计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2019,28(6):159−164 [doi: 10.15888/j.cnki.csa.006950]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于用户聚类与项目划分的优化推荐算法①

申晋祥, 鲍美英

(山西大同大学 计算机与网络工程学院, 大同 037009)

摘　要: 针对传统协同过滤推荐算法没有充分考虑用户属性及项目类别划分等因素对相似度计算产生的影响, 存在数据稀疏性, 从而导致推荐准确度不高的问题. 提出一种基于用户属性聚类与项目划分的协同过滤推荐算法, 算法对推荐准确度有重要影响的相似度计算进行了充分考虑. 先对用户采用聚类算法以用户身份属性聚类, 进而再对项目进行类别划分, 在相似度计算中增加类别相似度, 考虑共同评分用户数通过加权系数进行综合相似度计算, 最后结合平均相似度, 采用阈值法综合得出最近邻. 实验结果表明, 所提算法能够有效提高推荐精度, 为用户提供更准确的推荐项目.

关键词: 推荐系统; 协同过滤; 聚类算法; 相似度计算

引用格式: 申晋祥,鲍美英.基于用户聚类与项目划分的优化推荐算法.计算机系统应用,2019,28(6):159–164. http://www.c-s-a.org.cn/1003-3254/6950.html

Optimal Recommendation Algorithm Based on User Clustering and Project Partition

SHEN Jin-Xiang, BAO Mei-Ying

(College of Computer and Network Engineering, Shanxi Datong University, Datong 037009, China)

Abstract: The traditional collaborative filtering recommendation algorithm does not fully consider the impact of user attributes and item classification on similarity calculation, which results in data sparsity and low recommendation accuracy. This study proposes a collaborative filtering recommendation algorithm based on user attribute clustering and item partitioning. The algorithm fully considers the similarity calculation which has an important impact on recommendation accuracy. Firstly, users are clustered by user identity attributes using clustering algorithm, and then the items are classified. In the similarity calculation, category similarity is added. Considering the number of users scored jointly, comprehensive similarity is calculated by weighted coefficient. Finally, combined with average similarity, the nearest neighbor is synthesized by threshold method. The experimental results show that the proposed algorithm can effectively improve the recommendation accuracy and provide more accurate items for users.

Key words: recommender system; collaborative filtering; clustering algorithm; similarity calculation

引言

在信息爆炸的大数据时代, 互联网中海量数据的出现使用户想要获取自己所需要的信息变的越来越不容易[1,2]. 面对大量的数据信息, 如何有效改善“信息过载”问题[3,4], 是目前大数据研究者的主要内容之一. 比较成熟的信息过滤方法有网站导航、搜索引擎和推荐系统 (Recommender Systems)[5,6], 但是当用户不能明确表达自己的需求时, 前两种方法就略显无奈了. 推荐系统正是因此而被广泛使用的, 成为现今大数据环境下一种非常有效的信息过滤手段.

① 基金项目: 国家自然科学基金 (11871314); 山西省青年科技基金 (2015021101); 山西大同大学校级科研项目 (2017K7)
Foundation item: National Natural Science Foundation of China (11871314); Young Science and Technology Fund of Shanxi Province (2015021101); Scientific Research Fund of Shanxi Datong University (2017K7)
收稿时间: 2018-12-08; 修改时间: 2019-01-15; 采用时间: 2019-01-22; csa 在线出版时间: 2019-05-25

推荐算法是推荐系统的核心技术[7], 比较常用的有基于协同过滤推荐算法、基于内容推荐算法和混合推荐算法[8,9]. 其中协同过滤推荐算法因其可由已知用户的偏好预测其可能感兴趣的项目, 不依赖具体项目的特征信息, 对具体内容分析技术无过高要求等优点, 使其在理论研究及实践应用中有很大的发展. 但该算法在大数据环境下所显现出来的数据稀疏性、冷启动和时效性等问题[10,11], 需要对其进行有效完善.

