【服务与运作管理】

个性化推荐系统中协同过滤推荐算法优化研究

关 菲 周 艺 张 晗

【摘 要】协同过滤推荐算法是目前个性化推荐系统中应用比较广泛的一种算法。然而,它在处理数据稀疏性、可扩展性等方面存在一定不足。针对数据稀疏性问题,本文首先基于 Slope One 算法对初始的评分矩阵进行缺失值填充,其次利用基于 K-means 聚类的协同过滤算法预测目标用户的评分,并结合 MovieLens 数据集给出了相关对比实验;针对扩展性问题,本文首先提出了一种基于中心聚集参数的改进 K-means 算法,其次,给出了基于中心聚集参数改进 K-means 的协同过滤推荐算法流程,并结合 MovieLens 数据集设计了相关对比实验。实验结果表明,本文所提方法推荐精度均得到显著提高,数据稀疏性和扩展性问题得到了有效改善。因此,本文的研究结论不仅可进一步丰富协同过滤推荐算法的现有理论成果,还可以为提高推荐系统的精度提供理论依据和决策参考。

【关 键 词】推荐系统;决策;协同过滤;中心聚集参数;K-means 聚类

【作者简介】关菲(1985-),女,安徽人,河北经贸大学数学与统计学学院,副教授,博士,研究方向:模糊对策与决策,数据挖掘与推荐系统(河北 石家庄 050061);周艺,河北经贸大学数学与统计学学院(河北 石家庄 050061);张晗,河北经贸大学数学与统计学学院(河北 石家庄 050061)。

【原文出处】《运筹与管理》(合肥),2022.11.9~14

【基金项目】本文得到河北省高等学校科学技术研究计划项目(BJ2020011,ZD2021319);国家自然科学基金项目(71701001,71803037);河北经贸大学科研基金一般资助项目(2020YB01)。

0 引言

大数据时代下互联网的应用使人们的生活更加便捷与智能化。但随之而来的“信息过载”问题亟需解决,此时推荐系统应运而生。协同过滤[1]作为个性化推荐系统[2]中应用较为广泛的一种算法,因经常面临数据稀疏性和算法扩展性问题,近年来备受关注。因此,协同过滤算法的优化成为了国内外学者的研究热点。针对数据稀疏性问题,一些学者通过降维的方法来缓解稀疏性,比如文献[3]提出了基于矩阵分解的迭代最小二乘加权正则化协同过滤算法,对传统矩阵分解模型加入正则化约束以防止过拟合;文献[4]提出了基于矩阵分解和随机森林的多准则推荐算法;文献[5]利用自适应神经模糊推理系统,结合减法聚类和高阶奇异值分解提出了一种新的多准则协同过滤模型。还有一些学者通过矩阵填充来缓解数据稀疏性,比如文献[6]考虑了用户对项目的兴趣偏好,提出了一种结合用户兴趣偏好的矩阵填充方法;文献[7]通过非负约束下的低秩矩阵填充模型对矩阵缺失项取值进行预测。还有一些学者通过融合一些其他信息来缓解数据稀疏性带来的问题,比如文献[8~10]分别在推荐过程中融合了标签信息和时间效应、物品属性和用户间的社交联系、上下文对用户的影响以及项目的相关性等,均提高了推荐系统的准确性。针对算法扩展性问题,文献[11~13]分别采用了 K-means 聚类、二分 K-means 聚类、模糊 K-means 聚类先对物品进行聚类再推荐的方法,进而提高了推荐的精度和可扩展性;文献[14]提出了一种基于聚类矩阵近似的协同过滤推荐算法,通过对稠密矩阵块进行奇异值分解以及对各个稠密矩阵块奇异向量进行 Schmidt 正交来对原始评分矩阵进行重新近似;文献[15]将双聚类与协同过滤相结合,将双聚类的局部相似度和全局相似度结合在一起,构造了一个新的候选项目排名得分;分析上述文献可以看出:学者们大多采用降维、矩阵填充和融合其他信息的方法来缓解数据稀疏性。对于扩展性问题,学者们经常采用 K-means 聚类方法,然而 K-means 聚类存在一定的不足之处,可能会出现因聚类的用户簇不合理,从而

