协同过滤的相似度融合改进算法①

于世彩, 谢颖华, 王 巧

(东华大学 信息科学与技术学院, 上海 201620)

摘 要: 针对传统协同过滤推荐在数据稀疏性条件下性能不佳的问题, 在相似度计算上做出了优化, 提出了一种基于项目类别和用户兴趣相似度融合的协同过滤算法, 算法将相似度的计算分解为两个方面进行: 用户-项目类别评分相似度和用户-项目类别兴趣相似度, 将两者用合适的权值加以融合得到最终相似度, 参与最终预测评分的计算. 利用 MovieLens 公用数据集对改进前后的算法进行对比. 结果表明, 基于项目类别和用户兴趣的协同过滤改进算法有效地缓解了数据稀疏性问题的影响, 提高了推荐的准确性.

关键词: 协同过滤; 数据稀疏性; 项目类别; 用户兴趣; 相似度融合

Improved Collaborative Filtering Algorithm of Similarity Integration

YU Shi-Cai, XIE Ying-Hua, WANG Qiao

(School of Information Science and Technology, Donghua University, Shanghai 201620, China)

Abstract: Aiming at the poor recommendation quality due to the data sparsity problem of traditional collaborative filtering recommendation, this paper puts forward an improved collaborative filtering algorithm. The improved algorithm proposes a collaborative filtering algorithm based on the similarity integration of item categories and user interests to make optimization on the similarity calculation. The algorithm does not simply concentrate on similarity calculation, but divides it into two aspects: users-item category interest similarity and users-item category rating similarity, which will finally be integrated with appropriate weights to get the final similarity. After a series of verification and comparison carried out on the MovieLens public data set, it is concluded that the improved algorithm based on data sparsity of collaborative filtering indeed plays a positive role in reducing the influence caused by data sparsity and improves the accuracy of recommendation.

Key words: collaborative filtering; data sparsity; item category; user interest; similarity integration

互联网的发展带来了便利的同时, 也造成了信息量的急速增长和膨胀, 出现了“信息过载”[1]现象. 互联网信息过于丰富甚至超出了浏览者能够有效利用的范围, 导致了信息的利用率低下, 人们必须耗费大量的时间和精力才能找到需要的信息. 为缓解这一个问题, 多种方法被提了出来, 信息检索在其中扮演着重要的角色, 然而“一视同仁”的特性致使它忽略了用户特性, 其输出结果仅与输入的关键字有关, 对用户来说, 仍需要耗费大量时间筛选出所需信息, 信息过载问题并没有得到有效的解决. 个性化推荐系统能够在信息过载的大环境下, 帮助用户快速从海量数据中找到所需信息.

推荐过程依赖于 3 个要素: 推荐候选对象、用户和推荐方法[2]. 推荐系统使用户摆脱了单向的搜索服务, 实现了用户和系统的双向沟通, 在电子商务中扮演着重要的角色. 为了不断提高推荐效果的精确性与有效性, 提升推荐系统的整体性能, 不同的推荐技术陆续被提了出来, 其中包括基于用户统计信息的推荐、基于内容的推荐、协同过滤推荐以及基于混合模型的推荐等[3].

协同过滤推荐技术是目前最广为应用也是效果最理想的推荐技术[4], 它通过用户-项目评分矩阵来计算并确定目标用户的最近邻居用户集合, 根据集合中各

① 收稿时间:2016-04-20;收到修改稿时间:2016-06-01 [doi:10.15888/j.cnki.csa.005551]

邻居对各项目的评分得到目标用户的预测评分, 最终将评分最高的项目推荐给用户. 协同过滤推荐技术不需要提供领域的知识, 并且会随着时间的推移, 用户对项目评分的完善, 推荐的质量和准确度也会大幅提升; 另一方面, 由于协同过滤推荐对用户评分的依赖性很强, 网络结构、用户和项目数量的急剧增长使得数据稀疏等问题渐渐暴露了出来.

