过神经网络嵌入(Neural embedding)将两者有效地组合在一起.

除了地理位置信息, 社交网络信息<sup>[3-4, 15-20]</sup> 对 POI 推荐也有比较大的影响. 一些研究<sup>[1, 15]</sup> 采用基于用户的协同过滤方法, 结合朋友的签到信息来预测用户对尚未访问过的 POI 的偏好程度. 为了更好地将社交网络信息融合到协同过滤中, Noulas 等人<sup>[3]</sup> 将个性化的随机游走算法应用在用户-POI 二分图和社交网络图上, 然后结合矩阵分解来进行推荐. Ma 等人<sup>[4]</sup> 假设朋友之间可能具有相似的偏好并利用矩阵分解建模用户偏好, 然后将社交网络影响转化成一个正则化项去约束目标函数. Gao 等人<sup>[16]</sup> 通过地理位置与社交网络的关系建模了四种社会关系, 而用户签到概率则通过这四种关系的线性组合得到. 最近的工作中, Li 等人<sup>[17]</sup> 定义了三类朋友关系: 具有社交联系的社交朋友、与用户具有共同签到记录的位置朋友以及与用户居住地相对较近的近邻朋友. 通过这三类朋友来挖掘用户潜在感兴趣的 POI, 在采用矩阵分解模型的基础上提出了两种不同的损失函数去优化模型.

除了研究地理位置信息和社交网络信息对 POI 推荐的影响, 也有很多利用其他上下文信息进行 POI 推荐的研究. 文本信息是一类重要的上下文信息, 它记录着用户对 POI 的评论信息或标签信息等, 一些研究考虑了文本信息对 POI 推荐的影响<sup>[21-23]</sup>; 时间信息对用户选择 POI 有着深远的影响, 用户访问 POI 在时间上往往遵循着一定的规律, 一些研究建模了时间对 POI 推荐的影响<sup>[9, 24-25]</sup>; 另外, Liu 等人<sup>[26]</sup> 考虑了 POI 类别信息对 POI 推荐的影响, Wang 等人<sup>[27]</sup> 则使用 CNN 技术抽取图片的特征向量融合到矩阵分解中来提升推荐效果. 除此之外, 还有一些研究关注 POI 推荐领域中的一些特定场景, 例如路线规划或行程推荐问题<sup>[28-30]</sup>、异地 POI 推荐问题<sup>[31-32]</sup>、餐厅推荐<sup>[33]</sup> 等, 本文主要关注在传统的 POI 推荐上.

另外可以看到, 大部分研究只考虑结合部分上下文信息来进行 POI 推荐, 而考虑了多种上下文信

息的研究或者只是通过简单的协同过滤将上下文信息进行线性组合<sup>[1]</sup>, 或者通过矩阵分解简单地融合上下文信息<sup>[6, 8, 10, 17]</sup>, 这些都无法全面地建模用户的签到行为. 因此本文提出一个能有效融合多种上下文信息的矩阵分解模型来解决现有工作中的问题.

### 3 用户签到行为建模

#### 3.1 问题描述

POI 推荐的任务是给用户推荐一个其感兴趣但还未访问过的 POI 列表. 例如在用户旅游时给用户推荐其可能感兴趣的旅游景点. 个性化 POI 推荐是根据用户的历史签到数据, 结合用户和 POI 的上下文相关信息进行推荐. 表 1 是本文用到的基本符号的描述.

表 1 基本符号描述

符号	符号描述
$U, I, T$	用户集合, POI 集合, POI 类别集合
$u, i, t$	用户 $u \in U$ , POI $i \in I$ , POI 类别 $t \in T$
$K$	隐藏特征矩阵的维度
$R = [r_{ui}]$	$R \subseteq  U  \times  I $ , 用户对 POI 的签到次数矩阵
$C = [c_{ui}]$	$C \subseteq  U  \times  I $ , 签到矩阵 $R$ 的 0,1(二值化)矩阵
$W = [w_{ui}]$	$W \subseteq  U  \times  I $ , 签到矩阵 $R$ 对应的权重矩阵
$E = [e_{ui}]$	$E \subseteq  U  \times  I $ , 用户对 POI 在上下文信息上偏好程度矩阵
$G = [g_{ui}]$	$G \subseteq  U  \times  I $ , 用户对 POI 在地理位置上的偏好程度矩阵
$S = [s_{ui}]$	$S \subseteq  U  \times  I $ , 用户对 POI 在社交网络上的偏好程度矩阵
$P$	$P \subseteq  U  \times K$ , 用户隐藏特征矩阵
$Q$	$Q \subseteq  I  \times K$ , POI 隐藏特征矩阵
$D$	$D \subseteq  T  \times K$ , POI 类别隐藏特征矩阵
$p_u$	$P$ 中用户 $u$ 对应的隐藏特征向量
$q_i$	$Q$ 中 POI $i$ 对应的隐藏特征向量
$d_t$	$D$ 中 POI 类别 $t$ 对应的隐藏特征向量

#### 3.2 地理位置信息对用户签到行为的影响

在 LBSN 中, 用户的行为与地理位置息息相关, 用户的决策行为很大程度上受到地理位置信息的影响, 这也是 POI 推荐与其他推荐系统不同的地方. 例如, 用户在选择餐厅时往往选择距离自身位置更近的餐厅, 甚至有时候为了方便用户宁愿选择不是很符合自身偏好但是距离很近的餐厅. 另外, 根据地理学第一定律(Tobler's First Law of Geography): 所有事物都和其他事物相关, 相邻的事物更加相似, 远离的事物更为相异<sup>[2]</sup>. 所以相邻的 POI 往往更相似, 呈现出一种空间聚类现象, 而且用户更偏向于访问那些离自身更近的 POI, 大量研究工作也表明地理位置信息能够帮助提高 POI 推荐的准确率<sup>[1, 6, 17, 25]</sup>.

为了建模地理位置信息对用户签到行为的影

响,本文首先使用 Triangular 核函数去计算 POI 之间的相关程度.相比于传统的相似度计算方法(例如 cosine 相似度),核函数具有无参估计、能更好地模拟 POI 之间的距离分布等优点,而相比于其他核函数(例如 Gaussian 核函数),Triangular 核函数能够过滤掉那些相关程度较低的 POI 对,同时具有和其他核函数相似的拟合效果.具体地,本文使用  $KE_b(i, j)$  计算任意两个 POI  $i$  和  $j$  之间的相似度:

$$KE_b(i, j) \propto \left(1 - \frac{d(i, j)}{b}\right) \mathbf{I}_{\{d(i, j) \leq b\}} \quad (1)$$

其中  $d(i, j)$  表示 POI  $i$  和 POI  $j$  之间的距离.本文利用 POI 的经纬度信息,通过球面半正矢公式计算 POI 之间的距离. $b$  是核函数的宽度参数,  $\mathbf{I}$  是指示函数.可以看到,当 POI 之间的距离超过  $b$  时,它们的核函数值为 0;当 POI 之间的距离小于  $b$  时, POI 之间的距离越近核函数值越大.

本文定义偏好矩阵  $\mathbf{G}$ ,表示用户对 POI 在地理位置上的偏好程度,以此来量化地理位置信息对用户签到行为的影响程度.本文采用基于物品的协同过滤方法来获得偏好矩阵  $\mathbf{G}$ .具体地,通过考虑候选 POI  $i$  与用户  $u$  访问过的 POI 集合之间的相似度,就能获取用户  $u$  对候选 POI  $i$  在地理位置信息上的偏好程度  $g_{ui}$ :

$$g_{ui} = \frac{\sum_{j \in \mathbf{R}_u} KE_b(i, j) \cdot r_{uj}}{\sum_{j \in \mathbf{R}_u} r_{uj}} \quad (2)$$

其中  $\mathbf{R}_u$  表示用户  $u$  访问过的 POI 集合,  $r_{uj}$  是用户  $u$  访问 POI  $j$  的次数.

### 3.3 社交网络信息对用户签到行为的影响

不可否认的是,用户的行为会受到朋友的影响.例如,两个好朋友可能会一起去吃饭、看电影,或者一个用户在去过一家很不错的餐厅后可能推荐给自己的朋友,这些都可能会导致朋友在相同的 POI 产生签到行为.所以,考虑用户的社交网络对用户签到行为的影响应该能提高 POI 推荐的准确率.但是 Cheng 等人<sup>[6]</sup>通过在真实数据上进行对比实验分析发现,社交网络对用户签到行为的影响其实很有限,但它仍然是不可忽略的一个因素.事实上,本文在后面实验中也验证了这点,社交网络信息在影响用户签到行为的所有因素中所占的比重并不大,但是确实能提高 POI 的推荐效果.

