分步筛选邻居的协同过滤改进算法①

朱毅萌, 谢颖华

(东华大学 信息科学与技术学院, 上海 201620)

摘 要: 为了解决协同过滤算法用户邻居筛选的优化问题, 提高推荐结果的准确性, 提出了一种分步筛选邻居的协同过滤改进算法. 该算法首先采用改进的 Pearson 系数法计算用户间的相似度, 降序排列后, 计算用户特征值, 大于用户特征阈值的用户进入下一层筛选; 然后选择对优先项目集有过评分的用户形成最终的邻居集; 最后进行预测评分得到推荐. 实验结果表明, 该算法能够有效地获取用户最近邻居集, 改善准确性, 并且稳定性良好.

关键词: 邻居筛选; 用户特征; 优先项目集; 评分邻居优先

Collaborative Filtering Recommendation Algorithm with Step Screening Neighbors

ZHU Yi-Meng, XIE Ying-Hua

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

Abstract: To increase the accuracy of the neighbor screening in collaborative filtering algorithm, an improved system—collaborative filtering with step screening neighbors (SSN-CF)—is proposed in this paper. This algorithm firstly uses an improved Pearson method to compare the similarity between users. After arranging the data in descending order, the uses' characteristic value is calculated. Only those who surpass the threshold value are selected. Then the system gathers the users who graded the priority set to make up the final neighbor set. Finally the users' grades are estimated and recommendation is made. Experiments have shown that the algorithm can effectively get the most similar neighbor set of target uses. Meanwhile, it is tested that accuracy and stability is improved.

Key words: collaborative filtering; neighbor screening; users' characteristic; prefer set; rating neighbors' priority

随着互联网不断发展、信息技术的广泛应用, 信息资源正以指数级速度迅猛增长, 应接不暇的信息带来了信息过载(Information Overload)[1]的问题. 搜索引擎技术、专业数据索引的出现在一定程度上缓解了问题的扩大, 但是这些工具是根据用户主动输入的信息来寻找与其匹配的信息, 如果输入内容相同, 则返回的信息也基本相同, 只满足了主流的需求, 搜索缺乏个性化. 考虑到此局限性, 人们在不断地探索并发展了更具个性化特色的推荐系统(Recommendation System)[2], 推荐系统根据用户的兴趣偏好特点和曾经的购买行为, 推荐用户可能感兴趣的信息或商品, 常用的推荐算法有基于内容的推荐[3]、协同过滤推荐[4]、基于关联规则的推荐[5]等. 推荐算法各有其利弊, 基于内容的推荐算法简洁明了, 推荐响应时间短, 对文本型资源过滤效率很高, 但无法自动处理非文本型资源, 且缺乏实时性, 无法快速适应用户兴趣变化; 基于关联规则的推荐不需要用户主动输入, 实现相对简洁, 并且利用实际交易数据作为数据源, 符合"数据源的通用性"要求, 但发现关联规则耗时, 且存在项目同名同义问题. 然而, 协同过滤算法(Collaborative Filtering, CF) 凭借其可跨类型推荐、可推荐新信息等优点, 成为最常用、应用最广泛的的推荐技术, 它根据用户的兴趣特点和购买行为, 向用户推荐其感兴趣的信息. 协同过滤存在准确度低、数据稀疏性、可扩展性受限、第一评价[6]等问题. 本文针对准确性的提高, 从筛选邻居的角度出发, 增加筛选步骤, 通过三层对邻居的

① 收稿时间:2014-10-22;收到修改稿时间:2014-12-17

筛选, 每一层均在上一层的基础上引入新的筛选标准, 查找邻居时, 一方面考虑到用户特征属性的影响, 过滤一些特征差距较大的用户; 另一方面, 优选出对目标项目和其相似项目评过分的用户, 提高邻居用户的质量, 以此提高整个算法的准确性.

