

# 基于用户聚类 and 移动上下文的矩阵分解推荐算法研究

文俊浩 孙光辉 李 顺

(重庆大学软件学院 重庆 401331)

**摘 要** 随着移动互联网技术的快速发展,越来越多的用户通过移动设备获取移动信息和服务,导致信息过载问题日益凸出。针对目前上下文感知推荐算法中存在的稀疏性差、上下文信息融入不够、用户相似性度量被忽略等问题,提出一种基于用户聚类 and 移动上下文的矩阵分解推荐算法。该算法通过利用 k-means 对用户聚类找到偏好相似的用户簇,求出每簇中并对用户所处上下文之间的相似度并对其进行排序,由此找出与目标用户偏好和上下文均相似的用户集合,借助该集合改进传统矩阵分解模型损失函数,并以此为基准进行评分预测和推荐。仿真实验结果表明,所提算法可有效提高预测评分的准确度。

**关键词** 聚类,上下文信息,矩阵分解,推荐

**中图法分类号** TP391 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.04.036

## Study on Matrix Factorization Recommendation Algorithm Based on User Clustering and Mobile Context

WEN Jun-hao SUN Guang-hui LI Shun

(School of Software Engineering, Chongqing University, Chongqing 401331, China)

**Abstract** With the rapid development of mobile Internet technology, more and more individuals use mobile devices to acquire information and services, which makes information overload problem more and more serious. Aiming at the puzzle resulted from data sparsity, insufficient contextual information and ignoring context similarity measurement, this paper proposed a method of matrix factorization recommendation algorithm based on user clustering and mobile context (UCMC-MF) to predict user ratings and make recommendation. Firstly, the method clusters similar user by way of k-means, then finds similar contexts in each cluster, and searches users who are similar to the target user in preferences and context. Finally, experimental results on real datasets demonstrate that the proposed algorithm can effectively improve the accuracy of prediction.

**Keywords** Clustering, Context information, Matrix factorization, Recommendation

## 1 引言

随着移动互联网技术的日趋成熟,移动用户利用各种移动设备(智能手机、PDA等)以及无线通信网络(3G、4G、WiFi)随时随地浏览网页、阅读新闻以及进行社交活动等。然而,由于移动设备具有屏幕尺寸固定、输入能力差、存储空间有限等缺陷,再加上移动环境下服务多样、数量庞大,使得移动用户“信息过载”问题越来越严重,极大地影响了用户体验<sup>[1]</sup>。在如此复杂多变的移动环境下提供满足用户需求的个性化服务,逐渐成为移动推荐领域亟待解决的问题。

移动互联网领域的推荐是当今的热点,也是传统推荐的一种延伸,但比传统互联网推荐更加复杂。与传统互联网推荐相比,移动推荐面临诸多问题,如移动设备屏幕小、输入能力差;移动用户的决策受各种上下文(如网络环境、位置、时间、温度和气候)的影响较大等。结合传统推荐算法和移动环境的特点来有效地解决移动推荐面临的问题,从而为移动用

户产生高质量的推荐,受到了广泛关注。

不少学者对上下文感知推荐系统中的移动算法、上下文建模方式等做了深入的研究<sup>[2-4,6,14]</sup>。其中,基于上下文的推荐主要有基于位置和基于时间的移动推荐。如 Girardello 等人<sup>[3]</sup>利用用户当前的地理位置分析该位置附近的手机中应用程序的使用情况,将使用频繁的应用程序推荐给用户。Tong<sup>[4]</sup>使用朴素贝叶斯分类方法对当前时间上下文进行归类,将其分为工作日和周末,并计算项目在不同类别中的概率,从而进行移动推荐。也有学者将机器学习、数据挖掘等相关技术应用到移动上下文推荐中进行建模<sup>[9,15]</sup>。目前,在移动推荐精度和模型方面仍然存在某些问题,主要表现在:1)上下文引入方式相对比较简单,仅限于上下文过滤或者对某一下上下文进行研究;2)添加上下文进行建模后导致数据更加稀疏;3)上下文建模时,上下文因素与移动推荐模型的融合程度差,忽略了用户偏好信息,仅考虑移动上下文因素。

针对上述问题,本文提出了基于用户聚类 and 移动上下文

到稿日期:2017-03-14 返修日期:2017-05-18 本文受国家自然科学基金(6167060382,61379158)资助。

文俊浩(1969—),男,教授,博士生导师,CCF高级会员,主要研究方向为软件服务工程、个性化推荐,E-mail:jhwen@cqu.edu.cn(通信作者);孙光辉(1990—),女,硕士生,CCF会员,主要研究方向为个性化推荐,E-mail:ghSunOrLux@cqu.edu.cn;李顺(1991—),男,博士,CCF会员,主要研究方向为基于Web服务的个性化推荐。