目前国内外研究人员针对此问题提出了许多方法, 以完善算法的推荐结果. 丁少衡等人[12]提出基于用户属性和评分的协同过滤推荐算法, 从用户评分、用户兴趣变化等多个角度对相似度计算进行了改进, 但因实际推荐过程中评分数据稀少使得相似度计算仍存在问题. 杨尚君等人[13]提出基于 AntClass 算法的协同过滤推荐方法, 把用户评分定义成数据流, 采用 AntClass 算法和预处理过的数据流进行融合, 提高了推荐的精确度, 但存在计算复杂度较高, 耗时较长的问题. 王颖等人[14]提出融合用户自然最近邻推荐算法, 针对现有方法确定邻居个数困难导致推荐准确率不高问题, 通过自适应寻找自然最近邻居集, 采用融合的方法预测目标用户评分, 对推荐的准确率有所提高, 但存在计算中忽略用户和项目之间许多内在信息的问题.

基于以上研究及存在的问题, 针对传统协同过滤推荐算法没有充分考虑用户属性及项目类别划分等因素对相似度计算的影响, 提出一种基于用户属性聚类与项目划分的协同过滤推荐算法. 算法对推荐准确度有重要影响的相似度计算进行了充分考虑, 结合用户属性及项目类别划分计算相似度, 并且在项目最近邻选取时采用阈值计算, 提高了算法的准确度.

1 传统协同过滤推荐算法原理

协同过滤推荐算法对用户-项目评分矩阵的数据进行分析, 根据喜好相似的用户一般会对相同的物品有相近的喜好的原理, 为用户产生推荐. 分为基于用户的协同过滤推荐算法 (user-based) 和基于项目的协同过滤推荐算法 (item-based). 其实现过程分为三步:

(1) 构建用户-项目评分矩阵. 可由 $m\times n$ 的评分矩阵表示, m 和 n 分别表示用户和项目的值, 任一用户 i 对任一项目 j 的评分用 r_{ij} 表示. 当然, 实际的评分矩阵是极稀疏的.

(2) 最近邻的选取. 此步是协同过滤算法的核心, 通过计算项目间或用户间的相似度, 选取与目标用户最相似的最近邻集合为目标用户的最近邻. 余弦相似度及修正的余弦相似度和 Pearson 相关相似度是常用的计算方法. 余弦相似度计算如式 (1) 所示.

$$Sim(i,j)=\cos(i,j)=\frac{i\cdot j}{\|i\|\cdot\|j\|} \tag{1}$$

Pearson 相关相似度计算, 以项目之间相似度计算为例如式 (2) 所示, 用户之间相似度计算同理.

$$Sim^{b}(I_i,I_j)=\frac{\sum_{u\in U_{ij}}(r_{ui}-\overline{r_i})\cdot\left(r_{uj}-\overline{r_j}\right)}{\sqrt{\sum_{u\in U_{ij}}(r_{ui}-\overline{r_i})^2}\sqrt{\sum_{u\in U_{ij}}\left(r_{uj}-\overline{r_j}\right)^2}} \tag{2}$$

式中, U_{ij} 表示对项目 I_i 和 I_j 同时共同评过分的用户集, r_{ui} 和 r_{uj} 表示用户 u 对项目 I_i 的评分和对项目 I_j 的评分, $\bar{r}_i$和$\bar{r}_j$表示全体用户对项目 I_i 的评分平均值和对项目 I_j 的评分平均值.

(3) 产生推荐结果. 通过步骤 (2) 的计算结果, 将未评分项目中的预测评分较高的 N 个项目作为推荐结果.

综上所述, 相似度的准确计算对推荐结果有重要影响, 但传统协同过滤算法未考虑用户属性聚类及项目类别划分等因素对相似度计算的影响. 为此, 提出一种基于用户聚类与项目划分的优化推荐算法.

