·人工智能及识别技术· 文章编号:1000-3428(2017)08-0215-04 文献标志码:A 中图分类号:TP301.6

基于用户项目体验度的协同过滤推荐算法

郑苏洋,姜久雷,王晓峰

(北方民族大学 计算机科学与工程学院,银川 750021)

摘 要: 在用户相似度计算基础上,根据用户偏好以及项目特征对用户评分产生的影响,提出一种针对用户项目体验度的推荐算法。阐述项目体验度对用户产生的潜在影响,选择皮尔森相似性计算公式做进一步计算。通过用户对项目的好评数以及给项目的评分分别占该项目的总评数和总体项目评分中的比例,获得用户对项目的体验度权重。采用长尾理论平衡用户相似性和用户对流行项目的关注度,计算得出用户相似度并产生预测和推荐。实验结果表明,与传统协同过滤算法相比,该算法提高了相似度计算准确度,并能改善数据稀疏情况下的推荐效果。

关键词: 推荐系统;协同过滤;用户项目偏好;用户体验度;长尾理论

中文引用格式: 郑苏洋,姜久雷,王晓峰. 基于用户项目体验度的协同过滤推荐算法[J]. 计算机工程,2017,43(8):215-218,224.

英文引用格式: Zheng Suyang,Jiang Jiulei,Wang Xiaofeng. Collaborative Filtering Recommendation Algorithm Based on User Project Experience Degree[J]. Computer Engineering,2017,43(8):215-218,224.

Collaborative Filtering Recommendation Algorithm Based on User Project Experience Degree

ZHENG Suyang,JIANG Jiulei,WANG Xiaofeng

(College of Computer Science and Engineering,Beifang University of Nationalities,Yinchuan 750021,China)

【Abstract】 On the basis of the general user's similarity calculation,a recommendation algorithm based on user project experience degree is proposed according to the impact of user preferences and project characteristics on user ratings. This paper describes the potential impact of the project experience degree on the user,and selects Pearson similarity formula for further calculation. Through the number of users praise for the projects accounting for the total number of project reviews and the proportion of the user's rating to the overall project score,the user experience weight for the project is gained. The long tail theory is used to balance the user's similarity and the user's attention to the popular items. The user's similarity is calculated,and the prediction and recommendation are generated. Experimental results show that,compared with the traditional collaborative filtering algorithm, the proposed algorithm improves the accuracy of similarity computation and the recommendation effect under sparse data.

【Key words】 recommendation system; collaborative filtering; user project preference; user experience degree; long tail theory

DOI: 10.3969/j.issn.1000-3428.2017.08.036

0 概述

基于项目的协同过滤推荐算法首先计算项目与项目的相似程度,判断出若干个与目标项目相关的相邻集,通过计算各项目的评分,得出未评分项目的预测评分值。经证实基于项目的推荐算法性能更优,能解决传统算法面临的稀疏性和可扩展性问题[1]。但从批判的角度指出基于项目的协同过滤推荐的准确性与数据大小和计算量有关,普遍情况下还是基于用户的协同过滤推荐较准确[2]。

协同过滤推荐算法是目前使用度最高的推荐算法之一。它的思想是:如果不同用户对某项目的评分值相差无几,那么这些用户对其他项目的兴趣、体验度应该也是相近的[3]。但是,随着大数据时代的到来,推荐算法所需要计算的数据量剧增,算法复杂度变高。由于互联网的开放性,网络安全越来越引起人们的重视。用户为了自身信息安全对网络信息持保留态度,这些因素导致了推荐算法的数据稀疏