的矩阵分解推荐算法。该算法在每个簇中按照历史评分记录对相似用户进行聚类,结合用户所在地点和时间等上下文进行过滤,得到目标用户的相似集合,利用该集合优化损失函数,并以添加矩阵分解损失函数正则项的方式约束目标函数,保证矩阵分解预测出的评分与原始数据的误差尽可能小,同时保证在偏好相似的环境下具有相似上下文的用户间的差距也尽可能小,因此既可有效提高推荐的准确率,又可以保证模型具有良好的拟合性。最后在真实数据集上进行仿真实验,结果表明,本文所提算法的推荐准确率较高,并且具有很强的可行性和有效性。

## 2 相关技术

### 2.1 上下文感知推荐技术

上下文感知推荐系统(Context-Aware Recommender Systems, CARS)利用上下文信息在推荐方面的优势,通过获取和预测潜在用户偏好过滤掉不相关的信息,为用户提供满足其个性化需求的结果,逐渐成为缓解“信息过载”的有效手段。Schilit 等人<sup>[7]</sup>将上下文中最重要的方面归纳为:用户的位置、用户身边及周边可以利用的资源。而在移动推荐下,常见的上下文有地理位置、时间、天气、社交关系等。若能有效利用上下文进行推荐,可以进一步提高推荐精度和用户满意度,使得推荐发生在“任何时间、任何地点”,并以任何方式来帮助用户从海量的资源中获取满足其自身需要的服务/项目。Adomavicius 等人<sup>[8]</sup>按照上下文信息在推荐生成过程中所起的作用,将上下文推荐系统分为3类:上下文预过滤、上下文建模、上下文后过滤。推荐系统如图1所示。

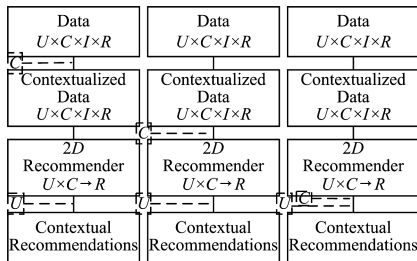


图1 Adomavicius 分类范式:上下文预过滤,上下文后过滤,上下文建模

Fig. 1 Adomavicius classification paradigm: contextual pre-filtering, contextual post-filtering, contextual modeling

### 2.2 矩阵分解技术

在传统的互联网推荐系统中,协同过滤推荐(Collaborative Filtering Recommendation)是应用得最广泛的技术之一。协同过滤分为两类:基于内存的协同过滤和基于模型的协同过滤。基于内存的协同过滤方法在处理数据稀疏性时显得力不从心;而矩阵分解作为典型的基于模型的推荐技术,在处理稀疏性数据时具有一定优势。Netflix 竞赛公布隐因子模型(Latent Factor Model, LFM)能产生良好的推荐结果,且易于扩展,这在学术界和工业界都获得了广泛认可<sup>[5,10,13]</sup>。

矩阵分解的主要思想是将用户评分矩阵分解为低秩的矩阵,使其乘积尽可能逼近原始评分矩阵,以使预测的矩阵与原始矩阵之间的误差平方最小。矩阵分解的核心可以描述为用户-项目的评分矩阵,通过不断地学习得到用户特征矩阵  $U \in R$  和项目特征矩阵  $V \in R$ ,使得  $R \approx UV$ 。为了进一步缩小

预测评分之与真实评分间的差距,设定损失函数  $L$ ,如式(1)所示。在机器学习中,常利用梯度下降法来优化损失函数。

$$L = \sum_{i \in M} \sum_{j \in K} (R_{ij} - U_i^T V_j)^2 \tag{1}$$

但原始的矩阵分解训练过程中很容易出现过拟合以及评分易受到用户或者项目单独影响的问题,从而导致预测评分有较大的误差,精确度较低。当前很多研究采用正则化模型来避免过拟合问题<sup>[5]</sup>,损失函数被改进为式(2):

$$L = \sum_{i \in M} \sum_{j \in K} I_{ij} (R_{ij} - U_i^T V_j)^2 + \lambda_1 \|U\|_F^2 + \lambda_2 \|V\|_F^2 \tag{2}$$

其中,  $I_{ij}$  为指示函数,如果用户对项目有评分则  $I_{ij} = 1$ ,否则为  $I_{ij} = 0$ ;  $\|\cdot\|_F^2$  为弗罗贝尼乌斯范数;  $\lambda_1$  和  $\lambda_2$  为正则化项的系数。