本文使用一种基于朋友的协同过滤方法<sup>[15]</sup>来刻画社交网络信息对用户签到行为的影响程度,用偏好矩阵  $\mathbf{S}$  表示,具体计算公式如下:

$$s_{ui} = \frac{\sum_{m \in \mathbf{F}_u} SI_{u,m} \cdot c_{mi}}{\sum_{m \in \mathbf{F}_u} SI_{u,m}} \quad (3)$$

$s_{ui}$  表示用户  $u$  对 POI  $i$  在社交网络上的偏好程度.其中,  $\mathbf{F}_u$  表示用户  $u$  的朋友集合,  $c_{mi}$  表示用户  $m$  对 POI  $i$  的访问情况,  $c_{mi} = 1$  表示用户  $m$  签到过 POI  $i$ ,  $c_{mi} = 0$  表示用户  $m$  未访问过 POI  $i$ .  $SI_{u,m}$  表示用户  $u$  和朋友  $m$  之间的相似度,不同研究<sup>[1,15]</sup>在建模这个相似度上采用的方法不同.

本文从两个方面考虑用户与朋友的相似度.一方面,我们认为朋友之间共同签到的 POI 越多,他们的兴趣爱好应该更相似.但是在很多情况下,用户与朋友并不在一个地区,即使他们的兴趣爱好相似,但是在相同的 POI 签到的概率却很小.在这种情况下,我们认为用户之间访问过相同的 POI 类别就具有一定的相似性,因为 POI 类别刻画了这类 POI 共有的特性.所以用户与朋友访问过的 POI 类别重合度越高,两者就会越相似.基于这两点,我们得到用户与朋友之间的相似度计算公式:

$$SI_{um} = \frac{|\mathbf{R}_u \cap \mathbf{R}_m|}{|\mathbf{R}_u \cup \mathbf{R}_m|} + \eta \frac{|\mathbf{CA}_u \cap \mathbf{CA}_m|}{|\mathbf{CA}_u \cup \mathbf{CA}_m|} \quad (4)$$

本文使用 Jaccard 相似度计算用户与朋友之间在签到 POI 和 POI 类别上的相似度.  $\mathbf{R}_u$  和  $\mathbf{CA}_u$  分别表示用户  $u$  签到的 POI 集合和签到 POI 所属的类别集合.  $\eta$  是权重因子,用来权衡共同签到的 POI 与共同签到的 POI 类别对用户之间相似度的影响程度比重.  $\eta = 1$  表示两者的影响程度是一样的,  $\eta < 1$  ( $\eta > 1$ ) 表示共同签到的 POI 对建模用户之间的相似度的影响程度更大(小).

### 3.4 基于矩阵分解的用户偏好建模

尽管地理位置信息、社交网络信息等上下文信息都能影响用户签到行为,但是决定用户签到行为最主要的因素是用户自身偏好.用户自身偏好可以从签到数据中发掘出来,而矩阵分解技术是一种常用的方法.但是签到数据是一种隐式数据集,海量的用户和 POI 导致签到矩阵十分庞大且稀疏,无法用一般的矩阵分解技术进行分解. Hu 等人<sup>[34]</sup>首次在大规模隐式数据上应用协同过滤技术并提出了加权矩阵分解 (Weighted Matrix Factorization, WMF) 模型.该模型给签到矩阵中缺失项分配一个较小的权重,而给非零项分配一个较大的权重,并通过一种改进的交替最小二乘方法 (Alternating Least Square, ALS) 来降低模型训练的复杂度.该模型目标函数定义如下:

$$J = \sum_{u=1}^{|U|} \sum_{i=1}^{|I|} \omega_{ui} (c_{ui} - \hat{r}_{ui})^2 + \lambda_p \sum_{u=1}^{|U|} \|\mathbf{p}_u\|^2 + \lambda_q \sum_{i=1}^{|I|} \|\mathbf{q}_i\|^2 \quad (5)$$

其中  $\mathbf{p}_u, \mathbf{q}_i$  分别表示用户  $u$  和 POI  $i$  的隐藏特征向量,  $\hat{r}_{ui} = \mathbf{p}_u^T \mathbf{q}_i$  表示用户  $u$  对 POI  $i$  偏好程度.  $\lambda_p$  和  $\lambda_q$  是正则化项系数.  $\omega_{ui}$  对应签到次数  $r_{ui}$  上的权重, 本文采用了与 WMF 中类似的权重设置方法:

$$\omega_{ui} = 1 + \log(1 + 10^\epsilon \cdot r_{ui}) \quad (6)$$

这种方式保证了所有的用户-POI 对都有一个非零的权重, 而随着用户访问 POI 的频率的增加, 其权重也相应增加. 其中  $\epsilon$  用来控制权重随签到次数增长的速率.

WMF 通过  $\mathbf{P}^T \mathbf{Q}$  建模了用户对 POI 的偏好, 但是无法建模用户对一类 POI 的兴趣偏好. 实际上用户会经常访问那些属于同一类别的 POI. 例如, 口味偏辣的用户在选择餐馆时可能更偏向川菜馆或湘菜馆, 而不是仅仅选择某个特定餐馆. 所以利用 POI 的类别信息能更好地建模用户自身偏好. 因此, 本文定义了 POI 类别隐藏特征矩阵  $\mathbf{D} \subseteq |T| \times K$ , 以挖掘 POI 类别信息对用户自身偏好的影响. 其中  $T$  为 POI 类别集合,  $K$  为隐藏特征矩阵维度. 所以用户  $u$  对 POI  $i$  的偏好程度可表示为

$$\hat{r}_{ui} = \mathbf{p}_u^T (\mathbf{q}_i + \gamma \cdot \mathbf{d}_t) \quad (7)$$

其中,  $\mathbf{d}_t$  表示 POI  $i$  所属类别  $t$  的隐藏特征向量.  $\gamma$  与  $\eta$  类似, 用来权衡 POI 与 POI 类别对建模用户偏好的影响程度比重.

## 4 融合上下文信息的矩阵分解模型

用户的签到行为是一个复杂的决策过程. 用户自身偏好起到主导作用, 上下文信息的影响也不可忽视. 本节提出了一个通用的矩阵分解模型 UCGSMF, 在融合用户自身偏好与上下文信息影响来建模用户的签到行为时, 对用户访问过的 POI 和未访问的 POI 采用不同的融合上下文信息策略, 能够更好地拟合用户自身偏好, 并且具有良好的可扩展性和较低的时间复杂度.

### 4.1 模型概览

WMF 通过给所有的数据分配一个合理的权重很好地建模了用户自身偏好, 但在 POI 推荐中, 用户的签到行为会受到多种因素影响. 除了用户自身偏好, 地理位置信息、社交网络信息等上下文信息都会影响用户的签到行为, 这些信息是 WMF 无法学习到的. 所以将这些有用的上下文信息融合到

WMF 中来建模用户的签到行为是提升 POI 推荐准确率的关键. 最近的一些研究<sup>[8-10]</sup> 考虑在矩阵分解中结合上下文信息来进行 POI 推荐, 但是这些研究往往只会融合几种特定的上下文信息, 并且对这些上下文的建模方式局限于模型本身, 修改建模方式的难度很大; 而且由于模型缺乏灵活性, 融合新的上下文信息可能使模型的复杂度变得不可控, 可扩展性差; 并且在融合上下文信息去拟合用户签到矩阵时, 对用户访问过的 POI 和未访问的 POI 是同等对待的, 而这种融合策略通过本文后续的分析可知其实是不合理的. 基于以上模型的局限性, 本文提出一个通用的融合上下文信息的矩阵分解模型.

首先给出一般的融合上下文信息的矩阵分解模型的目标函数:

$$J = \sum_{u=1}^{|U|} \sum_{i=1}^{|I|} \omega_{ui} (c_{ui} - \hat{r}'_{ui})^2 + \Theta(\mathbf{P}, \mathbf{Q}, \mathbf{D}) \quad (8)$$

$$\hat{r}'_{ui} = \hat{r}_{ui} + e_{ui} \quad (9)$$

$$\Theta(\mathbf{P}, \mathbf{Q}, \mathbf{D}) = \lambda_p \sum_{u=1}^{|U|} \|\mathbf{p}_u\|^2 + \lambda_q \sum_{i=1}^{|I|} \|\mathbf{q}_i\|^2 + \lambda_d \sum_{t=1}^{|T|} \|\mathbf{d}_t\|^2 \quad (10)$$

其中  $\hat{r}'_{ui}$  表示用户  $u$  对 POI  $i$  的最终偏好, 以刻画各种因素对用户签到行为的最终影响程度.  $\Theta(\mathbf{P}, \mathbf{Q}, \mathbf{D})$  表示正则化项. 一些研究中可能只包含用户隐藏特征矩阵  $\mathbf{P}$  和商品隐藏特征矩阵  $\mathbf{Q}$ , 这种情况下  $\lambda_d = 0$ . 最近的一些研究<sup>[8-10]</sup> 都可以归于这类模型.