导致推荐效果较差的现象。基于此,本文将从数据稀疏性和扩展性两个角度对协同过滤算法进行优化。

本文结构安排如下:第1部分介绍了本文所需用到的Slope One算法和基于K-means聚类的协同过滤推荐算法。第2部分提出了一种基于Slope One矩阵填充的协同过滤推荐优化算法。第3部分提出了一种基于中心聚集参数的改进K-means算法,并将该算法应用到协同过滤推荐中。第4部分,结论与研究展望,总结了本文算法的有效性并讨论了本文下一步的研究方向。

1 预备知识

1.1 Slope One算法

2005年,Daniel Lemire教授提出了Slope One算法[16]。其基本原理是计算不同物品间的一个评分差,用评分差预测最终用户对物品的评分。若将整个评分数据标记为R,主要分以下两步:

①在两个物品同时被评分的前提下,将两物品i、j的评分差取均值,记做评分偏差:

$$R(ij)=\frac{\sum_{u\in N(i)\cap N(j)}(r_{ui}-r_{uj})}{|N(i)\cap N(j)|} \tag{1}$$

其中,r_{ui}和r_{uj}表示用户u对项目i、j的评分,$N(i)$是对物品i评过分的用户,$|N(i)\cap N(j)|$是对物品i、j都评过分的用户数。

②根据用户历史评分和由(1)得出的评分偏差,预测用户对没有做出过评分的物品的分值。

$$p_{uj}=\frac{\sum_{i\in N(u)}|N(i)\cap N(j)|(r_{ui}-R(ij))}{\sum_{i\in N(u)}|N(i)\cap N(j)|} \tag{2}$$

1.2 基于K-means聚类的协同过滤推荐算法

设$X=\{x_i|x_i\in R^p,i=1,2,\cdots,n\}$为原数据集,每个数据由用户、项目、评分3部分组成。设目标用户为u,用户集合是$U=\{u_1,u_2,\cdots,u_m\}$,K-means聚类得出的用户集合可以表示为$U=\{C_1,C_2,\cdots,C_k\}$,k为聚类个数。算法步骤[17]如下:

(1)在X中随机选出k个样本作为初始簇心$M_i(i=1,2,\cdots,k)$;

(2)用初始簇心$M_i(i=1,2,\cdots,k)$对用户-项目评分矩阵$R_{m\times n}$执行经典K-means算法,得到k个类;

(3)使用欧氏距离计算目标用户u与k个簇心间的距离,根据最小距离,找到u所属的类别;

(4)在u所属类中,计算u与其他用户的相似性,获取u的最近邻居集$N_{uj}(j=1,2,\cdots,m)$;

(5)得到最近邻居集后,预测求得想要推荐给u的项目的评分,由高到低排序后,把前N个项目推荐给u。

2 基于Slope One的协同过滤推荐优化算法

为有效缓解数据稀疏性,本节中我们选取MovieLens 100k数据集,数据集详情如表1。

表1 MovieLens100k数据集基本信息

	MovieLens100k
用户数	943
电影数	1682
评分数	100000
用户平均评分	3.51
稀疏度(%)	93.7

(实验环境为Windows 8系统;硬件条件为CPU2.3GHZ,4G内存,100G;使用软件Python 3.7版本)我们选取9种不同方法进行稀疏矩阵的填充,并选用均方根误差(RMSE)来判定有效性。

参照文献[18],我们用全局均值法、用户平均法、物品平均法、用户活跃度 & 物品不平均、用户活跃度 & 物品平均、用户不平均 & 物品流行度、用户平均 & 物品流行度、用户活跃度 & 物品流行度、Slope One这9种不同方法进行填充,其均方根误差见表2。

表 2 MovieLens 数据集上不同填充方法的均方根误差

用户集	物品集	均方根误差
全局平均		1.1239
平均	不平均	1.0334
不平均	平均	0.9798
活跃度	不平均	1.1163
活跃度	平均	0.9779
不平均	流行度	1.0961
平均	流行度	1.0043
活跃度	流行度	1.0918
	SlopeOne	0.9507

从表2 的结果可以看出,当使用 Slope One 方法进行填充时,其 RMSE 是最小的,说明使用该方法填充时误差较小。下面我们将采用 Slope One 方法对 MovieLens 100k 版本数据集进行填充,结果见表 3。