1 协同过滤算法

协同过滤的基本原理是: 根据用户群体的历史行为, 寻找与目标用户兴趣偏好相似的用户作为其邻居, 然后根据邻居的历史偏好, 对这些偏好作出预测评价, 生成目标用户的推荐列表. 协同过滤主要可分为基于内存和基于模型[7]两类, 基于模型推荐的原理是采用训练集离线学习一个预测模型, 再利用这个模型进行在线评价预测产生推荐. 现阶段, 基于内存推荐方法是较常用的一种, 它通过直接计算相似度, 然后根据相似度的高低来寻找邻居, 以此产生推荐, 其基本步骤为: 建立用户与项目的矩阵→相似度计算→项目评价预测→产生推荐列表.

1.1 协同基于内存的协同过滤推荐实现过程

基于内存的协同过滤推荐实现过程主要可分为四步[8]:

步骤 1. 建立用户-项目评分矩阵

进行数据预处理, 生成一个可直观表示用户评分概况的数据集矩阵, 称为用户-项目评分矩阵, 设 Rm*n 为一个 m*n 维的矩阵, 如表 1 所示, m 为用户数, 用户集 $U=\{User_1,User_2,\ldots,User_m\}$, n 为项目数, 项目集 $I=\{Item_1,Item_2,\ldots,Item_n\}$, r_{ij} 为用户 i 对项目 j 的评分.

表 1 用户-项目评分矩阵 Rm*n

	项目					
用户	$Item_1$	$Item_2$	$Item_3$ …	$Item_j$ …	$Item_{n-1}$	$Item_n$
$User_1$	r_{11}	r_{12}	r_{13}	r_{1j}	$r_{1(n-1)}$	r_{1n}
$User_2$	r_{21}	r_{22}	r_{23}	r_{2j}	$r_{2(n-1)}$	$r_{2n}\cdots$
$User_i$	r_{i1}	r_{i2}	r_{i3}	r_{ij}	$r_{i(n-1)}$	$r_{in}\cdots$
$User_{m-1}$	$r_{(m-1)1}$	$r_{(m-1)2}$	$r_{(m-1)3}$	$r_{(m-1)j}$	$r_{(m-1)(n-1)}$	$r_{(m-1)n}$
$User_m$	r_{m1}	r_{m2}	r_{m3}	r_{mj}	$r_{m(n-1)}$	r_{mn}

步骤 2. 筛选邻居

筛选相似邻居通过相似度计算, 利用 K 邻近算法[9]找到相似度最高的 K 个用户作为相似用户. Pearson 相关系数法在相似度计算中表现出的效果更好, 其计算公式为:

$$SIMILARITY(u,v)=\frac{\sum_{i\in C_{uv}}(r_{ui}-\overline{r_u})(r_{vi}-\overline{r_v})}{\sqrt{\sum_{i\in C_{uv}}(r_{ui}-\overline{r_u})^2}\sqrt{\sum_{i\in C_{uv}}(r_{vi}-\overline{r_v})^2}} \quad (1)$$

其中, r_{ui} 为目标用户 u 对目标项目 i 的评分, r_{ui} 为用户 v 对目标项目 i 的评分. $\overline{r_u}$ 和 $\overline{r_v}$ 用户 u 和 v 在评过分项目集合上的评分均值, $C_{uv}=C_u\cap C_v$ 表示用户 u 和用户 v 共同评分项目集合.

步骤 3. 项目评分预测

预测评分由邻居对目标项目评分的加权平均得到, 并考虑不同用户对项目的评分尺度, 得到的计算公式为:

$$p_{ui}=\overline{r_u}+\varepsilon\cdot\frac{\sum_{v\in I(u)}(r_{vi}-\overline{r_v})\cdot SIMILARITY(u,v)}{\sum_{v\in I(u)}|SIMILARITY(u,v)|} \quad (2)$$

其中, P_{ui} 为目标用户 u 对目标项目 i 的预测评分, $I(u)$ 表示用户 u 的邻居集合.