根据有关资料的统计[5], 大型电子商务系统中, 所有用户购买过并给出评分的商品数只占到系统中商品总量的 1%~2%左右. 面对这种数据极端稀疏的情况, 在为用户找到邻居集合的方面, 协同过滤算法就变得力不从心, 这也就进一步导致了推荐质量的大幅降低, 这一问题也成为了制约协同过滤应用与发展的最主要问题.

1 数据稀疏性问题[6]

由于网站结构的日渐复杂, 用户和项目数量的急剧增长, 使评价过的项目数占系统中项目总数的比例越来越小, 导致相似度计算时没有足够的输入数据, 从而得不到准确的相似度, 进而降低了系统的推荐质量. 假设一个稀疏的评分矩阵如表 1.

表 1 稀疏的评分矩阵

	i_1	i_2	i_3	i_4
u_1	3			5
u_2		3	4	
u_3	2		3	5

由该表可以看出, 用户 u_1 和 u_2 没有共同评分的项目, 因此由传统的协同过滤算法计算的二者相似度为 0, 但是用户 u_1 与 u_2、u_2 和 u_3 之间的相似度都是不为 0 的, 换句话说就是 u_3 同时与 u_1、u_2 相似, 由相似的传递性可知用户 u_1、u_2 之间并不是完全不相关的, 这说明数据稀疏性影响了用户邻居的确定, 不仅降低了推荐质量, 也严重影响了推荐精度.

不少研究者从很早就意识到了数据稀疏性的对于推荐质量的制约, 研究并提出了多种方法来缓解数据稀疏性造成的影响. 其中应用较广泛的有矩阵填充技术、矩阵降维技术和基于聚类的方法[7].

1) 矩阵填充技术

所谓矩阵填充技术, 就是将用户-项目评分矩阵中没有评分的项目用特定的数值替换掉, 以此直接的降低数据的稀疏性, 提高推荐质量和精度. 在这种情况下, 阈值的选取就显得尤为重要, 一般来说, 这个值常取评分区间的中间值或者是所有评分的平均值. 这就产生了一个问题, 虽说这种方法改善了数据的稀疏性问题, 但是他忽略了用户的个性和评分习惯的差异, 并没能使数据稀疏问题从根本上得到解决.

2) 矩阵降维技术

矩阵降维技术, 就是降低用户-项目评分矩阵的维数, 将系统中未被评分的项目或者是未评过分的用户删掉就是最简单也是最直接矩阵降维方法, 但是, 这种方法应用起来会导致没有评过分的用户就不会接收到系统的推荐, 没有被评过分的项目也不会被推荐给用户.

3) 基于聚类的方法

聚类方法中, 系统用特定的标准将各个项目集合划分到若干个聚类中, 相同聚类是具有相似属性的若干个不同项目的集合, 不同的聚类中的各个项目则具有不同的属性. 通常采用的聚类方法[8]主要有: K-Means 聚类、基于网格的聚类、基于密度的聚类和 PAM 算法等.

以上几种方法的主要思想是对矩阵进行降维或者填充, 然而单纯的填充忽略了用户个性, 而降维技术又不可避免的导致了某些信息的丢失, 都不能较好的解决数据稀疏性问题. 因此需要寻求一种方法, 不改变评分矩阵稀疏性程度, 却能达到有效提高推荐算法精度的目的, 于是本文提出了基于项目类别和用户兴趣相似度融合的协同过滤算法.

2 协同过滤的相似度融合改进算法

考虑到数据稀疏性的影响, 本文对相似度的计算做出优化, 将传统的计算过程一分为二, 分别在项目类别评分矩阵和量化的用户兴趣矩阵上计算用户相似度并融合, 引入两个调节项来进行相似度的修正.

2.1 传统用户相似度的计算

用户相似度的计算是推荐算法中最核心的部分, 传统计算的相似度的方法主要有三种: 余弦相似度、修正的余弦相似度和相关相似度[9].