表 3 用 Slope One 填充后 MovieLens100k 信息表

	MovieLens100k
用户数	943
电影数	1682
评分数	1484961
用户平均评分	3.83
稀疏度(%)	6.37

由表 3 可知,进行了矩阵填充后,数据集的稀疏度变为 6.37%,说明运用 Slope One 填充的效果明显。我们使用填充后数据设计了两组对比实验,以验证有效性:

实验 1 确定最佳聚类个数。在这一阶段的实验中,分别利用由 Slope One 填充前和填充后的数据,得到两个最佳 K-means 聚类个数。在确定过程中,我们选取平均绝对误差(MAE)作为测量标准,当聚类个数以 2 为间隔,由 2 增加到 20 时,根据 MAE 的大小来确定出最佳聚类个数,为了保证推荐时选取的近邻数相同,将其固定为 25,以此增强 MAE 值的可靠性。图 1 中纵坐标代表 MAE 值,横坐标是聚类个数。

实验结果表明:图 1 中随着聚类个数从 2 增加到 20,在进行 SlopeOne 填充和未填充时,K-means 不同聚类个数的 MAE 值都呈现出先下降后上升的趋势,且波动幅度较小。两种情况的 MAE 最低值都出现在 $k=8$,因此最佳的聚类个数都是 8。

实验 2 推荐算法精度对比。我们采用三种算法来做对比试验,分别是:基于 SlopeOne 填充后的 K-means 协同过滤推荐、经典 K-means 协同过滤推荐和传统的协同过滤推荐算法。图 2 中纵轴是 MAE 值,横轴是近邻个数,以 5 为间隔从 5 增加到 50,结果如图 2 所示。

由图 2 可以看出,三种协同过滤推荐算法的 MAE 值都随着最近邻个数的增加呈现出缓慢下降的趋势,且在邻居个数变化相同时,用 Slope One 填充后的 K-means 聚类协同过滤算法的 MAE 值最小。因此,在推荐之前,先对评分矩阵进行填充,这样能增加一些用户评分数据,以便在推荐过程中发掘出更多的可供推荐的项目,填充完毕的矩阵再对用户进行 K-means 聚类,这样是为了能缩小近邻寻找范围,类中的所有用户相比类外的用户与目标用户具有更高的相似程度,在后续的近邻匹配选取时也就更便利。

3 基于改进 K-means 的协同过滤推荐优化算法

3.1 基于中心聚集参数的改进 K-means 算法

由于第 2 节中用到的传统 K-means 聚类算法的初始中心和聚类个数 k 均是随机产生的,其合理性会影响聚类结果。因此,本节中我们参考文献[19],首先给出如下定义:

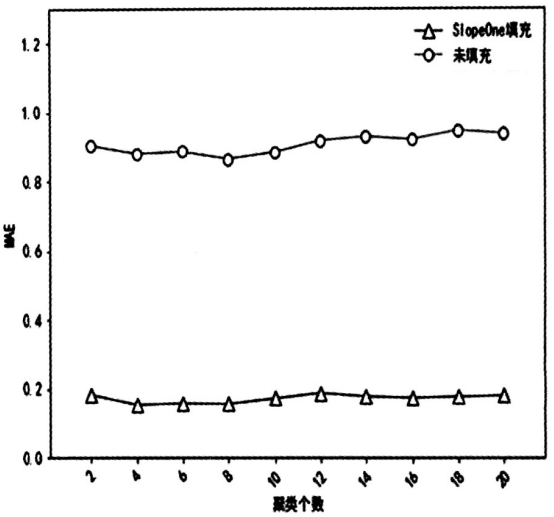

图1 K-means 算法不同聚类个数的 MAE 变化对比图

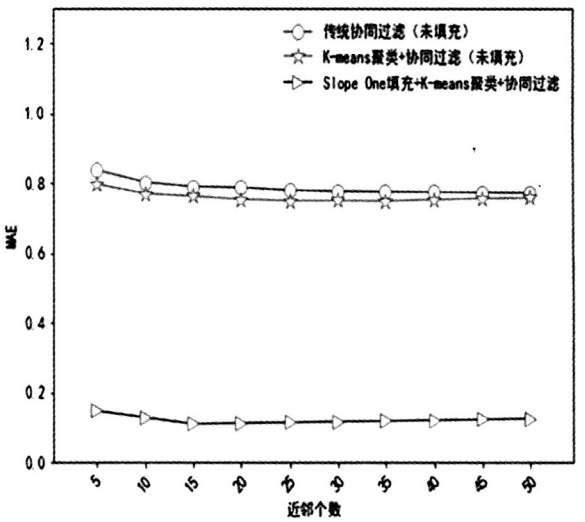

图2 不同近邻个数 MAE 值对比

定义 1 任意两个数据点间距离的平均值为：

$$Avgd(D)=\frac{2}{n(n-1)}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}d(x_i,x_j) \tag{3}$$