可以看到, 用户  $u$  对 POI  $i$  的最终偏好  $\hat{r}'_{ui}$  由两部分信息组成, 其中  $\hat{r}_{ui}$  表示用户  $u$  自身对 POI  $i$  的兴趣偏好,  $e_{ui}$  表示上下文信息对用户  $u$  访问 POI  $i$  的影响程度. 所以最小化模型的目标函数就是让  $\hat{r}'_{ui}$  与  $c_{ui}$  的差值尽可能小. 而隐式数据的一大特点就是只知道用户的正反馈, 缺乏负反馈. 也就是说对于用户访问过的 POI, 我们可以认为用户是对它感兴趣的, 当然也可能受到一些上下文信息影响(例如距离用户很近); 而对于用户未访问过的 POI, 绝大多数都是因为用户不知道该 POI 的存在而没有访问它, 所以用户是否对它感兴趣是不确定的, 当然也有可能用户对它不感兴趣或者受上下文信息影响(例如距离用户太远)导致用户没有访问, 在这种不明确用户兴趣偏好的情况下, 使用上下文信息去拟合用户签到行为(最终偏好)是不合适的. 因此本文考虑对用户访问过的 POI ( $c_{ui} = 1$ ) 和用户未访问 POI ( $c_{ui} = 0$ ) 这两种情况采用不同的融合上下文信息策略, 表示如下:

$$\hat{r}'_{ui} = \begin{cases} \hat{r}_{ui} + e_{ui}, & (u, i) \in \mathbf{R} \\ \hat{r}_{ui}, & (u, i) \notin \mathbf{R} \end{cases} \quad (11)$$

当用户  $u$  访问过 POI  $i$  时, 用户  $u$  对 POI  $i$  的最终偏好受用户自身偏好和上下文信息两方面的影响; 而当用户  $u$  未访问过 POI  $i$  时, 只利用用户自身偏好表示用户  $u$  对 POI  $i$  的最终偏好. 下面分析这种采用不同融合上下文信息策略去拟合用户最终偏好的合理性及优势:

(1) 当  $c_{ui} = 1$  时, 表示用户  $u$  对 POI  $i$  感兴趣, 当然也可能受到上下文信息的影响. 所以分析在不同地理位置信息下(加入其他上下文信息分析类似)用户最终偏好  $\hat{r}'_{ui}$  拟合用户实际偏好  $c_{ui} = 1$  的情况. 当用户  $u$  离 POI  $i$  较远时, 地理位置对用户签到行为的正面影响较弱,  $e_{ui}$  值相对较小, 为了更好地拟合  $\hat{r}'_{ui}$  和  $c_{ui}$ , 这就使得  $\hat{r}_{ui}$  值会相对较大. 这说明尽管 POI  $i$  离用户  $u$  较远, 但用户自身对它很感兴趣( $\hat{r}_{ui}$  较大), 所以访问了它; 当用户  $u$  离 POI  $i$  较近时,  $e_{ui}$  值相对较大,  $\hat{r}_{ui}$  相对较小. 这表明用户因为离 POI 较近所以访问了它, 而用户对它的兴趣可能不是那么强烈( $\hat{r}_{ui}$  相对较小). 但是可能存在一些离用户比较近并且用户确实非常感兴趣的 POI( $e_{ui}$  和  $\hat{r}_{ui}$  值均相对较大), 这种情况下, 其实可以通过权重矩阵  $\mathbf{W}$  弥补, 因为这样的 POI 用户访问频率会很高, 相应的权重变高会使  $\hat{r}'_{ui}$  与  $c_{ui}$  差值变小, 在  $e_{ui}$  不变的情况下  $\hat{r}_{ui}$  也会变大. 所以在  $c_{ui} = 1$  时, 融合上下文信息能更好地拟合用户自身偏好.

(2) 当  $c_{ui} = 0$  时, 因为在 LBSN 中 POI 数量十分庞大, 所以用户  $u$  没有访问 POI  $i$  的原因大概率是不知道该 POI 的存在(当然也有可能是用户  $u$  确实对 POI  $i$  不感兴趣或者受上下文信息影响所致), 所以用户  $u$  对 POI  $i$  是否感兴趣其实是不确定的. 在这种用户兴趣不明确的情况下, 使用上下文信息去拟合用户的最终偏好显然不合适. 而且因为上下文信息对用户签到行为的影响程度  $e_{ui}$  的值大部分都大于 0, 如果加入上下文信息, 为了拟合  $\hat{r}'_{ui} = 0$ , 将会导致用户自身的偏好  $\hat{r}_{ui}$  为负, 而用户未访问该 POI 并不代表用户对它不感兴趣, 用户只是不知道它的存在而已. 所以将用户未访问的 POI 都建模成用户对其不感兴趣显然是不合理的. 因此这种情况下, 只使用用户自身的兴趣偏好去拟合用户的签到行为更加合适.

另外, 在考虑模型的复杂度时, 因为矩阵分解模型的复杂度主要来源于签到矩阵中为零的项, 在融

合了上下文信息后, 一般模型因为无法有效区别对待签到矩阵中非零项和缺失项, 往往需要遍历整个签到矩阵进行矩阵计算, 这样会使得模型的复杂度很高. 而本文模型在建模用户对未访问过的 POI 的最终偏好时, 只考虑了用户自身偏好, 而没有融合上下文信息影响, 所以可以像普通矩阵分解一样将模型优化分为用户访问的 POI 和未访问的 POI 两种情况, 这能极大地降低模型的复杂度, 本文将在后面的小节给出复杂度分析.

同时, 在建模上下文信息对用户签到行为的影响上, 本文模型具有良好的可扩展性. 本文只定义了上下文信息融入矩阵分解模型的框架, 而对建模上下文信息对用户签到行为的影响程度  $E$  则是非常灵活的. 并不局限于本文提到的上下文信息, 而且对单个上下文信息的建模方式以及上下文信息的组合方式也允许多样性. 第 3 节建模了地理位置和社交网络两种上下文信息对用户签到行为的影响, 因此这里将这两种上下文信息进行线性组合来建模上下文信息对用户签到行为的影响, 公式如下:

$$e_{ui} = \alpha g_{ui} + \beta s_{ui} \quad (12)$$

其中  $\alpha$  和  $\beta$  分别控制地理位置信息与社交网络信息对用户签到行为的影响程度. 注意, 本文关注的问题是—般性的 POI 推荐问题, 在建模上下文信息影响( $\mathbf{G}$  和  $\mathbf{S}$ )时采用的是基本的协同过滤方法. 如果修改上下文信息的建模方式, 则可以将本文模型适用于更多 POI 推荐问题, 比如冷启动问题、异地 POI 推荐等.

本文提出的通用的基于矩阵分解的上下文感知 POI 推荐模型 UCGSMF 概述如下: 首先利用用户的签到数据( $\mathbf{U}$ )建模用户自身偏好, 并且考虑了 POI 类别信息( $\mathbf{C}$ )对用户自身偏好的影响. 因为用户签到行为除了受用户自身偏好影响之外, 还受到其他上下文信息的影响, 所以本文建模了地理位置( $\mathbf{G}$ )和社交网络( $\mathbf{S}$ )这两种重要的上下文信息对用户签到行为的影响, 最后提出一个通用的矩阵分解模型(UCGSMF), 对用户访问过的 POI 和未访问过的 POI 采用不同的融合策略, 使得模型能够很好地融合用户自身偏好与上下文信息影响来刻画用户签到行为. 因此, 本文模型的目标函数表示如下:

$$J = \sum_{(u, i) \in \mathbf{R}} \omega_{ui} (c_{ui} - (\hat{r}_{ui} + \alpha g_{ui} + \beta s_{ui}))^2 + \sum_{(u, i) \notin \mathbf{R}} \hat{r}_{ui}^2 + \Theta(\mathbf{P}, \mathbf{Q}, \mathbf{D}) \quad (13)$$

其中用户自身的偏好  $\hat{r}_{ui}$  可由式(7)获得. 上下文信息对用户签到行为的影响  $e_{ui}$  使用式(12)表示,

正则化项  $\Theta(\mathbf{P}, \mathbf{Q}, \mathbf{D})$  由式(10)获得.