用户评分数据用矩阵 R 表示, 其中, m 表示用户个数, r_{ij} 表示用户 u_i 对项目 j 的评分, 于是用户相似度可以简单地利用余弦相似度计算得到.

在这种情况下, 要求相似性的两个项目被看作m维用户空间的两个向量. 向量$\vec{u}$ 代表用户 u 对所有项

目的评分所构成的向量, 向量$\vec{v}$代表用户 v 对所有项目的评分所构成的向量, 用户 u 和用户 v 之间的相似度为:

$$sim(u,v)=\cos(\vec{u},\vec{v})=\frac{\vec{u}\bullet\vec{v}}{\|\vec{u}\|\|\vec{v}\|} \tag{1}$$

本文采用的是调整余弦相似度, 并在基本的计算方法上加以改进. 基本的余弦方式在计算相似度的方面有很多的不足和缺陷, 不同用户会有不同的评分习惯, 因而它们的评分范围可能存在较大差异, 此方法却没有考虑到这点. 调整余弦相似度通过从每个评分中减去该用户对所有项目评分的平均分值, 从而只考虑评分的偏差值, 这种方法计算的用户 u 和 v 之间的相似度为:

$$sim(u,v)=\frac{\sum_{a\in I}(r_{u,a}-\overline{r}_u)(r_{v,a}-\overline{r}_v)}{\sqrt{\sum_{a\in I}(r_{u,a}-\overline{r}_u)^2}\sqrt{\sum_{a\in I}(r_{v,a}-\overline{r}_v)^2}} \tag{2}$$

2.2 相似度计算的改进算法

2.2.1 用户-项目类别兴趣相似度

用户-项目类别兴趣相似度描述了不同用户之间的兴趣与关注点的相似性. 这一指标可以采用用户对该项目类别中所有项目的评价次数之和来表示, 这个值越高, 就表明用户对这个项目类别中的项目兴趣度越高, 它的值是一个有限的整数.

用户兴趣描述的是用户对某个项目类别的总体感兴趣程度. 由上文的分析可知, 单纯的通过降低维度的方式不能很好地解决数据稀疏性问题, 还需要一种方法与之结合, 弥补其精度上的不足. 建立一个兴趣矩阵 $T_{m\times k}$ 用来表现用户对各个类别的感兴趣程度.

$$T_{m\times k}=\begin{pmatrix}t_{11} & \cdots & t_{1k}\\ \vdots & \ddots & \vdots\\ t_{m1} & \cdots & t_{mk}\end{pmatrix} \tag{3}$$

其中, m 代表用户总数, k 代表项目类别总数, 元素 t_{ij} 代表第 i 个用户给第 j 个项目类别所包含的所有项目评价次数的和, 这个值可以用来描述用户对这个项目类别感兴趣的程度.

此外, 为了提高兴趣相似度的准确程度, 本文还考虑到了用户年龄可能产生的影响, 就日常经验来讲, 年龄越接近的用户拥有相同兴趣的可能性越大, 而年龄相差越大的用户之间很可能有着截然不同的兴趣与关注点. 就音乐来讲, 青年人更喜欢摇滚或流行歌曲, 中年人多喜欢抒情类的歌曲, 而老年人更喜欢年代性历史性比较强的歌曲. 因此, 除了要考虑到用户历史评分次数之外, 本文加入了年龄调节项以达到更高的相似精度. 由此可得到用户 u、v 之间的项目类别兴趣相似度 $sim_I(u,v)$ 计算公式如下:

$$sim_I(u,v)=\frac{\sum_{c\in C_{uv}}(t_{u,c}-\overline{t_u})(t_{v,c}-\overline{t_v})}{\sqrt{\sum_{c\in C_{uv}}(t_{u,c}-\overline{t_u})^2}\sqrt{\sum_{c\in C_{uv}}(t_{v,c}-\overline{t_v})^2}}\bullet \frac{\omega}{\omega+\left|u_{age}-v_{age}\right|} \tag{4}$$

其中, C_{uv} 是一个集合, 它由用户 u、v 评过分的所有项目类别构成; $t_{u,c}$ 代表用户 u 对项目类别 c 所包含的项目的累计评价次数; t_u 表示用户 u 对所有项目类别的评价次数的平均值; u_{age} 表示用户 u 的年龄, 另外设置了一个年龄调节参数 ω 来调节相似度计算的精度, 最佳的参数值可以在试验中进行对比选取和验证.