2 基于用户属性聚类的 User-based 协同过滤推荐算法

2.1 问题分析及改进思路

传统的 User-based 协同过滤推荐算法中因用户对项目的评分数据过少, 以至于评分矩阵过于稀疏, 使得该算法在相似度计算时精确度不高, 而且过于稀疏的用户-项目评分矩阵数据完全不能反映相似度计算的结果. 针对以上问题分析提出在计算中结合用户属性, 思路是不同的用户之间有相似的偏好或感兴趣的内容与其身份属性有很大的关联, 比如同龄人或相同职业的用户偏好或感兴趣的内容可能更相近. 具体实现首先采用 K-means 聚类算法对用户身份属性进行类别划分, 用户身份属性主要包括年龄、性别、职业、专业等等, 按照属性把用户划分到不同的类别, 然后在此聚类基础上实现协同过滤推荐算法.

2.2 改进算法的设计与实现

利用用户身份属性数据进行聚类, 改进算法的具体步骤如下:

(1) 用户身份属性数据预处理. 用户身份属性数据主要包括年龄、性别、职业、专业等. 年龄定义为数值数据. 性别定义为二元数据, 即输入性别数据时, 可以根据实际内容对应转化为二元数据 0 和 1(输入性别: 男或 1). 职业、专业等数据定义为标称型数据, 使用数值标号的形式进行标准化. 通过以上方式完成用户身份属性数据的预处理工作, 用户属性表达形式为 User=(35, 1, 12, 6), 表示用户是年龄为 35 左右从事数学专业的男教师.

(2) 采用 K-means 聚类算法实现用户身份属性聚类. 主要实现流程如图 1 所示. 算法的时间复杂度 $T(n)=O(n\times k\times t)$, n 代表对象总数, k 代表类簇的个数, t 代表迭代次数.

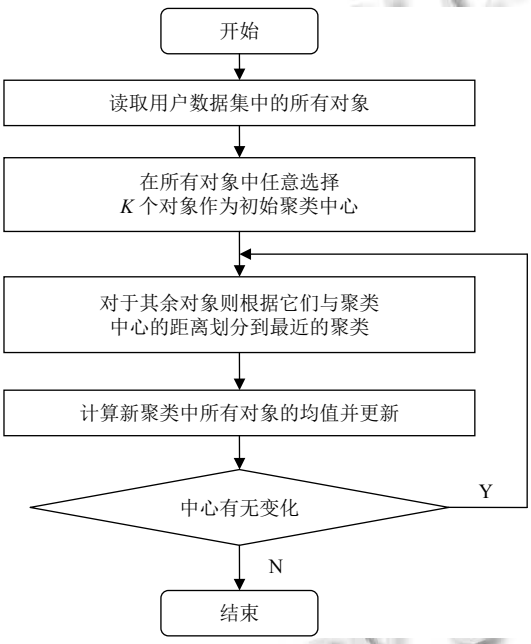

图 1 K-means 聚类算法的流程

(3) 对用户属性数据聚类处理后, 再进一步实现 User-based 协同过滤推荐算法.

3 基于项目划分的 Item-based 协同过滤推荐算法

3.1 问题分析及改进思路

传统的 Item-based 协同过滤推荐算法所分析的用户-项目评分矩阵数据客观存在数据过于稀疏的问题, 会影响相似度计算的准确性, 另外也可能存在用户在评分过程中会因为某种特殊原因给某个项目很高分或很低分, 此情况的发生也会给相似度的计算造成偏差, 而相似度计算的准确性是推荐结果质量的保障.