#### 4.2 算法优化与复杂度分析

为了解决模型的优化问题,本文借鉴元素级的最小交替二乘(eALS)算法<sup>[35]</sup>中优化方法.该优化算法与 ALS 一样,都是固定其他隐藏特征矩阵每次只计算一个隐藏特征矩阵,不同点在于 eALS 每次只更新一个元素而不是一个向量.类似于 eALS 的做法,首先更新用户隐藏特征矩阵  $\mathbf{P}$  的一个元素  $p_{uf}$ ,对式(13)求导可得:

$$p_{uf} = \frac{\sum_{i \in \mathbf{R}_u} w_{ui} h_{if} (c_{ui} - (\hat{r}_{ui}^p + e_{ui})) - \sum_{i \notin \mathbf{R}_u} h_{if} \hat{r}_{ui}^p}{\sum_{i \in \mathbf{R}_u} w_{ui} h_{if}^2 - \sum_{i \notin \mathbf{R}_u} h_{if}^2 + \lambda_p} \quad (14)$$

其中  $h_{if}$ ,  $\hat{r}_{ui}^p$  表示如下:

$$h_{if} = q_{if} + \gamma d_{i,f} \quad (15)$$

$$\hat{r}_{ui}^p = \hat{r}_{ui} - p_{uf} h_{if} \quad (16)$$

从式(14)中可以看到,因为本文对访问过的 POI 和未访问过的 POI 采用了不同的融合上下文信息策略,所以上下文信息只存在于用户访问过的 POI 的求和项  $\sum_{i \in \mathbf{R}_u}$  上,并没有出现在用户未访问过的 POI 的求和项  $\sum_{i \notin \mathbf{R}_u}$  上.而计算  $p_{uf}$  的主要复杂度来自于对缺失项的求和上,这样就确保了我们可以像 eALS 一样进行优化(如果在缺失项上出现了上下文信息,将无法像 eALS 那样抽取出缓存项进行优化).借鉴 eALS 的做法,对第一个缺失项求和部分进行如下变换:

$$\begin{aligned} \sum_{i \notin \mathbf{R}_u} h_{if} \hat{r}_{ui}^p &= \sum_{i=1}^{|I|} h_{if} \sum_{k \neq f} p_{uk} h_{ik} - \sum_{i \in \mathbf{R}_u} h_{if} \hat{r}_{ui}^p \\ &= \sum_{k \neq f} p_{uk} \sum_{i=1}^{|I|} h_{if} h_{ik} - \sum_{i \in \mathbf{R}_u} h_{if} \hat{r}_{ui}^p \end{aligned} \quad (17)$$

这样重构后,可以发现项  $\sum_{i=1}^{|I|} h_{if} h_{ik}$  是与  $u$  无关的,能在计算所有用户隐藏特征矩阵前预先计算.所以定义缓存矩阵  $\mathbf{X}^h = \sum_{i=1}^{|I|} (\mathbf{q}_i + \gamma \mathbf{d}_i)(\mathbf{q}_i + \gamma \mathbf{d}_i)^\top$ ,易知该缓存项的计算复杂度为  $O(|I|K^2)$ . 这样式(17)重写为

$$\sum_{i \notin \mathbf{R}_u} h_{if} \hat{r}_{ui}^p = \sum_{k \neq f} p_{uk} x_{fk}^h - \sum_{i \in \mathbf{R}_u} h_{if} \hat{r}_{ui}^p \quad (18)$$

因为  $x_{fk}^h$  可以提前计算,所以计算  $\sum_{i \notin \mathbf{R}_u} h_{if} \hat{r}_{ui}^p$  的复杂度为  $O(K + |\mathbf{R}_u|)$ . 类似地,对另一个缺失项的求和可进行相同的优化处理:

$$\sum_{i \notin \mathbf{R}_u} h_{if}^2 = \sum_{i=1}^{|I|} h_{if}^2 - \sum_{i \in \mathbf{R}_u} h_{if}^2 = x_{ff}^h - \sum_{i \in \mathbf{R}_u} h_{if}^2 \quad (19)$$

这样,式(14)可改写为

$$p_{uf} = \frac{\sum_{i \in \mathbf{R}_u} h_{if} (w_{ui} (c_{ui} - e_{ui}) - (w_{ui} - 1) \hat{r}_{ui}^p) - \sum_{k \neq f} p_{uk} x_{fk}^h}{\sum_{i \in \mathbf{R}_u} (w_{ui} - 1) h_{if}^2 + x_{ff}^h + \lambda_p} \quad (20)$$

现在重新考虑计算  $p_{uf}$  的时间复杂度.如果在计算  $p_{uf}$  前提前计算好  $\hat{r}_{ui}^p$ ,那么整个式(20)中每个变量是已知并可以直接获取的,所以计算  $p_{uf}$  的复杂度为求和的代价,即  $O(K + |\mathbf{R}_u|)$ . 那么计算所有的  $p_{uf}$  的复杂度为  $O(|U|K^2 + |\mathbf{R}|K)$ .

类似地,可以计算另外两个隐藏特征矩阵  $\mathbf{Q}, \mathbf{D}$ :

$$q_{if} = \frac{\sum_{u \in \mathbf{R}_i} p_{uf} (w_{ui} (c_{ui} - e_{ui}) - (w_{ui} - 1) \hat{r}_{ui}^q) - \sum_{k \neq f} q_{ik} x_{fk}^p - \gamma \sum_{k=1}^K d_{i,f} x_{fk}^p}{\sum_{u \in \mathbf{R}_i} (w_{ui} - 1) p_{uf}^2 + x_{ff}^p + \lambda_q} \quad (21)$$

$$d_{i,f} = \frac{\sum_{i \in \mathbf{Z}_t} (\sum_{u \in \mathbf{R}_i} \gamma p_{uf} (w_{ui} (c_{ui} - e_{ui}) - (w_{ui} - 1) \hat{r}_{ui}^d) - \gamma \sum_{k=1}^K q_{ik} x_{fk}^p - \gamma^2 \sum_{k \neq f} d_{i,f} x_{fk}^p)}{\sum_{i \in \mathbf{Z}_t} (\sum_{u \in \mathbf{R}_i} \gamma^2 (w_{ui} - 1) p_{uf}^2 + \gamma^2 x_{ff}^p) + \lambda_d} \quad (22)$$

其中  $\mathbf{Z}_t$  表示属于 POI 类别  $t$  的 POI 集合.  $\mathbf{X}^p = \mathbf{P}^\top \mathbf{P}$  也是可以预先计算的缓存项,  $x_{fk}^p$  是它的一个元素.其中  $\hat{r}_{ui}^q$  和  $\hat{r}_{ui}^d$  表示如下:

$$\hat{r}_{ui}^q = \hat{r}_{ui} - p_{uf} q_{if} \quad (23)$$

$$\hat{r}_{ui}^d = \hat{r}_{ui} - \gamma p_{uf} d_{i,f} \quad (24)$$

算法 1 展示了 UCGSMF 算法的优化过程.第 1 行初始化需要求解的隐藏特征矩阵  $\mathbf{P}, \mathbf{Q}, \mathbf{D}$ , 然后进入循环迭代直到达到收敛条件或者迭代次数.第 3 行预先计算缓存矩阵  $\mathbf{X}^h$ , 其计算复杂度为  $O(|I|K^2)$ . 第 4~8 行依据式(20)计算用户隐藏特征矩阵  $\mathbf{P}$ , 其计算复杂度为  $O(|U|K^2 + |\mathbf{R}|K)$ . 第 9 行计算另一个缓存矩阵  $\mathbf{X}^p$ , 其计算复杂度为  $O(|U|K^2)$ . 第 10~14 行和 15~21 行分别计算 POI 隐藏特征矩阵  $\mathbf{Q}$  和 POI 类别隐藏特征矩阵  $\mathbf{D}$ , 两者的计算复杂度一样,均为  $O(|I|K^2 + |\mathbf{R}|K)$  (复杂度分析与用户隐藏特征矩阵类似,这里不再赘述).所以整个模型的复杂度为  $O((|U| + |I|)K^2 + |\mathbf{R}|K)$ , 与数据集大小  $|\mathbf{R}|$ 、用户集合大小  $|U|$  和 POI 集合大小  $|I|$  成正比,与隐藏特征矩阵维度  $K$  成平方比,但是与 POI 类别集合大小  $|T|$  无关,这是因为在优化过程中,可以将 POI 类别集合空间转化到对应的 POI 集合空间上.可以看到,本文提出的模型复杂度相比 WMF 的复杂度  $O((|U| + |I|)K^3 + |\mathbf{R}|K^2)$  降低了  $K$  倍,与 eALS 模型的复杂度一致.在融合了各种上下文信息后,本文模型复杂度依然与没有融合任何上下