### 3 融合用户聚类和移动上下文的矩阵分解推荐算法

传统推荐模型大部分仅通过分析用户-项目(User-Item)评分矩阵来预测用户评分项的分值,进而将预测评分最高的项目推荐给用户。在基于模型的协同过滤推荐算法中,典型的矩阵分解推荐算法能够将用户和项目的上下文信息融合到模型中,但大多数仅仅只是利用上下文信息进行过滤,忽略或丢失了潜在用户或者项目之间的关联,其中比较重要的就是用户的相似性。而此处的相似不仅指用户偏好的相似,同时还考虑了用户移动上下文环境的相似性。本文同时考虑用户的相似性和上下文环境限制(上下文指地点、时间、用户需求)。假定上下文相似且偏好相似的用户评分应该是相似的。本文结合用户聚类和移动环境下的时间、地点及用户需求等上下文因素,以正则项的形式,利用上下文建模方式对矩阵分解损失函数进行限制,提出了基于用户聚类和移动上下文的矩阵分解推荐算法。该算法的整体框架图如图2所示。

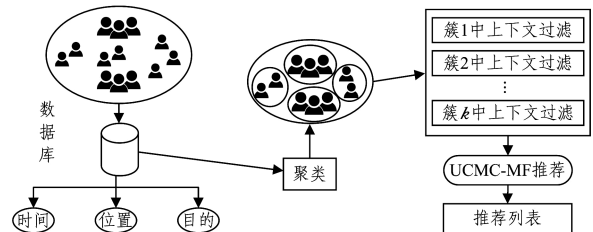


图2 融合用户聚类和移动上下文的矩阵分解推荐算法

Fig. 2 Matrix factorization recommendation algorithm based on user clustering and mobile context

基于聚类和移动上下文的矩阵分解推荐算法思想,先利用 k-means 对用户进行聚类,而后在每簇中利用改进后的矩阵分解推荐求解,可有效地缓解数据稀疏性问题。为此,首先查找所有数据中评分次数最多的  $k$  个用户,并将其设置为最活跃用户;将最活跃的  $k$  个用户初始化为簇质心,根据用户评分,利用欧氏距离计算所有用户到簇中心的距离,不断迭代更新  $k$  簇的质心,直至每簇质心不再发生变化,最终将所有用户划分为  $k$  簇。然后,计算每簇中移动上下文(地点、时间、需求)的相似性,找出其中移动上下文相似的用户集合,最终得到相似环境下偏好相似的用户集合  $F(i)$ 。假定所筛选出来的用户集合与目标用户是极其相似的,则在保证矩阵分解后用户特征因子矩阵和项目特征因子矩阵的乘积与原始评分矩阵尽可能接近的条件下,还须保证所筛选出来的相似用户特征因子之间也尽可能相近。本文借助  $F(i)$  改进矩阵分解函

数,同时考虑相似用户之间的强关联性,对损失函数(2)添加正则项约束,直接进行移动上下文信息的推荐。

### 3.1 基于用户相似度的聚类

为缓解移动推荐的稀疏性,本文并非对将直接使用的所有用户评分数据进行矩阵分解,而是首先对所有用户的评分数据进行聚类,然后利用矩阵分解预测每簇局部数据中的用户评分,进而进行推荐。时下比较流行的聚类算法较多,鉴于k-means聚类算法速度快、结构简单,文中利用k-means对所有移动用户的历史评分数据进行聚类,找出出偏好相似的用户簇。其具体聚类步骤如下:

Step1 统计用户历史评分数据,找出评分最多的前 $k$ 个用户,即比较活跃的用户,并将其初始化为聚类的初始质心。

Step2 假定用户评分表示为用户间的物理距离,由此遍历所有数据,利用式(3)计算所有其他用户到各个质心的距离,并根据距离的大小将各个用户划分至与其距离最近的质心所在的簇中。

$$E = \sum_{i=1}^k \sum_{x \in C_i} |c_i - x|^2 \quad (3)$$

其中, $c_i$ 为每个簇的质心, $x$ 为每个用户的评分。

Step3 重新计算 $k$ 个簇的质心,簇中所有对象值的中心即为质心,计算公式如式(4)所示:

$$c_i = \frac{1}{m} \sum_{x \in C_i} x \quad (4)$$

其中, $c_i$ 为每个簇的质心, $m$ 为每簇中用户的个数, $x$ 为每个用户的评分。

Step4 同样重复 Step2 和 Step3,经过多次迭代计算、反复调整后,最终将数据对象划分到 $k$ 个簇中,直到质心不再发生变化为止。

### 3.2 上下文相似度的计算

移动环境下,移动用户的信息需求决策不仅仅受一种上下文的影响,而是受多种上下文的作用。在复杂上下文中,难以确定某个上下文对用户影响的重要程度,因此文中综合考虑了多种上下文因素的影响。为缓解添加上下文后数据变得更稀疏的问题,可利用前述聚类结果对每簇用户数据分别进行局部研究。假定同一簇中上下文的相似性越高,推荐的结果就越相近。文献[11]提出了3种计算上下文相似性的方法:独立上下文相似性(Independent Context Similarity, ICS)、潜在上下文相似性(Latent Context Similarity, LCS)、多维上下文相似性(Multidimensional Context Similarity, MCS)。其中,ICS假定不同上下文条件之间是相互独立的,只考虑具有相同条件的上下文环境,把上下文条件下相似度的乘积作为两个上下文的相似性。文中采用ICS思路,假定各个移动上下文条件是相互独立的,将各个上下文相似度的和作为最终用户上下文的相似性;然后,将该思想应用于3.1节的聚类结果中,计算每簇中用户所在上下文的相似度。用 $C = \{C_1, C_2, C_3, \dots, C_n\}$ 表示 $n$ 种上下文因素;用 $Sim(C_1, C_2, \dots, C_n)$ 表示当前簇中用户 $u_i$ 和用户 $u_j$ 的上下文相似度,其具体计算方法如式(5)所示:

$$Sim(C_1, C_2, \dots, C_n) = \sum_{i=1}^n Sim_{C_i}(u_i, u_j) \quad (5)$$

本文所涉及的上下文 $C = \{C_{location}, C_{date}, C_{time}\}$ 中, $C_{location}$ 表示用户的位置信息; $C_{date}$ 表示用户的日期信息,即周末和工作

日; $C_{time}$ 表示用户一天中的时间,根据具体时间段可以划分为:早上时间、午饭时间、下午时间、晚饭时间、晚上。文中将地点作为硬性规定的上下文,即地点相同则认为 $Sim_{C_{location}}(u_1, u_2)$ 为1,否则为0;时间上下文 $C_{time}$ 指一天中都可以利用的时间,它是影响用户抉择的主要因素,采用Person系数来计算时间上下文的相似度,如式(6)所示:

$$Tsim(u_i, u_j) = \frac{\sum_{p \in P} (t_{u_i, p} - \bar{t}_{u_i})(t_{u_j, p} - \bar{t}_{u_j})}{\sqrt{\sum_{p \in P} (t_{u_i, p} - \bar{t}_{u_i})^2} \sqrt{\sum_{p \in P} (t_{u_j, p} - \bar{t}_{u_j})^2}} \quad (6)$$

其中, $t_{u_i, p}$ 表示 $u_i$ 使用项目/服务的时间, $\bar{t}_{u_i}$ 表示用户 $i$ 正常使用/项目的平均时刻。 $t_{u_j, p}$ 表示 $u_j$ 使用项目/服务的时间, $\bar{t}_{u_j}$ 表示用户 $j$ 正常使用/项目的平均时间。

在用户偏好相似簇中,采用上述方法计算出用户上下文的相似性,并找出每簇中与目标用户上下文比较相近的用户集合 $F(i)$ 。

### 3.3 基于用户聚类和移动上下文的矩阵分解模型

在移动推荐中,有效利用上下文信息可改善推荐的效果,为用户提供合适的个性化服务<sup>[15-16]</sup>。但大多数传统的推荐算法只是通过上下文对数据进行过滤,加剧了数据稀疏性问题。为此,文中结合用户聚类和移动上下文的信息相似性,提出了改进措施,利用用户聚类和移动上下文的相似性可得到更准确的用户相似集合。利用正则项的方式对式(2)进行改进以防止过拟合,在每次迭代分解时,通过综合考虑上次分解出的用户因子及与其相似的其他用户因子来对损失函数进行优化。

基于用户聚类和移动上下文的推荐算法,先利用3.2节讨论的方法对每簇用户上下文的相似度进行计算,取每簇中与目标用户上下文相似的前 $k$ 个用户-近邻用户集合 $F(i)$ ,并以正则项式(7)的形式将其添加到损失函数中,对预测评分进行约束。

$$g(i) = \frac{\alpha}{2} \sum_{i=1}^m \|U_i - \frac{1}{|F(i)|} \sum_{f \in F(i)} U_f\|_F^2 \quad (7)$$

为进一步保证 $R \approx UV$ 更加逼近,除添加由移动上下文限制的正规化项式(7)外,在每次迭代分解中,还综合考虑了上次分解出的用户因子及与其相似的其他用户因子,借助平衡参数 $\mu$ ,融合分解后的用户因子评分信息和局部与用户因子相似评分信息对损失函数进行优化,得到改进的损失函数式(8)。即每簇中的用户通过矩阵分解进行评分预测时,除保证了分解出的用户因子矩阵和项目因子矩阵的乘积尽可能与原始评分相近外,还保证了用户 $u_i$ 与其偏好相似的近邻用户的评分尽可能小,从而使预测后的评分更加准确。

$$L = \sum_{i \in M} \sum_{j \in K} (R_{ij} - (\mu U_i^T V_j + (1 - \mu) \sum_{k \in F(i)} S_k U_k^T V_j))^2 + \lambda_1 \|U\|_F^2 + \lambda_2 \|V\|_F^2 + \frac{\alpha}{2} \sum_{i=1}^m \|U_i - \frac{1}{|F(i)|} \sum_{f \in F(i)} U_f\|_F^2 \quad (8)$$