针对以上问题分析提出在相似度计算中结合项目划分然后再与项目评分共同计算, 引入综合相似度概念. 思路是对项目进行划分类别的预处理, 预处理过程主要是对项目类别进行定义, 然后再计算其相似度. 这样处理后不仅能够对用户-项目评分矩阵进行较好的数据填充, 还能有效的提高相似度计算的准确性. 项目划分类别对于项目之间有层次关系的可以定义项目类别树, 通过计算两个项目距离共同父节点的长度计算彼此之间的类别相似度. 对于彼此之间没有层次关系的则可进行平行划分, 需要着重关注不同类别之间的相关性对计算项目类别相似度的影响, 例如男生喜欢看武侠小说, 女生喜欢看爱情小说, 看似不同类, 但相似度却很高. 综合相似度就是融合了评分相似度与划分类别相似度, 通过加权系数综合计算项目相似度. 另外考虑到两个项目共同评分的用户数越多, 其相似性越高, 所以在计算时要加入共同评分用户数的因素. 最后关于目标用户最近邻个数的确定问题, 考虑到用户数较多对近邻个数选取的影响, 采用阈值法, 动态选取最近邻, 避免了固定值法的负效应.

3.2 改进算法的设计与实现

由改进思路可得, 综合相似度计算如式 (3) 所示.

$$Sim\left(I_i,I_j\right)=(1-\alpha)Sim_r\left(I_i,I_j\right)+\alpha Sim_c\left(I_i,I_j\right) \quad (3)$$

其中, $Sim_r(I_i, I_j)$ 表示项目评分相似度, $Sim_c(I_i, I_j)$ 表示项目划分类别相似度, α 是加权系数.

具体实现过程如下:

(1) 项目划分类别相似度计算. 结合如上讨论, 将项目划分的类别表示为 $p=\{p_1, p_2, p_3, \cdots, p_m\}$, 项目 I_i 所属类别 $P_i=\{p_x, p_y, \cdots\}$, 项目之间同属的相同类别越多相似性越近, 但不同类别的相关性情况也要考虑, 例如男生喜欢看武侠小说, 女生喜欢看爱情小说, 男生和武侠小说虽然不是同一类, 但彼此之间的相似度显然比同属于一类的武侠小说与爱情小说的相似度更高, 通过分析思考定义 $m\times m$ 的项目类别相似性矩阵 S_{mm}, 如式 (4) 所示.

$$S_{mm}=\begin{bmatrix} s_{11} & \cdots & s_{1m} \\ \vdots & \ddots & \vdots \\ s_{m1} & \cdots & s_{mm} \end{bmatrix} \quad (4)$$

方阵中 S_{ij} 为类别 p_i, p_j 的相似度, 计算方式如式

(5) 所示.

$$S(p_i,p_j)=\frac{v_i\cap v_j}{v_i\cup v_j} \tag{5}$$

其中, v_i 表示属于类别 p_i 的总个数, v_j 表示属于类别 p_j 的总个数, $s(p_i, p_j)$ 为同属于类别 p_i, p_j 的个数与属于类别 p_i 或类别 p_j 的个数的比值, 项目划分类别相似度计算如式 (6) 所示.

$$Sim_c(I_i,I_j)=\begin{cases}\dfrac{p_i\cap p_j}{p_i\cup p_j} & p_i\cap p_j\neq\varphi\\ \max(s(p_i,p_j)) & p_i\cap p_j=\varphi\end{cases} \tag{6}$$

式 (6) 表示项目 I_i 所属的类别 P_i 与项目 I_j 所属的类别 P_j, 两者没有共同类别时, 计算结果值为项目 I_i 与项目 I_j 分别所属类别之间的相似度的最大值.

(2) 考虑项目共同评分用户数对相似度的影响, 改进评分相似度计算. 对传统 Pearson 相似度计算公式 (2) 结合共同评分用户数, 融入相似度计算如式 (7) 所示.

$$Sim_r(I_i,I_j)=\frac{U_i\cap U_j}{U_i\cup U_j}*Sim^b(I_i,I_j) \tag{7}$$

式中, $U_i\cap U_j$ 是指对项目 I_i 和 I_j 共同评过分的用户总数, $U_i\cup U_j$ 是指对项目 I_i 或 I_j 所有评过分的用户总数.