基金项目: 国家自然科学基金"实例结构限制下信息传播算法的收敛性研究"(61462001);宁夏高校科研项目(NGY2015151)。

作者简介: 郑苏洋(1991—),女,硕士研究生,主研方向为推荐算法、信息系统、软件工程;姜久雷,教授、博士;王晓峰,讲师、博士。

收稿日期: 2016-07-18 **修回日期:** 2016-09-08 **E-mail:** 13649507955@163.com

问题,同时也降低了算法的精度。因此,许多学者针对具体现象提出了一些改进算法。例如,文献[4]通过类别分类的标签,利用贝叶斯网络对条件概率的精确计算提高了算法的时效性。针对大数据的稀疏性,文献[5]从用户兴趣和景点流行度着手,针对核心用户结合上下文感知产生基于传统算法的推荐算法,有效改善算法精确度。文献[6]通过计算用户评分与项目属性来考虑用户的偏好问题,借此提出UPPPCF 算法,弥补了传统算法仅靠评分值计算相似度的不足。但以上文献也都存在不足:首先,没有从用户角度考虑项目的知名度和体验好感度对用户选择产生的潜在影响。再者,仅仅计算两两用户之间的相似度并不能代表 2 个用户真的很相似,很多用户不仅会根据自己的喜恶来评判项目的好坏,也会根据身边用户或相似用户对项目的口碑来选择项目。也就是说项目的体验度和流行度会潜在地影响用户的判断和选择。

本文考虑项目体验度对目标用户的影响,在用户体验度方面改进算法,融合用户评分和项目评分平均值,并结合项目体验度等权重得出推荐列表。

1 传统协同过滤推荐算法

该算法的具体思想是:首先从已有数据中查找与目标用户相似的邻居集,然后进行挑选,从类似用户中找出目标用户喜欢或者从未接触却有可能喜欢的项目向目标用户进行针对性推荐。计算用户相似度方法分别是:

1)余弦相似性

建立一个 n 维向量,表示用户对每个项目的评分,用户之间的相似程度由向量间的余弦夹角表示。如果评分为 0 则用户对项目没有评分。设用户 m,n 之间的相似度为:

$$sim(m,n)=\frac{\sum_{a\in A_{m,n}}R_{m,a}R_{n,a}}{\sqrt{\sum_{a\in A_m}R_{m,a}^2}\sqrt{\sum_{a\in A_n}R_{n,a}^2}} \tag{1}$$

其中,$R_{m,a}$,$R_{n,a}$ 分别是用户 m,n 对项目 a 的评分。

2)修正的余弦相似性

式(2)使用评分值与用户对项目的平均分差值的方法来减少式(1)中不同用户的体验度评价标准不一的问题。A_m,A_n分别表示用户 m,n 的共同评分项目集合,则用户 m 和用户 n 的相似度为:

$$sim(m,n)=\frac{\sum_{a\in A_{m,n}}(R_{m,a}-\bar{R}_m)(R_{n,a}-\bar{R}_n)}{\sqrt{\sum_{a\in A_m}(R_{m,a}-\bar{R}_m)^2}\sqrt{\sum_{a\in A_n}(R_{n,a}-\bar{R}_n)^2}} \tag{2}$$

3)Pearson 相关系数

如果用户 m,n 共同评分的项目集合用 $A_{m,n}$ 表示,则用户 m 和用户 n 的相似度为:

$$sim(m,n)=\frac{\sum_{a\in A_{m,n}}(R_{m,a}-\bar{R}_m)(R_{n,a}-\bar{R}_n)}{\sqrt{\sum_{a\in A_{m,n}}(R_{m,a}-\bar{R}_m)^2}\sqrt{\sum_{a\in A_{m,n}}(R_{n,a}-\bar{R}_n)^2}} \tag{3}$$

其中,$R_{m,a}$分别表示用户 m,n 对项目 a 的评分;$\bar{R}_m$ 和 $\bar{R}_n$ 分别表示用户 m 和用户 n 在 $A_{m,n}$上的平均评分,即:

$$\bar{R}_m=\frac{1}{|A_{m,n}|}\sum_{a\in A_{m,n}}R_{m,a},\bar{R}_n=\frac{1}{|A_{m,n}|}\sum_{a\in A_{m,n}}R_{n,a}$$

2 本文算法描述

2.1 用户评分相似度

因为在相似度计算过程中,使用 Person 相似度公式计算用户相似度结果最优[7],所以本文也使用上文提及的式(3)进行相似度计算,具体计算公式如下所示:

$$sim(u,v)=\frac{\sum_{i\in I_{u,v}}(R_{u,i}-\bar{R}_u)(R_{v,i}-\bar{R}_v)}{\sqrt{\sum_{i\in I_u}(R_{u,i}-\bar{R}_u)^2}\sqrt{\sum_{i\in I_v}(R_{v,i}-\bar{R}_v)^2}} \tag{4}$$