其中, $R_{ij}$ 表示用户的实际评分, $U_i$ 为用户因子矩阵, $V_j$ 为项目因子矩阵, $\alpha$ 为学习参数, $F(i)$ 为目标用户偏好和上下文及其相似用户集合, $\mu$ 为目标用户因子与其局部相似用户因子之间的平衡参数。

利用梯度下降法对改进后的损失函数式(8)进行迭代求解,直至损失函数收敛为止,从而求出用户因子矩阵和项目因子矩阵。但在以矩阵分解为基础的推荐算法的预测评分中普

遍存在预测评分超过评分上下限的不合理情况,对此文中借助映射思想给出式(9)和式(10),使之无论高出评价分还是低于评价分,用户的评分一般都会在平均值左右浮动。

$$r_u = \bar{r}_u + |r_p - \bar{r}_u| \times \frac{(r_{\max} - \bar{r}_u)}{(r_{p\max} - \bar{r}_u)} \quad (9)$$

$$r_u = \bar{r}_u + |r_p - \bar{r}_u| \times \frac{(\bar{r}_u - r_{\min})}{(\bar{r}_u - r_{p\min})} \quad (10)$$

其中,  $r_u$  为用户最终的评分值,  $\bar{r}_u$  为用户历史评分的平均值,  $r_{\max}$  为历史评分的最大值,  $r_{p\max}$  为用户预测评分的最大值,  $r_{\min}$  为历史评分的最小值,  $r_{p\min}$  为用户预测评分的最小值。

### 4 实验及其分析

为了验证本文提出的基于用户聚类 and 移动上下文的矩阵推荐算法的性能,基于真实数据集进行了仿真实验。

#### 4.1 数据集

实验数据来自 Hyunchul Ann<sup>[12]</sup> 教授收集的真实移动用户数据。该数据集包含移动用户在不同环境下对不同项目评分。数据包含了 3419 个用户于不同时间在韩国的 5 个大区域利用移动设备搜索饮食、娱乐、住宿、购物以及学校的不同场合地点数和用户最后对地点的评分数据,详细说明如表 1 所列。

表 1 数据集说明

Table 1 Description of dataset

数据集	字段大小	备注说明
用户评分总数	99803 条	99803 条
评分值	1-7	分别代表 Very dissatisfied, Dissatisfied, SomeWhat dissatisfied, Neutral, SomeWhat Satisfied, Satisfied, Very Satisfied
日期	1,2	分别代表工作日,周末
时间	1-5	分别代表(上午(08:00am-11:00am),午饭时间(11:00-02:00pm),下午(02:00-05:00pm),晚饭时间(05:00-08:00pm),晚上
需求目的	1,2,3	娱乐性,功利性,两者都有

#### 4.2 评价指标

衡量个性化推荐系统的一种重要方式是评分预测。通常情况下,利用 MAE(平均绝对误差)、RMSE(均方根误差)计算评分预测系统的准确度。MAE 和 RMSE 的定义如式(11)和式(12)所示:

$$MAE = \frac{1}{|T|} \sum_{n * m \in T} |R_{n * m} - \hat{R}_{n * m}| \quad (11)$$

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{n * m \in T} |R_{n * m} - \hat{R}_{n * m}|^2} \quad (12)$$

其中,  $R_{ui}$  表示用户的实际评分,  $\hat{R}_{ui}$  表示用户的预测评分,  $|T|$  表示用户集合数量。

从式(11)和式(12)可以看出,MAE 和 RMSE 越小,预测的精确度越高,算法越有效。

#### 4.3 实验分析

1)在实验中,将数据集分为训练集和测试集(随机地抽取 80% 的数据作为训练集,剩下的 20% 作为测试集)进行实验。设用户/项目特征因子维度为 10,平衡参数  $\mu$  为 0,在循环多次迭代的情况下,为缓解数据稀疏性,UCMC-MF 算法经过多次迭代来训练合适的聚类簇数。设置簇数为 0,5,10,15,采用基础矩阵分解算法(Basic MF)、协同过滤推荐算法(Collaborative Filtering,CF)、规则化矩阵分解推荐算法(Regulari-

zed MF)以及本文提出的 UCMC-MF 推荐算法分别进行了仿真实验。不同算法在不同簇数量下的 MAE 和 RMSE 的比较结果如图 3 和图 4 所示。

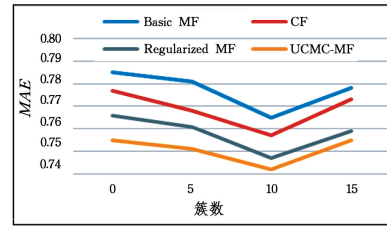


图 3 不同簇数下不同算法的 MAE 值的比较

Fig. 3 Comparison of MAE values for different algorithms and different cluster numbers