步骤 4. 产生推荐列表

根据得到的预测评分产生相应的推荐列表, 比较常用的方法是 Top-N 法[10]. Top-N 法分别统计邻居集中, 目标用户对不同项目的预测评分, 降序排列后取结果中 N 个排在最前面的但不属于已评价的项目作为 Top-N 法得到的推荐列表.

1.2 传统协同过滤算法主要存在的问题

传统的协同过滤算法存在准确度问题、数据稀疏性问题、可扩展性问题、第一评价问题、兴趣更新问题等. 本文针对准确性问题, 发现实验步骤的第二步是关键, 故需要从筛选邻居角度出发, 提高推荐精度.

2 改进的协同过滤算法

本节优化筛选邻居的过程, 提出分步筛选邻居的协同过滤算法(SSN-CF), 主要对筛选邻居模块与预测评分模块作出改进.

2.1 筛选邻居模块的改进

2.1.1 第一层筛选——改进的 Pearson 相关系数法

第一层筛选采用 Pearson 相关系数法的改进方法计算用户间相似度作为海选阶段. 文献[11]提出了一种改进相似度计算法, 在一定程度上, 减小了新注册用户由于评分数量较少而带来的影响.

$$SIMILARITY'(u,v)=\frac{\sum_{i\in C_{uv}}(r_{ui}-\overline{r_u})(r_{vi}-\overline{r_v})}{\sqrt{\sum_{i\in C_{uv}}(r_{ui}-\overline{r_u})^2}\sqrt{\sum_{i\in C_{uv}}(r_{vi}-\overline{r_v})^2}}\cdot\frac{\min(|C_{uv},w|)}{w} \quad (3)$$

w 为公共评分项目阈值，C_{uv} 表示目标用户 u 与用户 v 共同评分项的集合，$|C_{uv}|$ 表示两个用户间的共同评分项目个数.

$$\text{由于}\frac{\min(|C_{uv}|,w)}{w}=\begin{cases}1 & |C_{uv}|\geq w\\ \dfrac{|C_{uv}|}{w} & |C_{uv}|<w\end{cases} \tag{4}$$

则公式 3 可等价为:

$$SIMILARITY'(u,v)=\begin{cases}\dfrac{\sum_{i\in C_{uv}}(r_{ui}-\bar{r_u})(r_{vi}-\bar{r_v})}{\sqrt{\sum_{i\in C_{uv}}(r_{ui}-\bar{r_u})^2}\sqrt{\sum_{i\in C_{uv}}(r_{vi}-\bar{r_v})^2}} & |C_{uv}|\geq w\\ \dfrac{\sum_{i\in C_{uv}}(r_{ui}-\bar{r_u})(r_{vi}-\bar{r_v})}{\sqrt{\sum_{i\in C_{uv}}(r_{ui}-\bar{r_u})^2}\sqrt{\sum_{i\in C_{uv}}(r_{vi}-\bar{r_v})^2}}\cdot\dfrac{|C_{uv}|}{w} & |C_{uv}|<w\end{cases} \tag{5}$$

根据公式(5)计算用户间相似度，并进行降序排列，选出前 D 个用户添加至待选集 D1，进入第二层筛选.

2.1.2 第二层筛选——引入用户特征阈值和用户特征值

引入用户特征阈值和用户特征值，主要考虑到用户年龄、性别和所属职业对筛选邻居的影响，滤掉一些特征并不那么相近的用户，来提高邻居的质量. 用户特征值计算公式为:

用户特征值 $Char=\alpha\cdot Age+\beta\cdot Sex+\gamma\cdot Occupation$ (6)

其中，α、β 和 γ 称为用户特征权重，$\alpha+\beta+\gamma=1$.

以 Movielens 数据集为例，对年龄、职业进行量化操作.

1) 量化年龄

表 2　年龄量化表(基于 Movielens 数据集)

年龄段	7-17	18-25	26-35	36-45	46-60	61-73
量化值	1	2	3	4	5	6

若用户所属年龄段相同，则 *Age* 取值为 1，否则取值为 0.