2.2.2 用户-项目类别评分相似度

用户-项目类别评分相似度通过计算用户的评分倾向和偏好来确定他们之间的相似程度, 即若用户对相同项目类别的评分越接近, 就对应有更高的相似度. 考虑到用户打分习惯与范围的差别, 将对项目类别的评分用这个用户对此项目类别中所有项目评分的平均值来表示, 它的取值范围与评分范围一致, 通常在区间[0, 5]中.

所有的商品都带有自身的属性, 可以根据其本身的属性把它们归入多个不同的类别中, 这样, 同一个类别当然也会包含多个不同的商品, 即项目与项目类别是多对多的关系. 因此, 项目类别是一个集合, 其中包含了多个具有某个相同属性的项目. 项目类别这个概念的应用, 其实是通过聚类的方式降低了用户-项目评分矩阵的维度, 从而在一定程度上达到了缓解数据稀疏性的目的. 用户与项目类别的关系由用户-项目类别评分矩阵 $P_{m\times k}$ 来进行描述.

$$P_{m\times k}=\begin{pmatrix}p_{11} & \cdots & p_{1k}\\ \vdots & \ddots & \vdots\\ p_{m1} & \cdots & p_{mk}\end{pmatrix} \tag{5}$$

其中, m 代表用户总数, k 代表项目类别总数, 元素 p_{ij} 代表第 i 个用户给第 j 个项目类别中所包含的项目分数的平均值, 可以用这个值来衡量用户对此项目类别中包含的所有项目的满意程度.

同样地, 为了提高评分相似度的准确程度, 本文

还考虑到了用户评价的项目的一致性. 考虑一个极端的情况, 若两个用户对同一项目类别进行过评分, 且分值接近, 但是两人评价的项目完全不相交, 此时不能说两人具有较高的相似性. 因此, 除了考虑用户对项目类别的历史评分之外, 还加入了调节项 σ_{uv}^{c} 表示用户评分项目一致性来达到更高的相似度精度.

项目类别评分相似度利用改进后的皮尔森相似度来进行计算, 改进后的用户-项目类别评分相似度 $sim_R(u,v)$ 的具体计算如下:

$$sim_R(u,v)=\frac{\sum_{c\in C_{uv}}(r_{u,c}-\overline{r_u})(r_{v,c}-\overline{r_v})\bullet\sigma_{uv}^{c}}{\sqrt{\sum_{c\in C_{uv}}(r_{u,c}-\overline{r_u})^2}\sqrt{\sum_{c\in C_{uv}}(r_{v,c}-\overline{r_v})^2}} \quad (6)$$

其中, C_{uv} 是一个集合, 它由用户 u、v 评过分的所有项目类别构成; $r_{u,c}$ 由用户 u 对项目类 c 别中的所有项目的平均评分; r_u 表示用户给分的平均值. 调节项 σ_{uv}^{c} 表示在项目类别 c 中, 用户 u 与用户 v 共同评价的项目数与二人评价的项目总数的比值.