其中，$d(x_i,x_j)$是任意两点之间的欧式距离，n 表示数据点的个数。

定义 2 定义邻域半径 R 为：

$$R=\frac{Avgd(D)}{n^{releR}} \tag{4}$$

releR 是一个用来调节的系数，*releR* 取 0.13 时，聚类效果最好。

定义 3 点的聚集度定义：

$$Dp(x_i)=\sum_{j}^{n}f(d_{ij}-R) \tag{5}$$

其中 $f(x)=\begin{cases}1,x<0\\0,x>0\end{cases}$，$Dp(x_i)$表示与点 x_i 距离小于邻域半径 R 的点的个数，用来衡量 x_i 周围的密集程度，$Dp(x_i)$值越大，代表周围点越多，越密集。

定义 4 簇类平均距离定义为：

$$Gavgd(x_i)=\frac{1}{Dp(x_i)-1}\sum d(x_i,x_j) \tag{6}$$

簇类平均距离 $Gavgd(x_i)$衡量的是元素密集度，数值越小，说明在目标用户 x_i 所在类中，数据点之间越紧凑。

定义 5 聚集度距离 $G(x_i)$定义为：

$$G(x_i)=\begin{cases}\min(d(x_i,x_j)),\exists x_j,\ni Dp(x_j)Dp(x_i)\\\max(d(x_i,x_j)),不\exists x_j,\ni Dp(x_j)Dp(x_i)\end{cases} \tag{7}$$

$G(x_i)$是通过比较聚集度 $Dp(x_i)$来确定的，用它来衡量不同簇之间的差异性。在所有的点中，当 x_i 的聚集度最大时，$G(x_i)$是 x_i 与剩余所有点之间的最大距离，反之则为 x_i 与剩余所有点之间的最小距离。

定义 6 中心聚集参数定义为：

$$\omega(x_i)=\frac{Dp(x_i)\cdot G(x_j)}{Gavgd(x_i)} \tag{8}$$

基于以上概念，我们给出基于中心聚集参数的改进 K-means 算法流程：

输入：所有的数据集点 $x_1,x_2,\cdots,x_n$

输出：聚类结果

(1)计算出每个点的中心聚集参数 $\omega(x_i)$；

(2)选出使得 $\omega(x_i)$ 最大的点 x_i,由它做为第一个初始聚类中心,计算出 x_i 与剩余点间的距离,将得到的距离值与邻域半径 R 作比较,若距离小于 R 则说明可以与 x_i 划为一类,因此将从数据点中除去这些点,若距离大于 R,则说明与 x_i 的距离过远,不适宜与 x_i 归为一类,因此将这些点保留下来,进行下一步;

(3)在第(2)步中保留下的点里再选出 $\omega(x_i)$ 最大的点,作为第 2 个聚类中心,再次操作步骤(2);

(4)一直重复操作步骤(3),当数据集中的点 $x_1, x_2, \cdots, x_n$ 全部去除为止;

(5)输出 k 个最优初始中心 $M_i(i=1,2,\cdots,k)$;

(6)利用初始簇心 M_i,对 $R_{m\times n}$ 执行 K-means 算法,将数据集分成 k 类;循环执行 K-means 算法,直至其准则函数收敛,得到最终聚类结果;

为验证上述算法的有效性,我们选取以下 UCI 数据集进行验证(见表 4),同时选取的评价标准有:调整后的兰德指数(*RI*)、互信息(*MI*)以及 Fowlkes-Mallows 指标[19]。实验结果如图 3 ~ 图 5:

表 4 **UCI 数据集**

数据集名称	类数	实例数	维数
Iris	3	150	4
Wine	3	178	13
Soybean	4	47	35

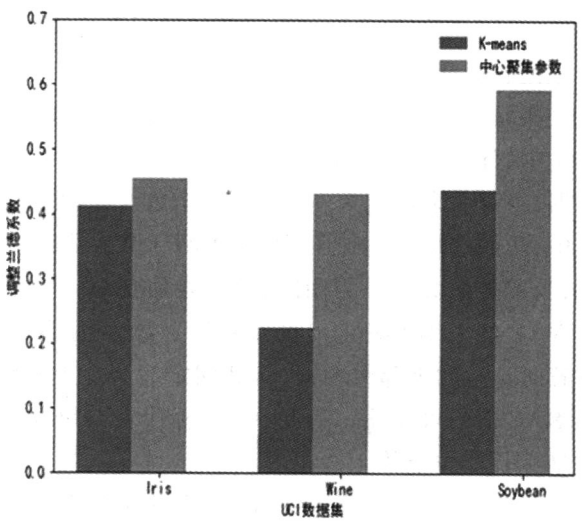

图 3 调整兰德指数指标对比

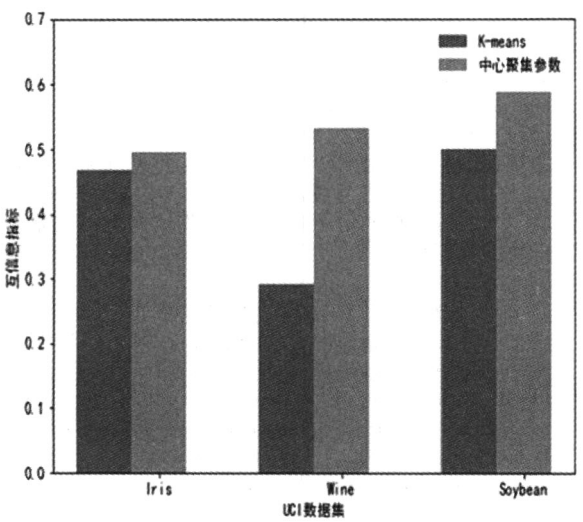

图 4 互信息指标对比

由图 3、图 4 和图 5,我们可以看出基于中心聚集参数的改进 K-means 算法得到的相关数值均比经典 K-means 聚类算法值要高,并且在 Wine 和 Soybean 两个数据集上的表现更为明显。因此通过此实验表明,基于中心聚集参数的改进 K-means 算法具有较好的聚类效果。

3.2 基于中心聚集参数的改进 K-means 协同过滤推荐算法

本节中,我们把 3.1 节中基于中心聚集参数的改进 K-means 算法用于协同过滤推荐过程中,为验证改进后的算法在推荐上的效果,为方便起见,我们从 MovieLens100k 数据中随机选取了 189 名用户的评分数据,并按 2∶8 分为测试集和训练集,设计了如下两组对比实验:

实验 1 确定最佳聚类个数。在这一阶段的实验中,需要得到两个最佳聚类个数。首先执行基于中心聚集参数的改进 K-means 算法,得出最佳的聚类个数是 19;其次需要判定经典 K-means 算法的最佳聚类个数,选取的评价指标是平均绝对误差(MAE),当聚类个数以 2 为间隔,由 2 增加到 20 时,根据 MAE 的大小来确定出最佳聚类个数,为了保证推荐时选取的近邻数相同,将其固定为 25,以此增强 MAE 值的可靠性。

图 6 表明:随着聚类个数的增加,MAE 值呈先下降后上升趋势,在聚类数 k=16 时最低。因此对经典 K-means 算法而言,最佳聚类个数是 16,从而保证获取有效的分组,提高在近邻选择时的便利性和可靠性,取得较好的推荐效果。