2) 判别性别

若与目标用户性别相同，则 *Sex* 取值为 1，否则取值为 0.

3) 量化职业

表 3　职业聚类量化表(基于 Movielens 数据集)

职业类	technician/programmer/engineer		
量化值	1		
职业类	executive/administrator	educator/librarian	
量化值	2	3	
职业类	marketing/salesman	doctor/healthcare	
量化值	4	5	
职业类	artist/ writer	scientist	entertainment
量化值	6	7	8
职业类	homemaker	lawyer	Student
量化值	9	10	11
职业类	retired	other	none
量化值	12	13	14

若用户所属职业类相同，则 *Occupation* 取值为 1，否则取值为 0.

由公式 6 计算用户特征值 *Char*，选出 *Char* 大于给定阈值 θ 的用户添加至待选集 D2，进入第三层筛选. 显然，$0\leq Char,\theta\leq 1$，阈值 θ 可以调节.

2.1.3 第三层筛选——评分邻居优先原则

从一个简单的例子可以看出在筛选相似邻居时可能遇到的问题，如用户 A 对《霍比特人》、《超人》评分高，用户 B 对《指环王》、《饥饿游戏》评分高，很明显，这两个用户对这类电影比较偏爱，应该是相似用户. 但是，由于他们没有对同一个项目评分，这就导致了在第二步时不能计算他们的相似度，造成相似用户的流失，将这个问题定义为“相似用户无共同评分集”现象，造成这个现象的根本原因是由于评分矩阵的稀疏性.

为了缓解这一问题，本文在第三层筛选前，先将项目根据类别划分类别子集，在类别子集中计算项目间的相似度，找出与目标项目 i 最相似的 p 个邻居项目，将此集合称为优先项目集 Prefer. 若项目只属于一个类别，则直接选取相似度最高的 p 个项目作为邻居项目集；若属于多个类别(设为 z 个类别)，先在 z 个类中计算项目间相似度，在 z 类中选出最高的 p/z 个，分别取整后，综合为邻居项目集，其个数为 p'=z*[p/z]. 通常 $p'<p$，随机选取一个类别按顺序选取紧邻的相似项目来补全 p 个. 其中，项目间相似度计算仍采用 Pearson 相关系数法.

优先项目集 Prefer 生成后，设立评分邻居优先选择原则，即在待选集 D2 中优先选取对 Prefer 集任何一个项目评过分的用户加入最终的用户邻居集 Neighbor. 若经过第三层筛选后，邻居个数不足起初规定的 K 个，则在待选集 D2 中选择未对 Prefer 集评分且相似度最高的用户作为补充.

2.2 预测评分模块的改进

本文引用文献[12]提出的预测评分改进公式, 定义邻居集 P(u)为与目标用户相似度较高且对优先项目集有评分记录的用户, NP(u)为与目标用户相似度最高但没有评价过优先项目集的用户, 设定优先权重 ε 调整两部分对预测评分的影响.

改写的计算公式如下:

$$p_{ui}=\bar{r}_u+\varepsilon\cdot\frac{\sum_{v\in P(u)}(r_{vi}-\bar{r}_v)\cdot SIMILARITY'(u,v)}{\sum_{v\in P(u)}|SIMILARITY'(u,v)|}+(1-\varepsilon)\cdot\frac{\sum_{v\in NP(u)}(r_{vi}-\bar{r}_v)\cdot SIMILARITY'(u,v)}{\sum_{v\in NP(u)}|SIMILARITY'(u,v)|} \tag{7}$$

其中, P_{ui} 为用户 u 对项目 i 的预测评分, NP(u)部分中 r_{vi} 为 $\bar{r}_v$ 随机加上(0,1)中小数后再取整.

3 实验结果与分析

3.1 实验数据集

实验所用的数据集为网络开源数据集 Movielens(100K), 此数据集包括 943 个用户对 1682 个电影的 100000 个评分记录, 其稀疏度为 [1-100000/(943×1682)]×100%=93.7%. 实验采用五折交叉法, 随机将数据集分为训练集 train 和测试集 test, 分别占数据集总量的 80%和 20%.