(3) 确定最近邻. 采用综合相似度计算公式分别计算目标项目 I_i 与其它所有项目的综合相似度值 $Sim(I_i, I_j)(1\leq j\leq n$, 其中 $j\neq i)$, 结果进行排序并选取最相似的前 K 个项目为目标项目 I_i 的最近邻居集合, 显然 K 值的选取直接影响推荐结果的质量. 结合如上讨论, 采用项目相似性邻居选取阈值 β 动态选取最近邻, 考虑平均相似度因素, 得到最近邻算法如式 (8) 所示.

$$K(I_i)=\left\{I_j\middle|Sim(I_i,I_j)-\overline{Sim_{I_i}}>\beta,j\neq i\right\} \tag{8}$$

确定最近邻时选取与目标项目 I_i 的相似度大于平均相似度与 β 之和的项目为目标项目 I_i 的最近邻居集合.

4 实验结果及分析

4.1 实验数据

本实验使用 GroupLens 提供的 MovieLens 电影评分数据集, 数据中有用户特征信息、电影属性信息、用户对电影的评分信息等. 评分数据的范围是从 1 到 5 的整数, 电影划分为 19(0–18) 个不同类别. 实验采用 1 MB 的数据集, 其中包括 6040 个用户对 3900 部电影的 1000 209 条评分数据.

4.2 实验评估标准

本实验通过平均绝对偏差 (*MAE*) 值来评估推荐算法质量, *MAE* 值越小, 说明预测值和真实值之间的误差越小, 预测的准确度越高, 推荐质量越优. 用 N 表示实验测试项目数, P_i 表示预测评分值, R_i 表示实际评分值, *MAE* 的计算如式 (9) 所示.

$$MAE=\frac{\sum_{i=1}^{N}|P_i-R_i|}{N} \tag{9}$$

4.3 实验结果及分析

为了验证提出的基于用户聚类与项目划分的协同过滤推荐算法的有效性, 通过以下实验验证.

(1) 基于用户属性聚类的 User-based 协同过滤推荐算法, 改进后是否能够提高推荐质量, 聚类个数 K 值及最近邻居个数都需要通过实验确定. 实验采用 1 MB 的数据集, 以 *MAE* 值作为推荐算法质量的衡量标准, 确定 K 值实验结果如图 2 所示.

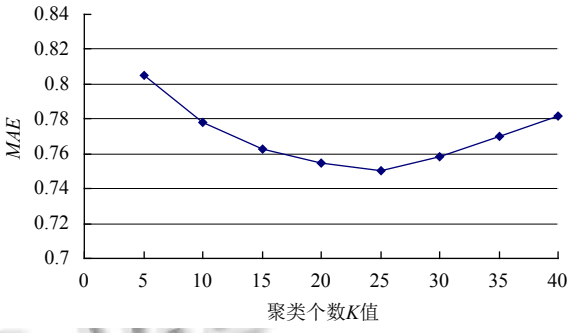

图 2 确定聚类个数 K 值

可以看出, K 值的确定对推荐算法的推荐质量有直接影响, K 值过小或过大都会引起 *MAE* 值的升高, 也就是误差增大. 当 K=25 时, 推荐效果最优, 平均绝对偏差 *MAE* 值最小. 为了进一步确定最近邻居个数, 分别使用不同的 K 值比较最近邻居个数和 *MAE* 值的大小, 确定近邻 N 值实验结果如图 3 所示.

实验可得, 近邻个数的逐渐增加使平均绝对偏差 *MAE* 值先减小又逐渐增大. 不同的 K 值和近邻个数相比较可以看出当 K 取值为 25 且近邻个数为 30 时, *MAE* 值最低, 推荐质量效果最好.

由以上实验结论可进一步实验验证改进算法和传统的 User-based 协同过滤推荐算法的推荐结果比较, 通过以上分析近邻个数 N 值的不同会直接影响算法结

果, 所以实验通过近邻个数 N 的不同取值比较两种算法的推荐效果, 比较结果如图 4 所示.