其中,$sim(u,v)$为用户 u 和用户 v 的相似度,算法中将所有用户进行两两计算得出相似度值。但传统的皮尔森相似度计算公式仅从用户对项目的评分计算其相似度,并不能精确地代表用户对项目的体验度。因为有的用户会习惯性好评,而有的用户会因个人喜恶对项目产生偏见,从而导致所得到的原始评分矩阵并不客观[8]。所以在皮尔森相似度计算公式上添加项目偏好权重和项目知名度权重。

表 1 中行代表用户 u,列为用户 u 对当前项目 i 的行为,$r_{u,m}$表示用户 n 对项目 m 的评分。大部分情况下评分范围为[1,5]。其中,$i\in[1,m]$,$u\in[1,n]$。若评分为 0 表示用户对项目没有行为。

表 1 用户项目评分矩阵

用户 u	项目 i_1	项目 i_2	…	项目 i_m
u_1	r_{11}	r_{12}	…	r_{1m}
u_2	r_{21}	r_{22}	…	r_{2m}
⋮	⋮	⋮		⋮
u_n	r_{n1}	r_{n2}	…	r_{nm}

2.2 项目体验度

虽然通过用户项目评分、对项目属性进行挖掘可以分析出用户对项目的喜好程度,但并没有将用户体验度对项目选择的影响考虑在内,如果一个项目的好评数相对其他项目高,那么该项目的口碑和用户体验度相对就会高于其他项目。很多用户都是凭借身边朋友的使用体验度来选择项目,所以,项目的体验度会影响用户的选择。将用户对项目的评价分为好评、中评和差评,项目的好评数越多,用户对

项目的体验度就越好,因为差评会影响用户对项目体验度的计算,故本文通过式(5)计算了项目好评、中评权重,据此提出了结合项目属性的过滤算法,即考虑项目体验度的问题。

$$weight(U,I)=\frac{1}{\sum\limits_{\substack{i\in I\\u\in U}}\frac{G_{u,i}+M_{u,i}}{S_m\times X_{u,i}}} \tag{5}$$

其中,$X_{u,i}$表示用户 u 对项目 i 的总评数;$M_{u,i}$表示用户 u 对项目 i 的中评数;$G_{u,i}$表示用户 u 对项目 i 的好评数;S_m表示项目 i 的总评数。

表 2 计算出每一项目的总评数和好评数,好评数越高,占总评的比值越高,说明用户体验度高,该项目越受用户的欢迎,目标用户对它感兴趣的可能性就会越高。

表 2　用户项目评分数统计矩阵

评论数	i_1	i_2	…	i_m
G	G_1	G_2	…	G_m
M	M_1	M_2	…	M_m
X	X_1	X_2	…	X_m

分别结合用户项目类别偏好和项目知名度,提出如下公式:

$$weight_{in}(u,v)=R_{u,i}\times\sum_{\substack{i\in I\\u\in U}}weight(U,i)\times\sum_{k\in T_i}weight\_tag(u,i,t_k') \tag{6}$$

其中,$weight_tag(u,i,t')$表示用户 u 对项目 i 的类别偏好,若项目 i 没有该类型则取值为 0。$R_{u,i}$为用户 u 对项目 i 的评分。所以用户对项目的偏好相似度如式(7)所示。

$$Sim_{in}(u,v)=\frac{\sum\limits_{i\in P_{u,v}}(Weight_{in}(u,i)-\overline{Weight\_in_u})\times(Weight_{in}(v,i)-\overline{Weight\_in_v})}{\sqrt{\sum\limits_{i\in P_u}(Weight_{in}(u,i)-\overline{Weight\_in_u})^2}\sqrt{\sum\limits_{i\in P_v}(Weight_{in}(v,i)-\overline{Weight\_in_v})^2}} \tag{7}$$

所以用户的综合相似度为:

$$Sim(u,v)=\alpha\times Sim(m,n)+\beta\times Sim_{in}(u,v)+(1-\alpha-\beta)\times Sim(m,n)\times Sim_{in}(u,v) \tag{8}$$

其中,$\alpha\in(0,1)$,$\beta\in(0,1)$。

通常情况下,根据二八法则:如果 80% 的用户对项目进行过评分,无论评分高低,说明项目的流行度已经很高了,那么,再向用户推荐该项目就没有太大的意义。用户即使没有评分,也有很大可能性已经了解该项目。如果只有 20% 的用户对项目进行过评分,那么该项目就符合长尾法则,相应的推荐就会给用户带来更好的体验度[9]。