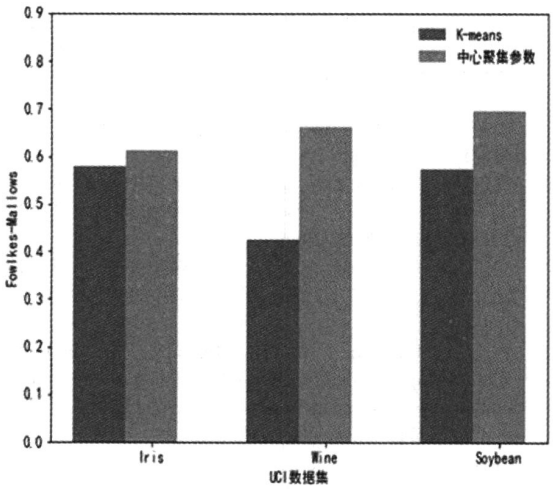

图 5 *Fowlkes-Mallows* 指标对比

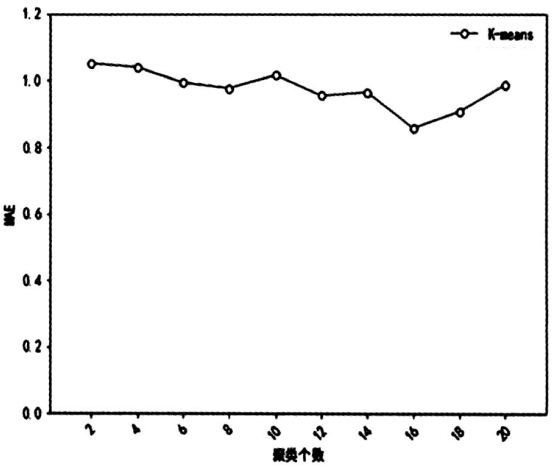

图 6 经典 K-means 不同聚类个数下的 MAE 变化

实验 2 推荐算法精度对比。我们采用三种算法来做对比试验,分别是:基于中心聚集参数的改进 K-means 协同过滤推荐算法、基于 K-means 的协同过滤算法与传统的协同过滤推荐算法。下图中纵轴是 MAE 值,横轴是近邻个数,以 5 为间隔从 5 增加到 50。

由图 7 可以看出,进行对比的三种协同过滤推荐算法它们的 MAE 值都呈现先下降后上升的趋势,整体上随着最近邻个数的增加而降低,且在同样的邻居个数变化的前提下,基于中心聚集参数改进 K-means 协同过滤推荐算法的 MAE 值最低。MAE 值越低表明推荐误差越小,所以当目标用户的近邻数不断增加时,推荐准确度也随之提高。因此,基于中心聚集参数的改进 K-means 协同过滤推荐算法的推荐效果较好。

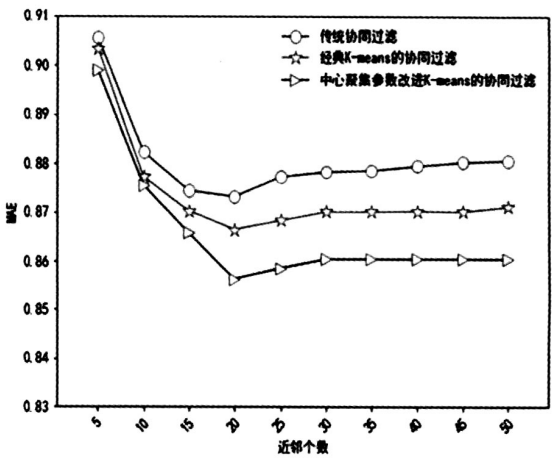

图 7 不同近邻个数下 MAE 变化

4 结论及展望

本文主要从两个角度对个性化推荐系统中的协同过滤推荐算法进行了优化。首先,基于 Slope One 算法对缺失数据进行填充,提出了基于 Slope One 的协同过滤推荐优化算法。其次,提出了一种基于中心聚集参数的改进 K-means 优化算法,并将该算法用于协同过滤推荐中。为验证本文提出算法的有效性,结合 MovieLens 数据集设计了相应的对比实验,结果表明:本文所提方法推荐精度均得到显著提高,数据稀疏性和扩展性问题得到了一定程度的改善。但本文仍有一定不足之处,比如第 3 节中因用 Slope One 填充后的矩阵过于庞大,未能将 Slope One 填充与基于中心聚集参数的协同过滤推荐算法相结合,这将是本文后续工作中需要解决的重要问题。

参考文献:

[1]Goldberg D,Nichols D,Oki B M,Terry D. Using collaborative filtering to weave an informationtapestry[J]. Communications of the

ACM,1992,35(12):61-70.