文信息的 eALS 保持一致,这主要是因为本文对访问过的 POI 和未访问过的 POI 采用了不同的融合上下文信息策略,使得我们可以像 eALS 那样进行优化,从而获得一致的时间复杂度,这是目前已知的最优的复杂度。

### 算法 1. UCGSMF 优化算法.

输入: 用户-POI 签到矩阵  $\mathbf{R}$ , 权重矩阵  $\mathbf{W}$ , 用户对 POI 在上下文信息上的偏好矩阵  $\mathbf{G}$  和  $\mathbf{S}$ , 隐藏特征维度  $K$ , 正则化系数  $\lambda_p, \lambda_q, \lambda_d$

输出: 用户隐藏特征矩阵  $\mathbf{P}$ , POI 隐藏特征矩阵  $\mathbf{Q}$ , POI 类别隐藏特征矩阵  $\mathbf{D}$

1. 初始化隐藏特征矩阵  $\mathbf{P}, \mathbf{Q}, \mathbf{D}$

2. REPEAT

//更新用户隐藏特征矩阵

3.  $\mathbf{X}^h = \sum_{i=1}^{|I|} (\mathbf{q}_i + \gamma \mathbf{d}_i)(\mathbf{q}_i + \gamma \mathbf{d}_i)^T$

4. FOR  $u$  from 1:  $|U|$  DO

5. FOR  $f$  from 1:  $K$  DO

6.  $p_{uf} \leftarrow$  式(20)

7. END

8. END

//更新 POI 隐藏特征矩阵

9.  $\mathbf{X}^p = \mathbf{P}^T \mathbf{P}$

10. FOR  $i$  from 1:  $|I|$  DO

11. FOR  $f$  from 1:  $K$  DO

12.  $q_{if} \leftarrow$  式(21)

13. END

14. END

//更新 POI 类别隐藏特征矩阵

15. FOR  $t$  from 1:  $|T|$  DO

16. FOR  $i$  from 1:  $Z_i$  DO

17. FOR  $f$  from 1:  $K$  DO

18.  $d_{if} \leftarrow$  式(22)

19. END

20. END

21. END

22. UNIT 收叙

23. RETURN  $\mathbf{P}, \mathbf{Q}, \mathbf{D}$

## 5 实验结果与分析

本节将详细描述本文实验中用到的数据集、实验参数设置、度量指标、评估方法以及实验结果和分析。

### 5.1 数据集

本文实验选取两个具有代表性的 LBSN 数据集, 大众点评数据集和 Foursquare 数据集. 大众点评数据集包含 2031 个用户对 5649 个 POI 的 177 686 条签到记录, 稠密度为  $1.55 \times 10^{-2}$ ; Foursquare 数据集包含 3447 个用户对 18 459 个 POI 的 225 081 条签到记录, 稠密度为  $3.54 \times 10^{-3}$ . 除了签到数据, 数据集中还包含 POI 的经纬度信息、POI 类别信息以及用户的社交网络信息, 具体数据集统计信息如表 2 所示, 其中列“DianPing”表示大众点评数据集信息, 每个数据集随机选取 80% 作为训练集, 20% 作为测试集。

表 2 数据集统计信息

	# 用户数量	# POI 数量	# POI 类别数量	# 签到总数	# 社交关系总数	# POI 平均被签到数	# 用户平均签到 POI 数	# POI 被签到最大数	# 用户签到 POI 最大数
DianPing	2031	5649	135	177 686	27 776	30.08	83.67	560	1086
Foursquare	3447	18 459	21	225 081	41 689	7.29	39.01	668	341

### 5.2 实验设置

在参数设置上, 本文中所用到的对比方法与其所在原文中设置尽量保持一致, 并调节到最优. 而本文提出的模型中, 核函数的宽度参数  $b$  在大众点评数据集上设置为 1.6 km 而在 Foursquare 数据集上设置为 3 km. POI 类别对社交网络的影响因子  $\eta$  在大众点评数据集上设置为 0.3 在 Foursquare 数据集上设置为 0 (表明在该数据集上社交网络信息的影响不明显). 正则化系数  $\lambda_p = \lambda_q = \lambda_d = 30$ . 权重参数  $\epsilon$  在大众点评数据集和 Foursquare 数据集上分别设置为 3 和 2. 上下文影响因子  $(\gamma, \alpha, \beta)$  在大众点评数据集和 Foursquare 数据集上分别设置为  $(0.3, 0.7, 0.3)$  和  $(0.8, 0.6, 0.1)$ . 调节实验参数采用的是网格搜索方式, 上述参数的取值均为实验效果达到

最好时的值. 另外, 实验的迭代轮数设置为 20 轮, 默认隐藏特征矩阵维度  $K = 20$ .

本文采用常见的准确率 ( $Precision@n$ ) 和召回率 ( $Recall@n$ ) 两个评价指标评测模型的推荐性能, 计算公式如下所示:

$$Precision@n = \frac{1}{|U|} \sum_{u=1}^{|U|} \frac{|\zeta_u^p \cap \zeta_u^t|}{n} \quad (25)$$

$$Recall@n = \frac{1}{|U|} \sum_{u=1}^{|U|} \frac{|\zeta_u^p \cap \zeta_u^t|}{|\zeta_u^t|} \quad (26)$$

其中  $\zeta_u^p$  表示模型对用户  $u$  预测的 POI 集合列表,  $\zeta_u^t$  表示测试集中用户  $u$  访问过的 POI 集合列表,  $|\zeta_u^t| = n$ . 本文选取  $n = 5$  时的  $Precision$  和  $Recall$  作为评测指标.

实验对比了 9 种可以应用在 LBSN 数据集上的

算法:

User\_CF: 基于用户的协同过滤方法, 通过用户相似度计算寻找相似的用户来进行推荐.

Item\_CF: 基于物品的协同过滤方法, 与 User\_CF 类似, 通过物品相似度计算寻找相似物品进行推荐. 实验中设置 User\_CF 和 Item\_CF 邻居数为 100.

WMF: 基于隐式数据的矩阵分解技术, 通过给所有数据(包括签到数据和缺失项)分配一个合理的权重来进行推荐<sup>[34-35]</sup>. 在不考虑签到矩阵缺失项的权重时, eALS 与 WMF 能获得基本一致的推荐性能, 在后面实验中, 没有特别说明, 本文将 WMF 和 eALS 统一称作 WMF.

GeoMF: 该方法在加权矩阵分解的基础上融合了地理位置信息来进行 POI 推荐<sup>[8]</sup>.

ASMF: 最新的基于矩阵分解的 POI 推荐算法<sup>[17]</sup>, 通过定义三种朋友关系去挖掘用户潜在感兴趣的 POI, 然后结合矩阵分解进行推荐.

UCGMF: 本文提出的模型, 只考虑用户自身偏好和地理位置信息来建模用户签到行为.

UCSMF: 本文提出的模型, 只考虑用户自身偏好和社交网络信息来建模用户签到行为.

UCGSMF: 本文提出的模型, 综合考虑用户自身偏好、地理位置信息和社交网络信息来建模用户签到行为.

UCGSMF\_GEN: 与 UCGSMF 类似, 考虑了用户自身偏好、地理位置信息和社交网络信息来建模用户签到行为. 但是对用户访问过的 POI 和未访问过的 POI 采用相同的融合上下文信息策略.

注意, 上述本文提出的模型中, 若没有特别说

明, 对用户访问过的 POI 和未访问过的 POI 均采用不同的融合上下文信息策略.

### 5.3 实验结果

本文首先对比在两个数据集上不同 POI 推荐方法的推荐结果, 结果如表 3 所示. 可以看出, WMF 在拥有足够的隐式特征矩阵维度 ( $K \geq 10$ ) 时推荐性能要优于简单的协同过滤算法 (Item\_CF 和 User\_CF), 而 GeoMF 和 ASMF 都是在 WMF 的基础上进行了优化, 所以它们的推荐性能基本上优于 WMF, 其中 ASMF 的推荐效果总体上略优于 GeoMF. 本文提出的模型中, 融合用户自身偏好与地理位置信息的 UCGMF 推荐性能基本高于融合用户自身偏好与社交网络信息的 UCSMF, 这说明地理位置信息在 POI 推荐中比社交网络信息的影响更大, 并且 UCSMF 和 UCGMF 的推荐总体上也优于 ASMF 和 GeoMF. 而融合了用户自身偏好与所有上下文信息影响的 UCGSMF 推荐性能显著地高于其他所有对比方法. 相比于 ASMF, UCGSMF 的推荐性能提高 10%~20% 左右. 这说明本文模型通过在 WMF 基础上有效地融合多种上下文信息, 确实能缓解数据稀疏性, 从而获得较好的推荐性能, 而且融入越多有效的上下文信息, 模型推荐性能就越好, 例如, 当用户碰到一个没访问过的 POI 时, WMF 只能依靠用户的历史行为来预测用户对该 POI 感兴趣程度, 而无法学习到该 POI 的特征, 而本文模型可以通过 POI 类别信息将该 POI 与用户历史记录相关联从而捕捉到用户对该 POI 的兴趣偏好, 即使缺乏用户的偏好信息, 我们还可以通过用户或 POI 的上下文信息给用户做出推荐, 所以本文模型能获得更好的推荐性能.