反而言之,如果一个项目非常热门,那么被推荐给用户的概率可能是冷门项目的 n 倍,所以,计算综合相似度时,利用长尾法则公式平衡了流行度较高的项目和比较冷门但用户评分普遍较好的项目,结合评分预测方法,本文提出最终用户相似度计算式:

$$sim_{\text{fin}}(u,v)=sim(u,v)\times\sum_{i\in U}\frac{1}{\log_a 1+|N(i)|} \tag{9}$$

其中,$\frac{1}{\log_a 1+|N(i)|}$降低了用户 u 和用户 v 共同感兴趣类表中的热门物品对其相似度的影响;U 为用户 u 和用户 v 共同感兴趣的用户集合。

2.3　评分预测

采用传统的预测方法得到预测评分:

$$p_{u,i}=\bar{R}_u+\frac{\sum\limits_{v\in N_u}sim_{\text{fin}}(u,v)\times(R_{v,i}-\bar{R}_v)}{\sum\limits_{v\in N_u}|sim_{\text{fin}}(u,v)|} \tag{10}$$

其中,N_u表示用户 u 的最近邻居集;$\bar{R}_u$ 和 $\bar{R}_v$ 表示用户 u,v 的评分平均值。

3　算法描述

算法步骤如下:

输入　用户评分矩阵,项目类别矩阵

输出　推荐集

1)根据皮尔森相关相似性公式(式(4))计算用户之间的相似度 $sim(u,v)$。

2)根据式(5)$weight(U,I)$计算项目流行度。

3)结合上式项目流行度和式(6)提出一种新的公式(式(7)),即通过用户项目类别偏好和项目体验度计算用户项目类别偏好。

4)根据皮尔森相关相似性计算公式(式(4))和用户项目相似度公式(式(7))产生综合用户项目相似度式(8)。

5)根据长尾法则结合式(8)提出新的计算公式(式(9)),平衡流行度较高的项目和比较冷门但用户评分普遍较好的项目,最后产生式(10),计算出目标用户对未评分项目的评分。

4　实验结果与分析

4.1　实验环境

实验语言为 Java,实验平台 Eclipse,实验数据为 Movielens 数据集,由美国明尼苏达大学的 GroupLens 项目组创办。通过 943 名用户对1 682 个项目产生的 100 000 条评分数据进行算法训练和验证。数据集的稀疏度为 93.7%[10]。

4.2　评价标准

针对协同过滤推荐算法性能优劣的衡量方法有很多,比如平均绝对误差(Mean Absolute Error,MAE)、召回率、均方根偏差(Root Mean Square Deviation,RMSE)[11]、项目新颖度、覆盖率(Coverage Rate,COV)

等。本文主要采用最常见的 MAE 评价标准。

目标用户未评分项目的预测评分值由 sp_n 表示，sr_n 则是真实的用户评分。N 为目标用户 u 待预测的项目总数。通过 sp_n 与 sr_n 的差与待预测项目总数的比值得出 MAE 值。MAE 的值越小说明算法的推荐精度和准确度越高。MAE 公式如下所示：

$$MAE = \frac{\sum_{n=1}^{N} | sp_n - sr_n |}{N} \tag{11}$$

可使用以下公式表示所有待预测用户的总 MAE 值：

$$MAE = \frac{\sum_{u}^{U} MAE_u}{U} \tag{12}$$

其中，MAE_u 表示用户 u 的 MAE 值；U 表示所有预测用户数。

4.3 评价结果及分析

为了统一和其他算法的 MAE 值进行对比，本文将最近邻居数取值范围定为 5 ~ 30[12]。又由于权重值会影响算法的推荐结果，因此进行了不同权重值的对比实验，借此选取最恰当的 α 和 β 值。表 3 表示了预测结果部分加权值下本文算法的 MAE 值。由图可知，当 $\alpha = 0.3$，$\beta = 0.2$ 时，算法精确度最高，故本文采用此权值进行计算并与其他算法进行 MAE 值比较。