2.2.3 相似度融合

由前两节的介绍可知 $sim_R(u,v)$ 是评分相似性, 它只单纯的考虑了用户评分的相似性, 可能存在评分很相似但是兴趣差异很大的情况; 而 $sim_I(u,v)$ 则是用户间的兴趣与关注点的吻合程度, 却并没有将任何与评分有关的信息纳入考虑, 而这一指标对相似度的计算来说是至关重要的. 因此, 要提高相似度精度, 就要多个方面综合考虑, 引入一个权重参数 α 将两种相似度进行融合, 得到最终的用户相似度.

$$sim(u,v)=\alpha sim_I(u,v)+(1-\alpha)sim_R(u,v) \quad (7)$$

3 实验结果与分析

3.1 实验数据集

本文的实验数据集采用由明尼苏达州大学在 GroupLens 研究项目中收集的 MovieLens 公用数据集[10], 它是一个基于网页的推荐研究系统, 提供了用户信息表、电影信息表和评分信息表三张表. 这个数据集包含了 943 个独立的用户信息. 这些用户共曾标记过 1682 部电影, 为数据库中的电影的评分更是超过了 10 万. 特别的, 只考虑为 20 部以上的电影评过分的用户, 并将数据库分为 70%的训练集和 30%的测试集, 然后将数据集转换成一个有 943 行(用户)和 1682 列(用户中至少有一人评过分的电影)构成的用户-电影矩阵.

3.2 实验评价标准

推荐系统研究者们用许多不同的方式评价推荐或者是预测是否成功, 本文采用了一个普遍应用的统计学的准确性度量, 叫做平均绝对误差(MAE)[11]. 这种方法就是衡量推荐与真实的用户赋给值的偏差. 对于每一对评分预测数据 $\langle p_i,q_i\rangle$ 的具体误差也就是 $|p_i-q_i|$ 进行处理. 平均绝对误差的计算方法是先计算 N 对评分-预测数据对的误差之和, 然后计算平均值, 如下式:

$$MAE=\frac{\sum_{i=1}^{N}|p_i-q_i|}{N} \quad (8)$$

一般来说, 平均绝对误差越小, 推荐结果越准确, 系统性能就越好.

3.3 确定未知参数的值

3.3.1 年龄调节参数 ω 的确定

在实验过程中, 以 1: 4 的比例随机地将数据集分成两组不同的测试集和训练集, 分别用 D1、D2 表示, 然后分别在 D1、D2 上进行仿真实验. 在算法执行过程中, 将形成的用户最近邻居集的大小(K)分别设为 10、20 和 30, 进行对比试验. 推荐质量的高低用平均绝对误差 MAE 的大小来描述. 得到的实验结果分别如图 1、图 2.

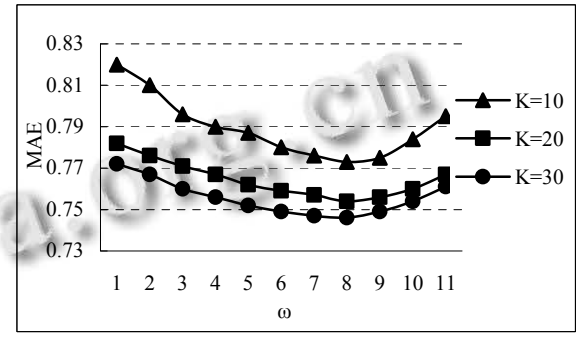

图 1　数据集 D1 上的实验结果

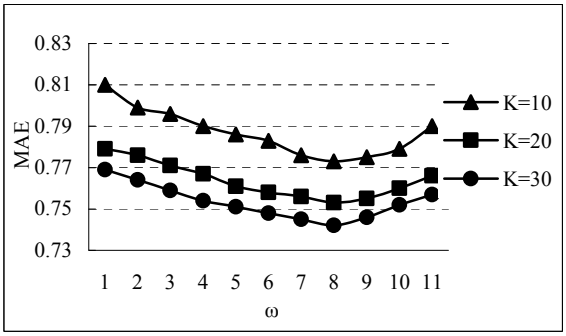

图 2　数据集 D2 上的实验结果

由以上的实验结果不难发现: 平均绝对误差值的大小, 也就是推荐系统推荐质量的高低跟用户的最近邻居集合大小有关, 邻居数量越大, 推荐越精确.