表 3 不同方法准确率和召回率对比, 列“Improve”表示 UCGSMF 相比方法 ASMF 推荐效果的提高百分比

Database	K	Metrics	User_CF	Item_CF	WMF	GeoMF	ASMF	UCGMF	UCSMF	UCGSMF	UCGSMF_GEN	UCGSMF	Improve
DianPing	5	Precision@5	0.0601	0.0666	0.0797	0.0862	0.0851	0.0991	0.0855	<b>0.1007</b>	0.0988	0.1606	
		Recall@5	0.0217	0.0226	0.0290	0.0298	0.0299	0.0395	0.0339	<b>0.0430</b>	0.0412	0.3770	
	10	Precision@5	0.0601	0.0666	0.0925	0.0982	0.0979	0.1065	0.0987	<b>0.1103</b>	0.1072	0.0951	
		Recall@5	0.0217	0.0226	0.0340	0.0343	0.0363	0.0425	0.0387	<b>0.0462</b>	0.0442	0.2173	
	20	Precision@5	0.0601	0.0666	0.1010	0.1022	0.1046	0.1110	0.1082	0.1107	<b>0.1153</b>	0.1023	
		Recall@5	0.0217	0.0226	0.0380	0.0369	0.0392	0.0436	0.0427	0.0467	<b>0.0476</b>	0.2145	
	40	Precision@5	0.0601	0.0666	0.0933	0.1033	0.0972	0.1048	0.1049	0.1031	<b>0.1091</b>	0.1223	
		Recall@5	0.0217	0.0226	0.0362	0.0386	0.0396	0.0420	0.0436	0.0457	<b>0.0461</b>	0.1635	
Foursquare	5	Precision@5	0.0429	0.0566	0.0527	0.0518	0.0564	0.0577	0.0524	0.0583	<b>0.0621</b>	0.1007	
		Recall@5	0.0320	0.0362	0.0327	0.0324	0.0364	0.0388	0.0319	0.0404	<b>0.0418</b>	0.1475	
	10	Precision@5	0.0429	0.0566	0.0569	0.0572	0.0595	0.0617	0.0573	0.0652	<b>0.0657</b>	0.1051	
		Recall@5	0.0320	0.0362	0.0338	0.0347	0.0372	0.0402	0.0334	0.0427	<b>0.0431</b>	0.1603	
	20	Precision@5	0.0429	0.0566	0.0612	0.0600	0.0609	0.0687	0.0642	0.0686	<b>0.0712</b>	0.1683	
		Recall@5	0.0320	0.0362	0.0369	0.0358	0.0378	0.0442	0.0374	0.0444	<b>0.0450</b>	0.1898	
	40	Precision@5	0.0429	0.0566	0.0621	0.0584	0.0577	0.0670	0.0628	0.0693	<b>0.0702</b>	0.2166	
		Recall@5	0.0320	0.0362	0.0376	0.0364	0.0347	0.0433	0.0373	0.0451	<b>0.0459</b>	0.3220	

而且对比于 UCGSMF\_GEN, 可以看到 UCGSMF 在不同的数据集上推荐性能基本上都优于 UCGSMF\_GEN, 这也说明对用户访问过的 POI 和未访问过的 POI 采用不同的融合上下文信息策略确实能够更好地建模用户的签到行为, 从而获得更优的推荐性能. 在大众点评数据集上, 当维度较低时可以发现 UCGSMF\_GEN 的推荐性能越略优于 UCGSMF, 这可能是因为过低的特征维度使得矩阵分解模型无法很好地建模用户和 POI 之间隐藏的联系, 而这时 UCGSMF\_GEN 通过对所有数据点都考虑上下文信息的影响则能弥补这一点, 所以获得更好的推荐性能. 但是当矩阵维度增加时, 矩阵分解模型能够很好地建模用户的兴趣偏好, 这时对所有数据都考虑上下文信息的影响反而不利于用户自身偏好的拟合. 另外从表中还能看出, 在大众点评和 Foursquare 数据集上隐藏特征矩阵维度  $K=20$  时, 基于矩阵分解

的方法的推荐性能都达到最优, 当维度上升到 40 维时, 这类方法的推荐性能开始出现下降, 这表明模型已经过拟合了, 所以在接下来的实验中设置隐藏特征矩阵维度为 20.

接下来分析不同上下文信息在矩阵分解模型中所占比重对模型性能的影响. 首先分析 POI 类别信息对建模用户自身偏好的影响(设置  $\alpha=\beta=0$ ). 图 2 展示 POI 类别信息影响因子  $\gamma$  在取不同值对模型推荐性能的影响. 从两个数据集上可以看到, 不同取值的  $\gamma$  对模型推荐性能的影响很大, 合适的取值能够有效地提高模型的推荐性能, 而不好的取值则会影响模型的推荐性能. 在大众点评数据集上  $\gamma=0.3$ , 在 Foursquare 数据集上  $\gamma=0.8$ , 模型推荐性能达到最优. 因此在接下来的实验中,  $\gamma$  的取值均为模型推荐性能达到最优情况下的取值.

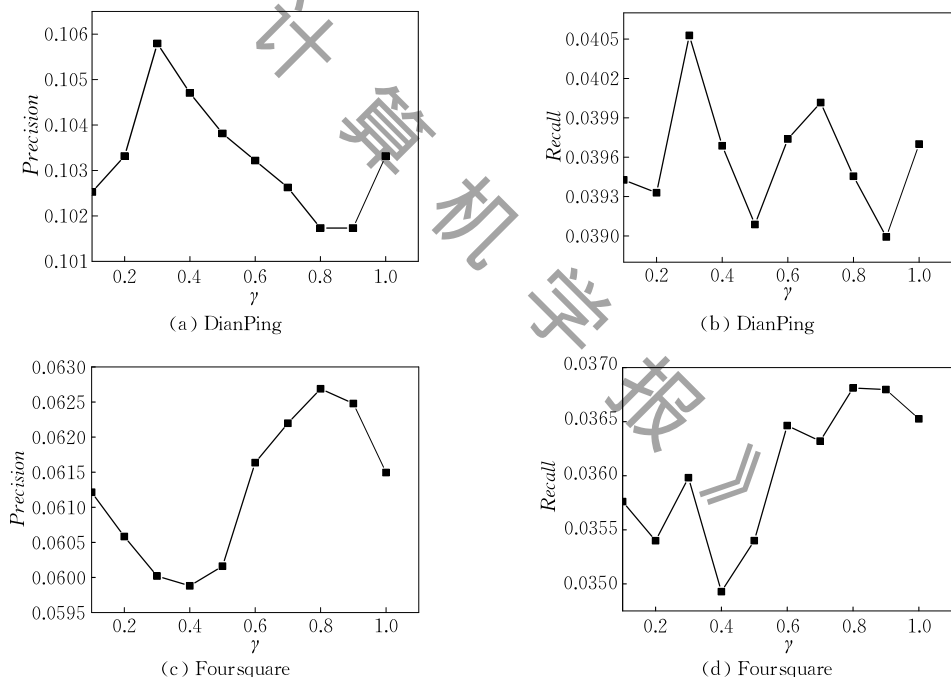


图 2 不同  $\gamma$  对 UCGSMF 推荐性能影响

用户的签到行为除了受自身偏好影响, 还受到地理位置信息以及社交网络信息等上下文信息共同影响, 而后两者对用户签到行为的影响程度分别由  $\alpha$  和  $\beta$  控制, 单独分析地理位置信息或者社交网络信息对推荐性能的影响显然是不合适的. 所以下面分析不同的  $\alpha$  和  $\beta$  取值对模型性能的影响, 如图 3 所示. 其中, 横坐标表示  $\alpha$  取值, 纵坐标表示在不同  $\beta$  取值下的推荐性能. 我们验证了  $\beta$  在范围  $[0, 1]$  之间以 0.1 为间隔的推荐结果, 由于对于不同  $\beta$  取值模型推荐性能呈现出一定的规律, 因此本文只画出