3.2 实验评价标准

本文实验中采用平均绝对误差(*MAE*)[13]作为评价指标, *MAE* 是推荐系统评价指标中应用最广泛的标准, *MAE* 的值越小表示推荐准确度越高, 其计算公式为:

$$MAE(u)=\frac{\sum_{i=1}^{n}|r_{ui}-p_{ui}|}{n} \tag{8}$$

$$MAE=\frac{\sum_{i=1}^{N}MAE(u)}{N} \tag{9}$$

*MAE(u)*为目标用户 u 对目标项目 i 的评分预测值与测试集中真实评分值偏差绝对值和的平均, 其中 n 为目标用户在测试集中已有的评分个数, 整个推荐算法的 *MAE* 为所有测试集用户 *MAE* 的平均, N 为测试集中用户的个数.

3.3 实验结果与分析

本文设计五个实验来度量各参数的变化对推荐系统产生的影响, 并在最优参数的情况下, 与传统的基于用户的协同过滤算法进行比较. 实验之前, 首先考察改进的 Pearson 相关系数的优越之处, 取 w=20, 与未改进之前作比较, 如图 1 所示, 改进后的相似度计算法的 *MAE* 值有明显的减小, 因此在以下四个实验中本文均为设定 w=20.

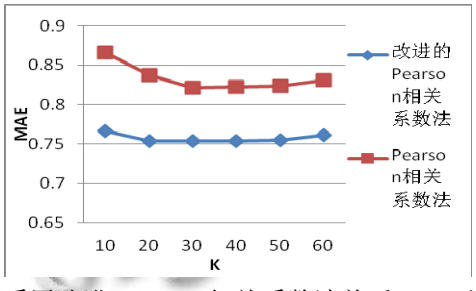

图 1 采用改进 Pearson 相关系数法前后 MAE 比较图

3.3.1 实验 1: 参数 D/K 对推荐系统性能的影响

在实验 1 中仅改变 *D/K* 值(设 ε=0.8, θ=0.2, p+1=6), 邻居数 K=20, 由图 2 可见, 不断增大 *D/K* 的值, 当 *D/K* 值较小时, *MAE* 值下降明显且迅速, 随着 *D/K* 的不断增大, *MAE* 逐渐趋于平缓, 存在微量减小. 这是由于 *D/K* 的值越大表示参与筛选邻居的用户越多, 能够筛选出有用的邻居的几率就高, 从而使得推荐系统的准确度越高. 但当 *D/K* 增大到一定程度时, 最有效邻居已经全部能够参与筛选, 并且 *D* 值超过训练集用户数, 第一层筛选起不到作用, 继续扩大 *D/K* 值则没有明显的优势, 反而会降低准确度. 因此, 是否选取了适当的 *D/K* 值对推荐结果的准确度有着很大的影响, 选取 *MAE* 稳定段[15,25]中的值为最优 *D/K* 值.

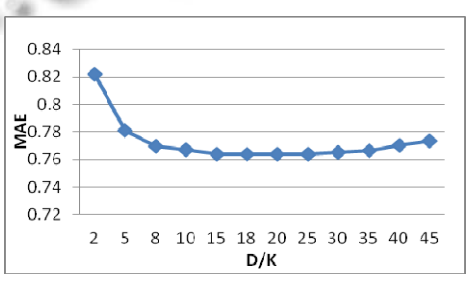

图 2 K=20 时, MAE 随 D/K 变化图

3.3.2 实验 2: 优先权重 ε 对推荐系统性能的影响

实验 2 仅改变优先权重参数 ε(设 K=50, D=100, θ=0.2, p+1=6), 由图 3 可知, 当 ε=0.7 时, 即当 Prefer 集评分用户对预测评分贡献度占 70%时, *MAE* 值最小, 推荐结果的准确度最高.