年龄调节参数 ω 的改变也能影响推荐系统的推荐精度. 在 ω 的值从 1 变化至 20 的过程中, MAE 的值随之先减小后增大, 在三个数据集上都呈现出相同的变化趋势, 并且在 ω 值取 8 的时候得到最小的 MAE 值, 也就是说, 当年龄调节参数 ω 的值取 8 时, 推荐系统能够达到最高的精度. 综上所述, 可以将年龄调节参数 ω 的值确定为 8.

3.3.2 相似度融合参数 α 的确定

在对用户项目评分相似度和兴趣相似度进行融合得到最终相似度的过程中, 参数 α 的值是不确定的, 在区间[0, 1]之间取得, 可以将 α 作为变量, 分别在数据集 D1、D2 上研究 MAE 随着 α 值变化而变化的趋势, 从而得到最佳的融合参数值. 此次将邻居集数目(K)分别设置为 10、20 和 30, 进行对比试验, 实验结果分别如图 3、图 4.

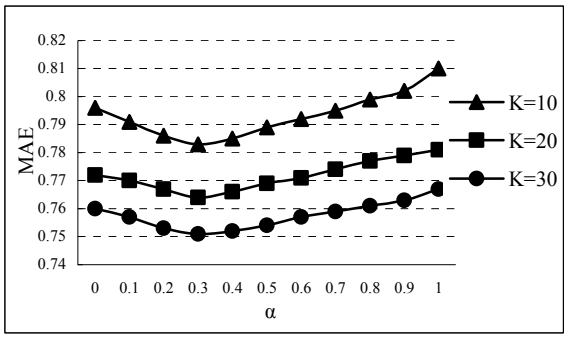

图 3 数据集 D1 上的实验结果

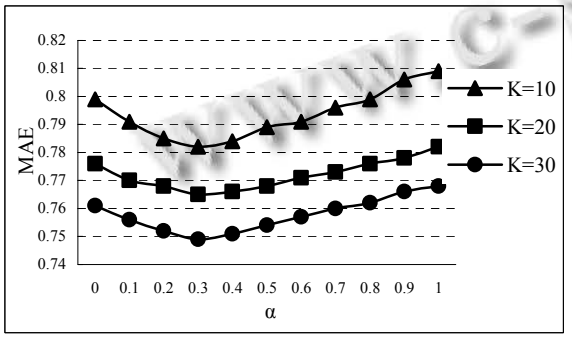

图 4 数据集 D2 上的实验结果

由以上的实验结果可得: 首先, 平均绝对误差值得大小跟用户的最近邻居集合中邻居的数量有关, 邻居集数量越大, MAE 越小, 推荐越精确.

相似度融合参数 α 的改变也能影响推荐系统的推荐精度. 在 α 的值从 0 变化至 1 的过程中, MAE 的值随之先减小后增大, 在三个数据集上都呈现出相同的变化趋势, 并且在 α 值取 0.3 的时候得到最小的 MAE 值, 也就是说, 当参数 α 的值取 0.3 时, 推荐系统能够达到最高的推荐精度. 由此可以取相似度融合参数 α 的值为 0.3. 也就是说, 在总的用户相似度中, 用户项目类别兴趣相似度所占的比重为 0.3, 对应的评分相似度所占的比重为 0.7.

3.4 算法有效性验证

在年龄调节参数 ω 的值为 8, 相似度融合参数 α 的值为 0.3 的情况下, 将传统的协同过滤算法与本文提出的基于数据稀疏性的改进协同过滤算法分别在数据集 D1、D2 上进行仿真实验, 比较两种算法性能随着邻居集数量(K)的增大的变化趋势, 验证改进算法的有效性. 实验结果如图 5、图 6.