了  $\beta=\{0.1, 0.3, 0.5, 0.7, 0.9\}$  5 个值的结果曲线, 其中 UCMF 模型表示只考虑用户自身偏好不考虑任何上下文信息影响. 从图中可以分析得到以下几点结论:

(1) 融合地理位置信息和社交网络信息能提高 POI 推荐性能. 可以看到, 在两个数据集上, 随着  $\alpha$  或  $\beta$  的增大, POI 的推荐性能 ( $Precision$  和  $Recall$ ) 都逐渐上升, 但是当  $\alpha$  或  $\beta$  达到某个阈值时, 推荐性能不再增加并逐步降低. 在大众点评数据集上,  $\alpha=0.7, \beta=0.3$  模型推荐性能达到最优; 在 Foursquare

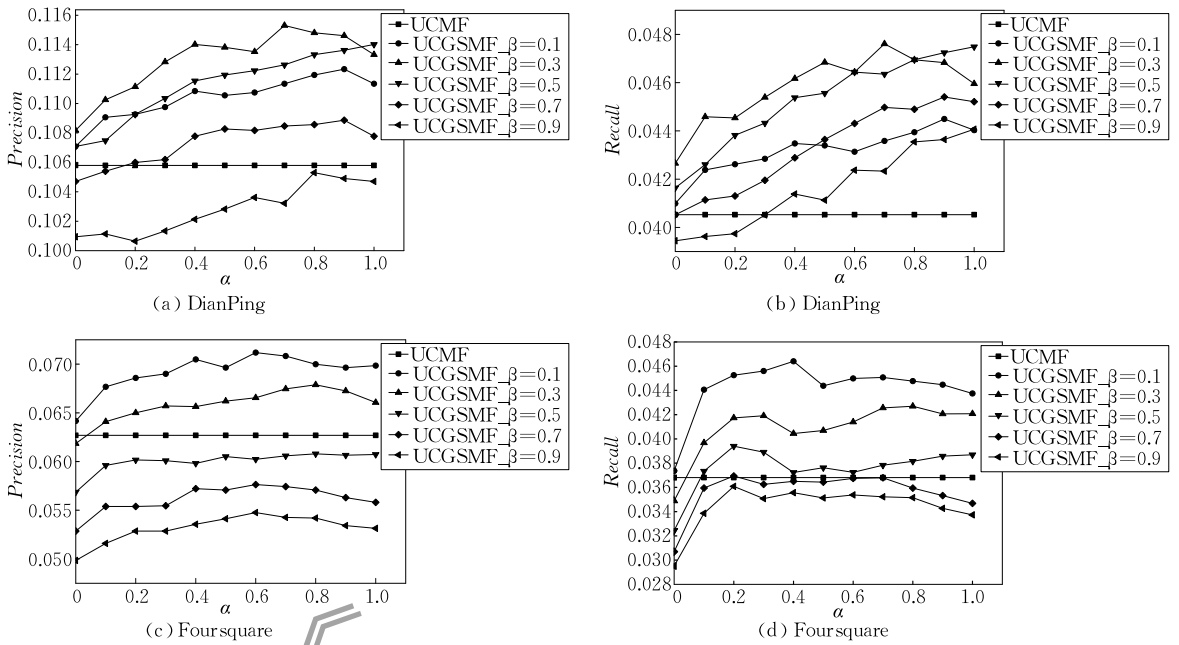


图 3 不同  $\alpha$  和  $\beta$  对 UCGSMF 推荐性能影响

数据集上,  $\alpha=0.6, \beta=0.1$  模型推荐性能达到最优;

(2) 地理位置信息和社交网络信息对 POI 推荐性能的贡献是不一样的. 当  $\alpha=0$  时, 表示的是只考虑用户自身偏好和社交网络信息影响建模用户签到行为, 可以发现相比 UCMF (只考虑用户自身偏好建模用户签到行为), 社交网络信息的影响带来的增益很小, 尤其是在 Foursquare 数据集上, 社交网络信息带来的推荐性能增益只有 2% 左右, 而且社交网络信息在建模用户签到行为中所占的比重也较小 (大众点评数据集上  $\beta=0.3$ , Foursquare 数据集上  $\beta=0.1$  模型推荐性能达到最优), 一旦超过这个临界点, 推荐性能就会明显下降, 甚至比 UCMF 的推荐性能还差; 而地理位置信息对 POI 推荐性能的增益效果则比较明显. 随着  $\alpha$  的增大, POI 的推荐性能提升比较明显, 并且可以发现地理位置信息在建模用户签到行为中所占的比重也较大 (大众点评数据集上  $\alpha=0.7$ , Foursquare 数据集上  $\alpha=0.5$  左右模型推荐性能达到最优), 而且即使  $\alpha$  的比重超过模型最优时的取值, 甚至稍微超过用户自身偏好的比重 ( $\alpha>1$ ), 它仍然对用户签到行为有所增益;

(3) 地理位置信息和社交网络信息对用户签到行为的影响是相对独立的. 尽管地理位置信息和社交网络信息共同影响着用户签到行为, 但是从中可以发现, 在两个数据集上, 无论是准确率还是召回率, 地理位置信息对 POI 推荐性能的影响在不同社交网络影响因子  $\beta$  取值下都呈现出相似的变化趋势. 这说明地理位置信息和社交网络信息对用户签

到行为的影响是相对独立的.

最后对本文提出的 UCGSMF 模型的运行时间进行分析. 表 4 是 UCGSMF 和其他五种算法在两个数据集上的运行时间对比, 实验结果为 10 轮迭代时间的平均值.

首先将本文模型 UCGSMF 与传统的矩阵分解模型 WMF 和 eALS 进行对比. 可以发现在两个数据集上, 随着维度  $K$  的增加, WMF 的运行时间相比 eALS 运行时间的倍数逐渐增加, 但是没有达到  $K$  倍的差距, 其原因是 WMF 在计算矩阵转置时采用了优化技术<sup>[35]</sup>, 而 UCGSMF 模型的运行时间与 eALS 模型基本处在一个量级上, 这就验证了本文模型的时间复杂度确实与 eALS 是一致的, 比 eALS 时间复杂度略高的原因是 UCGSMF 模型需要做更多的预处理和参数计算.

然后将 UCGSMF 模型与考虑了上下文信息的矩阵分解模型 ASMF 和 GeoMF 进行对比. 可以看到 UCGSMF 运行时间显著低于 ASMF 和 GeoMF. GeoMF 模型复杂度高的原因在于它构建的网格数量非常大, 而网格大小等于矩阵分解隐空间的大小, 所以时间复杂度很高; 而 ASMF 是基于普通的 WMF 进行优化的, 所以其时间复杂度也明显高于本文提出的基于 eALS 进行优化的 UCGSMF.

最后, 可以看到 UCGSMF 运行时间远低于 UCGSMF\_GEN, 因为 UCGSMF 在拟合用户未访问过的 POI 时没有考虑上下文信息的影响, 这使得我们可以像普通矩阵分解那样进行模型优化, 有效

地降低了模型的时间复杂度;而 UCGSMF\_GEN 在模型优化过程中则需要访问整个用户-POI 矩阵,所以时间复杂度非常高.综上所述,本文提出的 UCGSMF 模型具有较低的时间复杂度.

表 4 不同算法在两个数据集上运行时间对比

DataSet	K	eALS	WMF	ASMF	GeoMF	UCGSMF- GEN	UCGSMF
DianPing	5	0.033	0.310	0.649	182.013	7.552	0.121
	10	0.110	0.476	1.252	184.435	12.254	0.365
	20	0.319	1.164	3.296	164.909	21.581	0.404
	40	0.487	3.364	12.493	166.588	30.370	0.820
Four-square	5	0.055	0.757	1.061	900.501	7.554	0.207
	10	0.179	1.511	2.016	905.307	11.405	0.255
	20	0.342	3.285	5.788	880.165	22.350	0.641
	40	0.922	7.729	23.551	873.703	39.863	1.958

## 6 结束语

本文提出一个通用的基于矩阵分解的上下文感知 POI 推荐模型 UCGSMF,该模型在融合上下文信息与用户自身偏好以建模用户签到行为时,对用户访问过的 POI 和未访问过的 POI 采用不同的融合上下文信息策略,能更有效地刻画用户自身偏好,并且具有良好的可扩展性和较低的模型复杂度.为了模拟用户的签到行为,本文建模了地理位置信息、社交网络信息以及用户自身偏好等多个方面对用户签到行为的影响,有效地缓解了签到数据的稀疏性.最后,通过两个真实数据集上的实验,表明 UCGSMF 在准确性上优于当前流行的 POI 推荐算法,同时具有较低的时间复杂度.