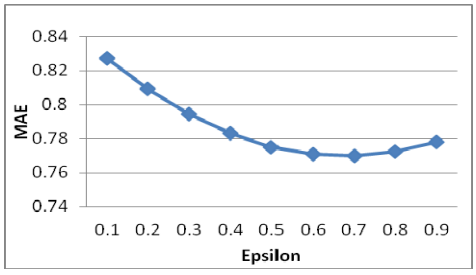

图 3　MAE 随优先权重 ε 变化图

3.3.3 实验 3: 用户特征阈值 θ 对推荐系统性能的影响

由图 4 可见 θ=0.2, 即选取三个特征满足任一个的用户通过第二层筛选时, 推荐质量最好, 而 θ 增大到 0.4 或 0.5 时, 过分提高了邻居筛选时用户特征的门槛, 让一部分相似度很高的用户淘汰, 造成推荐准确度的降低, 因此当推荐系统探测到推荐准确度降低的时候, 则需要适当地降低用户特征阈值 θ, 避免有效相似用户无法进入下一层筛选.

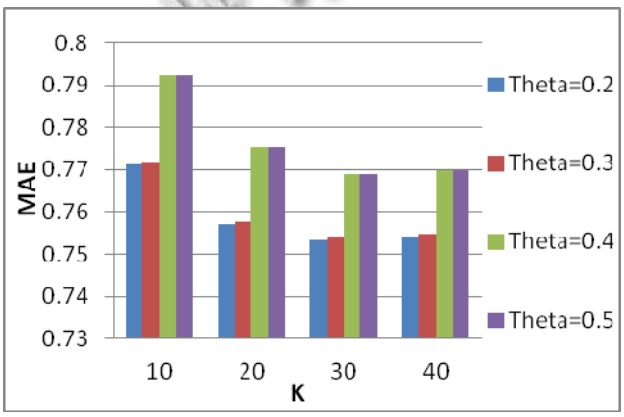

图 4　不同用户特征阈值下, MAE 随邻居数 K 的变化图

3.3.4 实验 4: 优先项目集中项目个数 p+1 对推荐系统的影响

实验四设 K=50, D=100, ε=0.7, θ=0.2, 当 p+1=2 或 3 时, 即目标项目的相似邻居取 1 或 2 个时, *MAE* 值相差不大且最优.

3.3.5 实验五: 推荐性能随邻居数 K 的变化

根据上述实验所得的结果, 选取相对较优参数: D/K=15, ε=0.7, θ=0.2, p+1=2, 与传统基于用户的协同过滤算法作出比较.

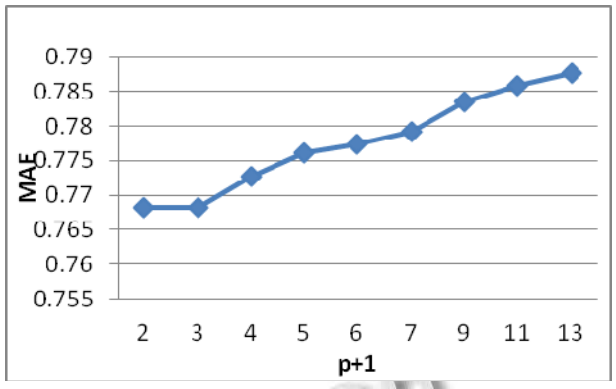

图 5　MAE 随 Prefer 集中项目个数 p+1 变化图

由图 6 可看出, 改进算法 SSN-CF 得到的 *MAE* 值存在明显地降低, 当选取邻居数 K=30 时, 系统性能最好, 总体而言, 邻居数 K 在[20,50]段中, 系统性能都较为稳定.