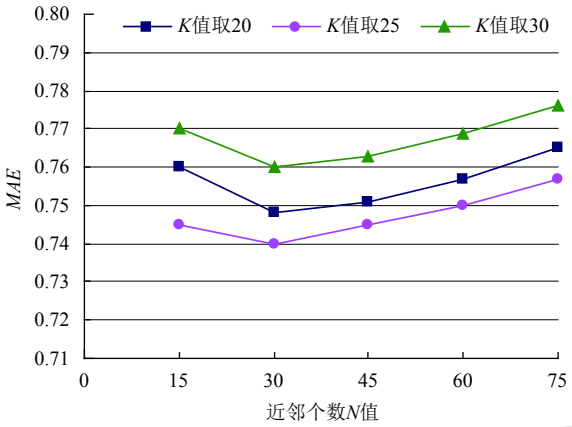

图 3 确定近邻个数 N 值

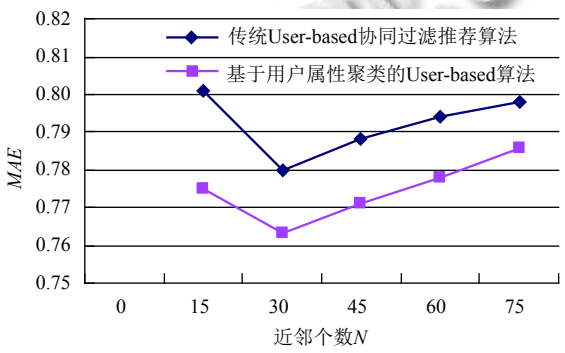

图 4 传统算法与改进算法推荐效果比较

实验表明基于用户属性聚类的 User-based 协同过滤推荐算法在不同近邻个数的环境下比传统的 User-based 协同过滤推荐算法 *MAE* 值都小, 说明改进算法能够有效的提高推荐质量.

(2) 基于项目划分的 User-based 协同过滤推荐算法, 改进后的算法采用综合相似度的计算方法计算项目相似度, 通过实验确定系数 α 的权值, 实验结果如图 5.

由实验结果可知, 不同的数据集在加权系数 α 值增加过程中相应的 *MAE* 值均是先减小然后再增大, 且影响表现一致, 当 α 取值为 0.2 时, 各数据集的 *MAE* 值最低, 达到最优推荐质量. 说明综合相似度计算中项目评分数据占的比重更高.

(3) 将基于用户属性聚类与项目划分的协同过滤推荐算法 (即改进算法) 与传统的协同过滤推荐算法 (CFRA), 并同时选取文献[8,15,16]提出的算法 (分别简写为 SCCF、CRF、UCSP) 通过实验与本文所提出的算法进行比较, 各算法的推荐效果对比实验结果见表 1.

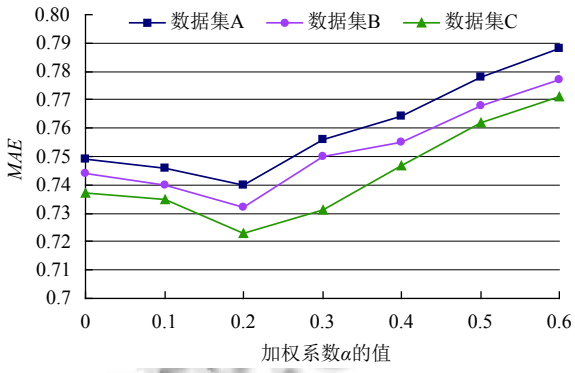

图 5 确定加权系数 α 的值

表 1 多种算法推荐效果对比结果表

算法	数据集 A	数据集 B	数据集 C	数据集 D
CFRA	0.824	0.817	0.805	0.821
SCCF	0.757	0.739	0.744	0.755
CRF	0.748	0.726	0.739	0.746
UCSP	0.735	0.714	0.723	0.734
改进算法	0.716	0.705	0.712	0.709

对比结果可知, 与其他算法相比, 因本文提出的改进算法对推荐准确度有重要影响的相似度计算进行了充分考虑, 结合用户属性及项目类别划分计算相似度, 并且在项目最近邻选取时采用阈值计算, 因此 *MAE* 的值均最小, 有效提高推荐精度, 能为用户推荐更准确的项目.