## 参 考 文 献

[1] Ye M, Yin P, Lee W C, et al. Exploiting geographical influence for collaborative point-of-interest recommendation// Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. Beijing, China, 2011: 325-334

[2] Tobler W R. A computer movie simulating urban growth in the Detroit region. *Economic Geography*, 1970, 46(Sup1): 234-240

[3] Noulas A, Scellato S, Lathia N, et al. A random walk around the city: New venue recommendation in location-based social networks//Proceedings of the 2012 International Conference on Privacy, Security, Risk and Trust (PASSAT), and 2012 International Conferenece on Social Computing (SocialCom). Amsterdam, The Netherlands, 2012: 144-153

[4] Ma H, Zhou D, Liu C, et al. Recommender systems with

social regularization//Proceedings of the 4th ACM International Conference on Web Search and Data Mining. Hong Kong, China, 2011: 287-296

[5] Zhang J D, Chow C Y. IGSLR: Personalized geo-social location recommendation: A kernel density estimation approach// Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. Orlando, USA, 2013: 334-343

[6] Cheng C, Yang H, King I, et al. Fused matrix factorization with geographical and social influence in location-based social networks//Proceedings of the 26th AAAI Conference on Artificial Intelligence. Toronto, Canada, 2012, 12: 1

[7] Liu B, Fu Y, Yao Z, et al. Learning geographical preferences for point-of-interest recommendation//Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Chicago, USA, 2013: 1043-1051

[8] Lian D, Zhao C, Xie X, et al. GeoMF: Joint geographical modeling and matrix factorization for point-of-interest recommendation//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2014: 831-840

[9] Li X, Cong G, Li X L, et al. Rank-GeoFM: A ranking based geographical factorization method for point of interest recommendation//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. Shanghai, China, 2015: 433-442

[10] Liu Y, Wei W, Sun A, et al. Exploiting geographical neighborhood characteristics for location recommendation// Proceedings of the 23rd ACM International Conference on Information and Knowledge Management. Shanghai, China, 2014: 739-748

[11] Cho E, Myers S A, Leskovec J. Friendship and mobility: User movement in location-based social networks//Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego, USA, 2011: 1082-1090

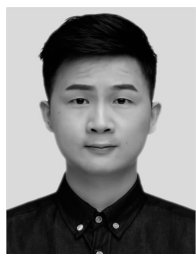
[12] Lichman M, Smyth P. Modeling human location data with mixtures of kernel densities//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2014: 35-44

[13] Yuan Q, Cong G, Sun A. Graph-based point-of-interest recommendation with geographical and temporal influences// Proceedings of the 23rd ACM International Conference on Information and Knowledge Management. Shanghai, China, 2014: 659-668

[14] Yang C, Bai L, Zhang C, et al. Bridging collaborative filtering and semi-supervised learning: A neural approach for POI recommendation//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Halifax, Canada, 2017: 1245-1254

[15] Ma H, King I, Lyu M R. Learning to recommend with social trust ensemble//Proceedings of the 32nd International ACM

- SIGIR Conference on Research and Development in Information Retrieval. Boston, USA, 2009; 203-210
- [16] Gao H, Tang J, Liu H. gSCorr: Modeling geo-social correlations for new check-ins on location-based social networks//Proceedings of the 21st ACM International Conference on Information and Knowledge Management. Maui, USA, 2012; 1582-1586
- [17] Li H, Ge Y, Zhu H. Point-of-Interest recommendations; Learning potential check-ins from friends//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, USA, 2016; 975-984
- [18] Wang H, Terrovitis M, Mamoulis N. Location recommendation in location-based social networks using user check-in data//Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. Orlando, USA, 2013; 374-383
- [19] Ye M, Yin P, Lee W C. Location recommendation for location-based social networks//Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. San Jose, USA, 2010; 458-461
- [20] Scellato S, Noulas A, Mascolo C. Exploiting place features in link prediction on location-based social networks//Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego, USA, 2011; 1046-1054
- [21] Yin H, Sun Y, Cui B, et al. LCARS: A location-content-aware recommender system//Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Chicago, USA, 2013; 221-229
- [22] Gao H, Tang J, Hu X, et al. Content-aware point of interest recommendation on location-based social networks//Proceedings of the 29th AAAI Conference on Artificial Intelligence. Austin, USA, 2015; 1721-1727
- [23] Li Y, Nie J, Zhang Y, et al. Contextual recommendation based on text mining//Proceedings of the 23rd International Conference on Computational Linguistics: Posters. Beijing, China, 2010; 692-700
- [24] Gao H, Tang J, Hu X, et al. Exploring temporal effects for location recommendation on location-based social networks//Proceedings of the 7th ACM Conference on Recommender Systems. Hong Kong, China, 2013; 93-100
- [25] Yuan Q, Cong G, Ma Z, et al. Time-aware point-of-interest recommendation//Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 2013; 363-372
- [26] Liu X, Liu Y, Aberer K, et al. Personalized point-of-interest recommendation by mining users' preference transition//Proceedings of the 22nd ACM International Conference on Information & Knowledge Management. San Francisco, USA, 2013; 733-738
- [27] Wang S, Wang Y, Tang J, et al. What your images reveal; Exploiting visual contents for point-of-interest recommendation//Proceedings of the 26th International Conference on World Wide Web. Perth, Australia, 2017; 391-400
- [28] Ge Y, Xiong H, Tuzhilin A, et al. An energy-efficient mobile recommender system//Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Washington, USA, 2010; 899-908
- [29] Lim K H, Chan J, Karunasekera S, et al. Personalized itinerary recommendation with queuing time awareness//Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. Tokyo, Japan, 2017; 325-334
- [30] Zheng Y, Zhang L, Xie X, et al. Mining interesting locations and travel sequences from GPS trajectories//Proceedings of the 18th International Conference on World Wide Web. Madrid, Spain, 2009; 791-800
- [31] Pham T A N, Li X, Cong G. A general model for out-of-town region recommendation//Proceedings of the 26th International Conference on World Wide Web. Perth, Australia, 2017; 401-410
- [32] Wang H, Fu Y, Wang Q, et al. A location-sentiment-aware recommender system for both home-town and out-of-town users//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Halifax, Canada, 2017; 1135-1143
- [33] Zhang F, Yuan N J, Zheng K, et al. Exploiting dining preference for restaurant recommendation//Proceedings of the 25th International Conference on World Wide Web. Montreal, Canada, 2016; 725-735
- [34] Hu Y, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets//Proceedings of the 8th IEEE International Conference on Data Mining. Pisa, Italy, 2008; 263-272
- [35] He X, Zhang H, Kan M Y, et al. Fast matrix factorization for online recommendation with implicit feedback//Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval. Pisa, Italy, 2016; 549-558



**PENG Hong-Wei**, M.S. His major research interests include data mining and recommender system.

**JIN Yuan-Yuan**, Ph.D. candidate. Her major research interests include recommender system and machine learning.

**LV Xiao-Qiang**, M.S. His major research interests include data mining and recommender system.

**WANG Xiao-Ling**, Ph.D., professor, Ph.D. supervisor. Her major research interests include data management technology and data service.

## Background

In recent years, with the popularity of smart phones and wireless networks, people can more easily access their real-time locations, which greatly facilitates their trip and makes the Location-Based Social Network (LBSN, such as restaurants, cinemas, etc.) widely used. People use LBSN to find POIs (Point-of-Interest) they are interested in, and they can also check in or comment on these POIs and share them with their friends. Hence, LBSN contains a lot of useful information, including users' check-in data and rich context information, such as social networks, POI location and category information, etc. The rich context information undoubtedly provides a huge opportunity for personalized POI recommendation.

Although there are many studies for POI recommendations, there are still many problems with the personalized POI recommendation system. On the one hand, users' check-in data in LBSN is very sparse. On the other hand, users' check-in behavior is the result of complex decision-making process, which may be influenced by many factors. Therefore,

to model users' decision-making process is very difficult.

In order to solve the above problem, in this paper, we propose a context-aware POI recommendation model based on matrix factorization. Specifically, we consider the following aspects to model the user's check-in behavior: (1) Users' preference; (2) POI geographic information; (3) Social network. Further, we use different fitting strategies for POIs that users have visited and have not visited, which can more effectively characterize users' preferences, better model users' check-in behavior, and obtain a lower time complexity and good extensibility.

This research is supported by the National Key Research and Development Program of China (No. 2017YFC0803700), the National Natural Science Foundation of China Project Nos. 61532021 and 61472141, the Shanghai Knowledge Service Platform Project (No. ZF1213), and the Shanghai Agriculture Science Program (2016) Number 2-1.