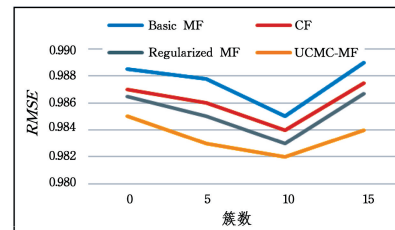


图 4 不同簇数下不同算法的 RMSE 值的比较

Fig. 4 Comparison of RMSE values for different algorithms and different cluster numbers

分析图 4 和图 5 中 Basic MF,CF,Regularized MF 以及本文提出的 UCMC-MF 推荐算法的 MAE 和 RMSE 的变化趋势可知,聚类后的结果有所降低,且当簇数取为 10 时效果最佳。开始阶段,随着簇数增多,MAE 和 RMSE 逐渐减小,当簇数达到一定值后 MAE 和 RMSE 反而增加。其原因在于,簇数太多易造成每簇下的用户个数太少,从而引起过拟合问题,导致准确率不高。

2)在用户/项目因子矩阵维度为 30、聚类簇数为 10 的条件下,对目标用户因子和与目标用户因子相似的用户因子平衡参数  $\mu$  进行训练。设置平衡参数  $\mu$  的值分别为 0,0.2,0.4,0.6,0.8 和 1,对 UCMC-MF 算法进行训练。不同平衡参数下的 MAE 和 RMSE 值如图 5 和图 6 所示。

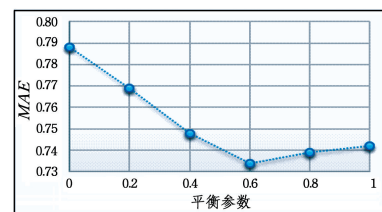


图 5 不同平衡参数下 UCMC-MF 的 MAE

Fig. 5 MAE of UCMC-MF with different equilibriums

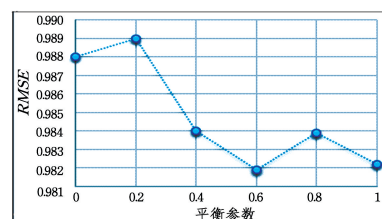


图 6 不同平衡参数下 UCMC-MF 的 RMSE

Fig. 6 RSME of UCMC-MF with different equilibriums

通过分析图 5 和图 6 可知,在不同的平衡参数下,UCMC-MF 算法的 MAE 和 RMSE 呈急速下降的趋势,表明有效结合上次的迭代效果有助于提高算法的准确率,并且在平衡参数约为 0.6 时效果最佳。

3)在用户/项目因子矩阵维度为 30、聚类簇数为 10、平衡参数  $\mu$  为 0.6 的条件下,通过多次迭代,UCMC-MF 与 Basic MF,Regularized MF 以及协同过滤算法(CF)的 MAE 和 RMSE 的对比结果如表 2 所列。

表 2 各算法在不同迭代次数下的 MAE 和 RMAE 比较

Table 2 Comparison of MAE and RMAE of different algorithms under different iteration times

评价标准	算法名称	迭代次数			
		5000	10000	15000	20000
MAE	CF	0.7913	0.7573	0.7572	0.7573
	Basic MF	0.7512	0.7651	0.7634	0.7636
	Regularized MF	0.7662	0.7469	0.7921	0.7917
	UCMC-MF	0.7493	0.7341	0.7415	0.7416
RMSE	CF	0.9913	0.9843	0.9846	0.9845
	Basic MF	0.9892	0.9854	0.9851	0.9852
	Regularized MF	0.9854	0.9827	0.9826	0.9827
	UCMC-MF	0.9827	0.9815	0.9816	0.9819

由表 2 可以看出,不同迭代次数下,基于用户聚类 and 移动上下文的 UCMC-MF 推荐算法的准确率都相对较高,且比较稳定。

4)针对矩阵分解预测评分出现 3.3 节所提到的评分超出上下限的问题,文中给出了优化方案,结果处理前与结果处理后的比较如图 7 和图 8 所示。经处理后的结果对预测效果有所改善,说明借助映射思想处理预测评分超出上下限的问题是可行且有效的。