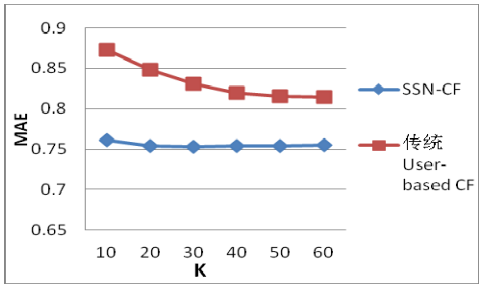

图 6　SSN-CF 改进算法与传统 User-based CF 的 MAE 比较图

原因分析: SSN-CF 算法将邻居筛选过程从一步增加至三步, 达到提高邻居质量的目的, 第一层通过考察共同评分项, 以减小了新注册用户的不利影响, 第二层考虑用户特征属性对其兴趣偏好的作用, 第三层通过在项目类别子集上寻找相似度较高的项目邻居, 能够挖掘出用户的潜在偏好项目和潜在邻居, 在一定程度上提高找到相似邻居的可能性. 因此, 通过三层筛选, 由 SSN-CF 算法预测得到的推荐项目更贴近目标用户的偏好, 推荐的准确度能够得到显著提升, 同时, 经过多层筛选后, 邻居数 K 的影响相对减弱, 故算法性能较稳定.

4　结语

本文重点分析了协同过滤算法的实现过程和其优缺点, 针对推荐质量不高的问题, 提出了分步筛选邻居的方法, 通过邻居质量的提高来提升推荐质量, 具

体的研究成果如下:

① 提出了用户特征阈值的筛选条件, 在传统协同过滤算法改进的 Pearson 相关系数法计算相似度的基础上, 在第二层查找邻居时, 加入用户的特征属性的影响, 设定用户特征阈值, 考虑邻居在固有特征方面的筛选优化.

② 提出了优先项目集的概念, 通过在项目类别子集上寻找相似度较高的项目邻居加入到优先项目集中, 避免因“相似用户无共同评分集”问题而导致邻居的流失, 在一定程度上减小了数据稀疏造成的负面影响. 同时设定评分邻居优先原则, 优选出对目标项目与潜在偏好项目评过分的近邻, 权衡此类相似用户对预测评分的影响比重.

③ 采用网络开源数据集 Movielens 对各个相关参数作出调整, 进行了相应的实验, 验证了新算法的优越性. 证明分步筛选邻居的协同过滤算法选出的邻居相似度更高, 由最终筛选出的邻居预测的项目也更贴近目标用户的偏好, 推荐质量更高. 同时, 当选取邻居数变化时, 改进算法也保持了良好的稳定性.

参考文献

1 蔺丰奇,刘益.信息过载问题研究述评.情报理论与实践,2007,30(5):710–714.

2 赵亮,胡乃静,张守志.个性化推荐算法设计.计算机研究与发展,2002,39(8):986–991.

3 Balabanovid M, Shoham Y. Content-based collaborative recommendation. Communications of the ACM, 1997, 40(3): 66–72.

4 王均波.协同过滤推荐算法及其改进研究[硕士学位论文].重庆:重庆大学,2010.

5 索琪,卢涛.基于关联规则的电子商务推荐系统研究.哈尔滨师范大学自然科学学报,2005,21(2):50–53.

6 Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowledge and Data Engineering, 2005, 17(6): 734–749.

7 傅鹤岗,彭晋.基于模范用户的改进协同过滤算法.计算机工程,2011,37(3):70–71,74.

8 张卫星.基于协同过滤技术的电子商务个性化推荐研究[硕士学位论文].重庆:重庆大学,2008.

9 Michael J, Andreas T, Robert L. Combining predictions for accurate recommender system. Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2010. 693–702.

10 程淑玉,基于协同过滤算法的个性化推荐系统的研究[硕士].合肥:合肥工业大学,2010.

11 沈健.电子商务的个性化协同过滤推荐算法研究[硕士学位论文].上海:上海交通大学,2013.

12 蔡观洋.个性化推荐中协同过滤算法的改进研究[硕士学位论文].长春:吉林大学,2013.

13 Bobadilla J, Hernando A, Ortega F, Bernal J. A framework for collaborative filtering recommender systems. Expert Systems with Applications, 2011, 38: 14609–14623.