[2]Konstan J A,Miller B N,Maltz D,Herlocker J L,Gordon L R,Riedl J. GroupLens:applying collaborative filtering to usenet news [J]. Communications of the ACM,1997,40(3):77-87.

[3]李改,李磊. 基于矩阵分解的协同过滤算法[J]. 计算机工程与应用,2011,47(30):4-7.

[4]林栢全,肖菁. 基于矩阵分解与随机森林的多准则推荐算法[J]. 华南师范大学学报(自然科学版),2019,51(02):117-122.

[5]Nilashi Mehrbakhsh,Ibrahim Othman Bin,Ithnin Norafida. Multi-criteria collaborative filtering with high accuracy using higher order singular value decomposition and neuro fuzzy system[J]. Knowledge-Based Systems,2014,60(Apr.):82-101.

[6]李征,段垒. 基于用户兴趣评分填充的改进混合推荐方法[J]. 工程科学与技术,2019,51(01):189-196.

[7]袁卫华,王红,杜向华. 结合非负矩阵填充及子集划分的协同推荐算法[J]. 小型微型计算机系统,2017,38(12):2645-2651.

[8]赵文涛,任行学. 融合标签信息和时间效应的矩阵分解推荐算法[J]. 信息与控制,2020,49(04):472-477.

[9]Zhang JiaDong,ChowChiYin,XuJin. Enabling kernel-based attribute aware matrix factorization for rating prediction[J]. IEEE Transactions on Knowledge and Data Engineering,2017,29(4):798-812.

[10]Lu Qibei,GuoFeipeng. Personalized informationrecommendation model based on context contribution and item correlation[J]. Measurement,2019,142:30-39.

[11]Sun Nannan,Zhuang Yayun,Ni Shuqin. A trust-based collaborative filtering algorithm using a user preference clustering[J]. Management Science and Engineering,2017,11(3):9-19.

[12]赖锴,王雪,杜柏翰. 基于时间和聚类的协同过滤推荐策略研究[J]. 数学的实践与认识,2020,50(23):41-48.

[13]许智宏,田雨,闫文杰,暴利花. 基于模糊聚类和改进混合蛙跳的协同过滤推荐[J]. 计算机应用研究,2018,35(10):2908-2911.

[14]张文,崔杨波,李健,陈进东. 基于聚类矩阵近似的协同过滤推荐研究[J]. 运筹与管理,2020,29(04):171-178.

[15]Faris A,Chandan K Reddy,HuJunling,Hatim F A. Biclustering neighborhoodbased collaborative filtering method for top-nrecommender systems[J]. Knowledge and Information Systems,2015,44(2):475-491.

[16]Lemire D,Maclachlan A. Slope one predictors for online rating-based collaborative filtering[C]. Proceedings of the SIAM Data Mining Conference,2005:471-780.

[17]赵伟,林楠,韩英,张洪涛. 一种改进的 K-means 聚类的协同过滤算法[J]. 安徽大学学报(自然科学版),2016,40(02):32-36.

[18]项亮. 推荐系统实践[M]. 北京:人民邮电出版社,2012.

[19]王巧玲,乔非,蒋友好. 基于聚合距离参数的改进 K-means 算法[J]. 计算机应用,2019,39(09):2586-2590.

Research on Collaberative Filtering Recommendation Algorithm Optimization in Personalized Recommendation

Guan Fei　　Zhou Yi　　Zhang Han

Abstract:Collaborative filtering recommendation algorithm is widely used in personalized recommendation system. However,it has some shortcomings in data sparseness and scalability. This paper proposes a novel approach to solve the problems,which contains two components. The first component is to solve the problem of data sparseness by filling the initial scoring matrix with missing values based on Slope One algorithm and using the collaborative filtering algorithm based on K-means clustering to predict the score of target users. The second one is to solve the scalability problem by proposing an improvedimproveu K-means algorithm based on central aggregation parameters. The results of the relevant comparative experiments on Movielens dataset show that the recommended precision is significantly improved and the data sparseness and scalability issues are effectively improved. Therefore,the research conclusions of this paper can not only further enrich the existing theoretical achievements of collaborative filtering recommendation algorithm,but also provide theoretical basis and decision-making reference for improving the accuracy of the recommendation system.

Key words:recommendation system;decision making;collaborative filtering;central aggregation parameter;K-means clustering