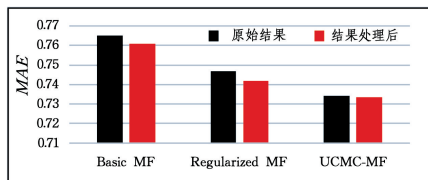


图 7 结果处理后与处理前值的比较

Fig. 7 Comparison of processed results and non-processed results

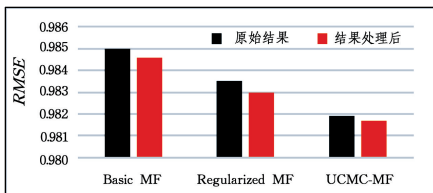


图 8 结果进行处理后与处理前的比较

Fig. 8 Comparison of processed results and non-processed results

**结束语** 针对移动环境下存在的信息过载等问题,本文提出了一种基于用户聚类 and 移动上下文的矩阵分解推荐算法。该算法首先在用户评分矩阵中找出与目标用户相似的用户簇,以添加正则项的方式进行矩阵分解;然后对结果进行规范化处理;最后在真实数据上进行仿真实验。实验结果表明,该算法可有效提高预测评分的准确度。与传统基于矩阵分解

推荐的上下文感知推荐方法相比,该算法利用聚类等方式缓解了由于添加上下文因素而导致的数据稀疏性问题,并结合多种移动上下文等因素的影响进行建模,因此推荐效果比较理想。由于移动环境下上下文关系复杂,对于不同上下文对用户影响程度不同的情况,仍需进一步深入研究。

参 考 文 献

[1] MENG X W, HU X, WANG L C, et al. Mobile Recommender Systems and Their Applications[J]. Journal of Software, 2013, 24(1):91-108. (in Chinese)  
孟祥武,胡勋,王立才,等. 移动推荐系统及其应用[J]. 软件学报, 2013, 24(1):91-108.

[2] WANG Z M, YANG F. An optimized location-based mobile restaurant recommend and navigation system[J]. Wseas Transactions on Information Science & Applications, 2009, 6(5): 809-818.

[3] GIRARDELLO A, MICHAHELLES F. AppAware: which mobile applications are hot? [C]// Conference on Human-Computer Interaction with Mobile Devices and Services (Mobile HCI 2010). Lisbon, Portugal, DBLP, 2010:431-434.

[4] TONG Q L, PARK Y, PARK Y T. A time-based approach to effective recommender systems using implicit feedback[J]. Expert Systems with Applications, 2008, 34(4): 3055-3062.

[5] SALAKHUTDINOV R, MNH A. Probabilistic matrix factorization[C] // International Conference on Machine Learning. 2007:880-887.

[6] TU D D, SHU C C, YU H Y. Using Unified Probabilistic Matrix Factorization for Contextual Advertisement Recommendation [J]. Journal of Software, 2013, 24(3): 454-464. (in Chinese)  
涂丹丹,舒承椿,余海燕. 基于联合概率矩阵分解的上下文广告推荐算法[J]. 软件学报, 2013, 24(3): 454-464.

[7] SCHILIT B, ADAMS N, WANT R. Context-Aware Computing Applications[C]// First Workshop on Mobile Computing Systems and Applications. IEEE Computer Society, 1994:85-90.

[8] ADOMAVICIUS G, TUZHILIN A. Context-aware recommender systems[C]// ACM Conference on Recommender Systems. ACM, 2008:335-336.

[9] HAI B Z, XIE R Y. Bayesian Network-based Context-aware Recommendation Algorithm [J]. Computer Science, 2014, 41(7):275-278. (in Chinese)  
海本斋,解瑞云. 基于贝叶斯网络的上下文推荐算法[J]. 计算机科学, 2014, 41(7):275-278.

[10] ZHENG Y, MOBASHER B, BURKE R. Incorporating context correlation into context-aware matrix factorization[C]// International Workshop on Intelligent Personalization. 2015:21-27.

[11] MA H, ZHOU D, LIU C, et al. Recommender systems with social regularization[C]// Forth International Conference on Web Search and Web Data Mining (WSDM 2011). Hong Kong, China, DBLP, 2011:287-296.