5 结论

提出一种基于用户属性聚类与项目划分的协同过滤推荐算法, 实验结果证明, 所提算法能够有效提高推荐精度, 为用户提供更加准确和优质的推荐项目. 下一步将结合用户兴趣变化以及社交数据等因素对推荐算法完善进行研究.

参考文献

1 黄立威, 江碧涛, 吕守业, 等. 基于深度学习的推荐系统研究综述. 计算机学报, 2018, 41(7): 1619–1647.
2 Frémal S, Lecron F. Weighting strategies for a recommender system using item clustering based on genres. Expert Systems with Applications, 2017, 77(7): 105–113.
3 黎新志, 高茂庭. 基于用户分类的隐含因子模型研究. 计算机应用研究, 2018, 35(8): 2289–2292. [doi: 10.3969/j.issn.1001-3695.2018.08.012]

4 王志虎, 黄曼莹. 基于用户历史行为的协同过滤推荐算法. 微电子学与计算机, 2017, 34(5): 132–136.
5 石进平, 李劲, 和凤珍. 基于社交关系和用户偏好的多样性图推荐方法. 计算机科学, 2018, 45(S1): 423–427.
6 Katarya R, Verma O P. A collaborative recommender system enhanced with particle swarm optimization technique. Multimedia Tools and Applications, 2016, 75(15): 9225–9239. [doi: 10.1007/s11042-016-3481-4]
7 王光, 张杰民, 董帅含, 等. 基于内容的加权粒度序列推荐算法. 计算机工程与科学, 2018, 40(3): 564–570. [doi: 10.3969/j.issn.1007-130X.2018.03.024]
8 孙辉, 马跃, 杨海波, 等. 一种相似度改进的用户聚类协同过滤推荐算法. 小型微型计算机系统, 2014, 35(9): 1967–1970. [doi: 10.3969/j.issn.1000-1220.2014.09.006]
9 于琨, 孙波, 海本斋. 基于超图排序和组稀疏最优化的推荐系统. 计算机工程与设计, 2018, 39(7): 1996–2001.
10 杨丰瑞, 郑云俊, 张昌. 结合概率矩阵分解的混合型推荐算法. 计算机应用, 2018, 38(3): 644–649.
11 田保军, 胡培培, 杜晓娟, 等. Hadoop 下基于聚类协同过滤推荐算法优化的研究. 计算机工程与科学, 2016, 38(8): 1615–1624. [doi: 10.3969/j.issn.1007-130X.2016.08.016]
12 丁少衡, 姬东鸿, 王路路. 基于用户属性和评分的协同过滤推荐算法. 计算机工程与设计, 2015, 36(2): 487–491, 497.
13 杨尚君, 孙永维, 庞宇. 基于改进鱼群算法的多无人机任务分配研究. 计算机仿真, 2015, 32(1): 69–72, 102. [doi: 10.3969/j.issn.1006-9348.2015.01.015]
14 王颖, 王欣, 唐万梅. 融合用户自然最近邻的协同过滤推荐算法. 计算机工程与应用, 2018, 54(7): 77–83.
15 杨兴雨, 李华平, 张宇波. 基于聚类和随机森林的协同过滤推荐算法. 计算机工程与应用, 2018, 54(16): 152–157. [doi: 10.3778/j.issn.1002-8331.1712-0089]
16 高茂庭, 段元波. 结合用户聚类和评分偏好的推荐算法. 计算机应用研究, 2018, 35(8): 2260–2264. [doi: 10.3969/j.issn.1001-3695.2018.08.005]