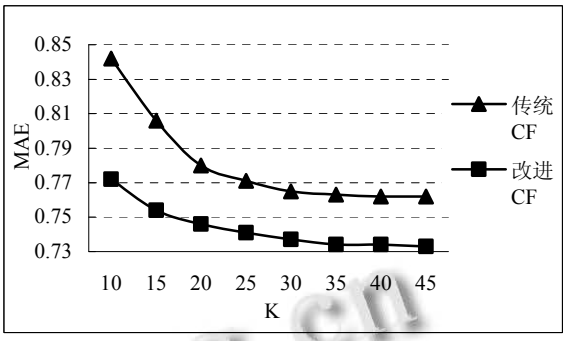

图 5 数据集 D1 上的实验结果

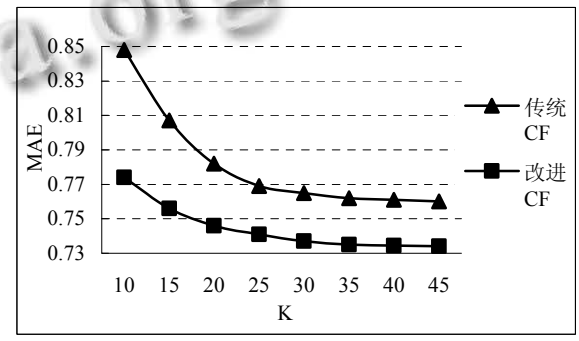

图 6 数据集 D2 上的实验结果

由以上实验结果不难发现, 推荐系统的推荐质量随着邻居集的不断增大, 推荐质量也不断提高, 最终趋于一个稳定值, 由此结果也可以帮助在应用推荐系统时选择合适的邻居集大小. 通过实验可以看出改进后的协同过滤算法对推荐系统的推荐质量是有明显的

提升的, 也就是说, 本文提出的基于数据稀疏性的协同过滤算法能够在一定程度上达到缓解数据稀疏性问题的目的.

4 结语

本文重点针对协同过滤中的数据稀疏现象, 引入项目类别的概念, 进行数据压缩, 将传统的相似度转化为融合相似度来计算, 有效地缓解了数据稀疏性的不利影响. 本文从用户评分和兴趣两方面分别计算相似性, 引入年龄调节因子和用户评分一致性因子两个调节项来对用户相似度进行进一步的修正, 最后选取合适权值进行相似度融合得到最终用户相似度.

利用 MovieLens 标准数据集进行实验仿真, 对结果进行比较分析, 从而确定了改进算法中参数 ω 和 α 的取值, 验证了改进算法的有效性. 本文提出的改进算法在同样的数据环境中能够选出相似度更高的用户, 为用户提供更加理想的推荐, 提高了推荐质量.

参考文献

1 刘鲁,任晓丽.推荐系统研究进展及展望.信息系统学报, 2008,4(1):82–90.

2 李珊.个性化推荐系统研究综述.科技致富向导,2014(11): 157–157.

3 邓晓亮.基于数据稀疏性的协同过滤推荐算法研究[硕士学位论文].重庆:重庆邮电大学,2013.

4 丁卯.基于协同过滤的推荐系统研究[硕士学位论文].天津: 河北工业大学,2013.

5 Hu JM. Application and research of collaborative filtering in e-commerce recommendation system. International Conference on Computer Science and Information Technology. 2010, 4. 686–689.

6 郭少聃.数据稀疏和隐性反馈条件下用户偏好挖掘方法[硕士学位论文].武汉:华中科技大学,2012.

7 Xia WW, He L, Chen MH, Ren L, Gu JZ. A new collaborative filtering approach utilizing item's popularity. IEEE International Conference on Industrial Engineering and Engineering Management. 2009. 1480–1484.

8 王骏,王士同,邓赵红.聚类分析研究中的若干问题.控制与决策,2012,27(3):321–328.

9 邓爱林.电子商务推荐系统关键技术研究[博士学位论文]. 上海:复旦大学,2003.

10 Zhao K, Lu PY. Improved collaborative filtering approach based on user similarity combination. International Conference on Management Science & Engineering. 2014. 238–243.

11 Zhang L, Qin T, Teng PQ. An improved collaborative filtering algorithm based on user interest. Journal of Software, 2014, 9(4).