表 3 不同权值下算法的 MAE

α	β	最近邻居数为 5	最近邻居数为 10	最近邻居数为 15	最近邻居数为 20	最近邻居数为 25	最近邻居数为 30
0.00	0.15	0.733 7	0.727 0	0.726 6	0.726 7	0.728 0	0.729 0
0.15	0.15	0.733 6	0.727 4	0.726 3	0.726 5	0.727 6	0.728 4
0.20	0.30	0.734 0	0.727 1	0.726 4	0.726 6	0.728 1	0.728 7
0.30	0.20	0.733 9	0.727 2	0.726 3	0.726 6	0.715 4	0.728 5
0.35	0.15	0.734 3	0.727 6	0.726 5	0.726 7	0.727 6	0.728 4
0.40	0.30	0.733 4	0.727 6	0.726 2	0.726 7	0.727 6	0.728 6
0.55	0.25	0.734 3	0.727 6	0.726 4	0.726 7	0.727 7	0.728 5
0.60	0.40	0.733 9	0.727 5	0.726 1	0.726 6	0.727 6	0.728 6
0.75	0.25	0.734 1	0.727 5	0.726 6	0.726 8	0.727 7	0.728 5
0.80	0.20	0.735 0	0.727 6	0.726 5	0.726 8	0.727 8	0.728 5
0.95	0.00	0.736 0	0.728 2	0.726 6	0.726 8	0.728 1	0.729 0
1.00	0.00	0.736 0	0.728 3	0.7266	0.726 8	0.728 1	0.729 0

由图 1 可知，在邻居数为 5 ~ 25 之间，传统算法 MAE 值均高于 1，而从协同过滤中基于用户兴趣度的相似性度量算法[13]可以看出，随着邻居数的增加精确度逐渐提高，但邻居数越少，计算精度越差，推荐质量越低。基于用户兴趣点和特征的优化协同过滤推荐算法[14]介于 0.7 ~ 0.8 之间，折线图相较平稳，而本文算法相对基于用户兴趣点和特征的优化协同过滤推荐算法在不同邻居数下计算精度更加平稳。该算法从邻居数 5 ~ 25，MAE 值差 7.02%，而本文算法差 0.65%。以上结果说明，本文提出的算法在一定程度上提高了相似度计算准确度，并有效改善了数据稀疏下的推荐算法。

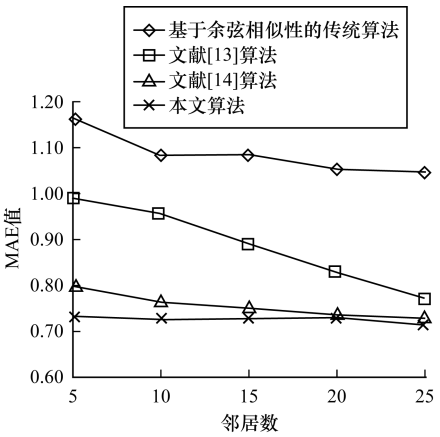

图 1 不同算法的 MAE 值比较

5 结束语

针对项目体验度和流行度对目标用户存在的潜在影响，本文通过提出项目特征权重，结合用户对项目类型的感兴趣程度来改进协同过滤推荐算法的精确度，给用户带来更好的体验度。又因为邻居用户中普遍好评的项目通过邻居用户的传播就会产生很好的反响，而用户对项目差评数较多的项目，即便推荐给目标用户也不一定会达到很好的用户体验度，所以提出项目体验度，通过用户对项目评价中的好评和中评数体现用户对该项目的满意程度，结合用户偏好预测推荐评分并产生推荐。实验结果表明，本文提出的算法在一定程度上提高了 MAE 值。但是，在改进算法的过程中忽略了用户属性与项目类型之间存在的潜在关系，并且没有考虑评论时间对推测结果产生的影响[15]以及冷启动问题[16]，下一步将对这些问题进行研究与改进。

参考文献

[1] 郁 雪. 基于协同过滤技术的推荐方法研究[D]. 天津：天津大学，2009.

[2] 项 亮. 推荐系统实践[M]. 北京：人民邮电出版社，2012.

[3] 李春喜. 一种混合模式电子商务推荐技术的研究[D]. 苏州：苏州大学，2010.

[4] 温 梅. 个性化推荐中基于贝叶斯网络的用户兴趣模型研究[D]. 武汉：华中师范大学，2013.

[5] 于蓓佳. 基于时空数据的旅游信息推荐研究[D]. 济南：山东师范大学，2015.

[6] 姚平平，邹东升，牛宝君. 基于用户偏好和项目属性的协同过滤推荐算法[J]. 计算机系统应用，2015，24(7)：15-21.