能够改善推荐效果。

由于本文在分析项目主题相关性时仅考虑了标签的共现关系与标注频次,因此在后续的研究中将尝试在本文算法的基础上结合标签语义分析与标签重要程度来挖掘项目主题,更加准确地分析项目主题相似度,以提高推荐的质量。

### 参 考 文 献

- [1] BOBADILLA J, ORTEGA F, HERNANDO A, et al. Recommender systems survey[J]. Knowledge-based Systems, 2013, 46(1):109-132.
- [2] RESNICK P, IACOVU N, SUCHAK M, et al. GroupLens: an open architecture for collaborative filtering of netnews[C]//ACM Conference on Computer Supported Cooperative Work. ACM, 1994:175-186.
- [3] LINDEN G, SMITH B, YORK J. Amazon.com recommendations: item-to-item collaborative filtering [J]. IEEE Internet Computing, 2003, 7(1):76-80.
- [4] FENG W, WANG J. Incorporating heterogeneous information for personalized tag recommendation in social tagging systems [C]//ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2012:1276-1284.
- [5] ZHANG B, GAO Y, GAO K N, et al. Combining Relation and Content Analysis for Social Tagging Recommendation[J]. Journal of Software, 2012, 23(3):476-488. (in Chinese)  
张斌,高引,高克宁,等.融合关系与内容分析的社会标签推荐[J].软件学报,2012,23(3):476-488.
- [6] SONG H, LU P, ZHAO K. Improving item-based collaborative filtering recommendation system with tag [C]//International Conference on Artificial Intelligence, Management Science and Electronic Commerce. IEEE, 2011:2142-2145.
- [7] CAI Q, HAN D M, LI H S, et al. Personal Resource Recommendation Based on Tags and Collaborative Filtering[J]. Computer Science, 2014, 41(1):69-71. (in Chinese)  
蔡强,韩东梅,李海生,等.基于标签和协同过滤的个性化资源推荐[J].计算机科学,2014,41(1):69-71.
- [8] LI G, WANG S, LI Z Y, et al. Personalized Tag Recommendation Algorithm Based on Tensor Decomposition[J]. Computer Science, 2015, 41(4):30-35. (in Chinese)  
李贵,王爽,李征宇,等.基于张量分解的个性化标签推荐算法[J].计算机科学,2015,41(4):30-35.
- [9] WANG W P, WANG J H. Hybrid Recommendation Method Based on Tag and Collaborative Filtering[J]. Computer Engineering, 2011, 37(14):34-35. (in Chinese)  
王卫平,王金辉.基于Tag和协同过滤的混合推荐方法[J].计算机工程,2011,37(14):34-35.
- [10] KIM H N, JI A T, HA I, et al. Collaborative Filtering Based on Collaborative Tagging for Enhancing the Quality of Recommendation [J]. Electronic Commerce Research & Applications, 2010, 9(1):73-83.
- [11] HALPIN H, ROBU V, SHEPHERD H. The complex dynamics of collaborative tagging[C]//International Conference on World Wide Web(WWW 2007). Banff, Alberta, Canada, May. DBLP, 2007:211-220.
- [12] WU C, ZHOU B. Complex network analysis of tag as a social network[J]. Journal of Zhejiang University (Engineering Science), 2010, 4(11):2194-2197. (in Chinese)  
吴超,周波.基于复杂网络的社会化标签分析[J].浙江大学学报:工学版,2010,4(11):2194-2197.
- [13] AHN Y, BAGROW J, LEHMANN S. Link Communities Reveal Multiscale Complexity In networks [J]. Nature, 2010, 466(7307):761-764.
- [14] GroupLens[EB/OL]. <http://www.grouplens.org>.
- [15] SARWAR B, KARYPIS G, KONSTAN J, et al. Item-based collaborative filtering recommendation algorithms [C]//International Conference on World Wide Web. ACM, 2001:285-295.
- [16] JI H, LI J, REN C, et al. Hybrid collaborative filtering model for improved recommendation [C]//IEEE International Conference on Service Operations and Logistics, and Informatics. IEEE, 2013:142-145.
- [12] DAO T H, JEONG S R, AHN H. A novel recommendation model of location-based advertising: Context-Aware Collaborative Filtering using GA approach[J]. Expert Systems with Applications, 2012, 39(3):3731-3739.
- [13] LIU R S, YANG T C. Improving Recommendation Accuracy by Considering Electronic Word-of-Mouth and the Effects of Its Propagation Using Collective Matrix Factorization [C]//IEEE Datacom. IEEE, 2016.
- [14] ZHEN G L, ZHU F X, YAO X. Recommendation Rating Prediction Based on Attribute Boosting with Partial Sampling[J]. Chinese Journal of Computers, 2016, 39(8):1501-1514. (in Chinese)  
郑麟,朱福喜,姚杏.基于属性提升与局部采样的推荐评分预测[J].计算机学报,2016,39(8):1501-1514.
- [15] EL-MOEMEN S A, HASSAN T, SEWISY A A. A Context-Aware Recommender System for Personalized Places in Mobile Applications [J]. International Journal of Advanced Computer Science & Applications, 2016, 7(3):442-448.
- [16] CHAMPIRI Z D, SHAHAMIRI S R, SALIM S S B. A systematic review of scholar context-aware recommender systems [J]. Expert Systems with Applications, 2015, 42(3):1743-1758.

(上接第219页)