[7] 刘 淇. 基于用户兴趣建模的推荐方法及应用研究[D]. 合肥：中国科学技术大学，2013.

（下转第 224 页）

身特征外,还考虑句子与标题之间主题相关性和语义相关性。本文方法的抽取效果比 Title 方法仍有提高,主要原因是综合考虑事件之间的相互关联关系,加入了影响句子权重的 3 个影响因子,同时对事件句进行真伪辨别。

5 结束语

本文针对原子事件级别的事件抽取粒度过细、主题级别的事件抽取粒度较粗、精确提取事件信息的效率较差的问题,提出一种新的新闻事件主题句提取方法。该方法利用 ICTCLAS 中文处理模块对新闻文本进行处理,如分句、停用词过滤、分词等;采用词语位置加权的 TextRank 方法提取关键词,并计算文本句子关键词的 MI 值,将句子划分为事件句和非事件句;利用 TextRank 模型对事件句建立关系图模型,引入位置关系、句子相似度和关键词覆盖率 3 个影响因子,计算事件句之间的影响力权重,从而迭代算出权重最高的句子作为关键事件主题句。实验结果表明,针对单文档的关键事件主题句提取,与基于 TF-IDF 和 Title 的关键事件主题句抽取方法相比,本文方法的抽取效果更好。

参考文献

[1] 韩客松,王永成,沈 洲,等. 三个层面的中文文本主题自动提取研究[J]. 中文信息学报,2001,15(4):20-26.

[2] 马颖华,王永成,苏贵洋. 一种基于字共现频率的汉语文本主题抽取方法[J]. 计算机研究与发展,2003,40(2):874-878.

[3] 温有奎,温 浩,徐瑞颐,等. 基于创新点的知识元挖掘[J]. 情报学报,2005,24(6):664-668.

[4] 张云涛,龚 玲,王永成. 基于综合方法的文本主题句的自动抽取[J]. 上海交通大学学报,2006,40(5):771-774,782.

[5] Biadsy F, Hirschberg J, Filatova E. An Unsupervised Approach to Biography Production Using Wikipedia[C]// Proceedings of Meeting of the Association for Computational Linguistics. Columbus, USA: [s. n.], 2008:807-815.

[6] 何 维,王 宇. 基于句子关系图的网页文本主题句抽取[J]. 现代图书情报技术,2009(3):57-61.

[7] 刘 茵,李弼程,郭映月. 一种基于聚类算法的主旨句提取方法[J]. 情报学报,2008,27(1):49-55.

[8] 张其文,李 明. 文本主题的自动提取方法研究与实现[J]. 计算机工程与设计,2006,27(15):2744-2746,2766.

[9] 薛扣英,原 盛,张心严. 基于 WFC 和 MI 的主题句提取方法[J]. 计算机工程,2009,35(20):184-186.

[10] 王 伟,赵东岩,赵 伟. 中文新闻关键事件的主题句识别[J]. 北京大学学报(自然科学版),2011,47(5):789-796.

[11] 孔 胜,王 宇. 基于句子相似度的文本主题句提取算法研究[J]. 情报学报,2011,30(6):605-609.

[12] 葛 斌,李芳芳,李 阜,等. 基于无向图构建策略的主题句抽取[J]. 计算机科学,2011,38(5):181-185.

[13] 顾益军,夏 天. 融合 LDA 与 TextRank 的关键词提取研究[J]. 现代图书情报技术,2014,30(7):41-47.

[14] 王 力,李培峰,朱巧明. 一种基于 LDA 模型的主题句抽取方法[J]. 计算机工程与应用,2013,49(2):160-164.

[15] 夏 天. 词语位置加权 TextRank 的关键词抽取研究[J]. 现代图书情报技术,2013,237(9):30-34.

[16] 方 康,韩立新. 基于 HMM 的加权 Textrank 单文档的关键词抽取算法[J]. 信息技术,2015(4):114-116,120.

编辑 顾逸斐

(上接第 218 页)

[8] Castro-Schez J J, Miguel R, Vallejo D, et al. A Highly Adaptive Recommender System Based on Fuzzy Logic for B2C e-commerce Portals[J]. Expert Systems with Applications, 2011, 38(3):2441-2454.

[9] 郝立燕,王 靖. 基于项目流行度的协同过滤 TopN 推荐算法[J]. 计算机工程与设计,2013,34(10):3497-3501.

[10] 李春喜. 一种混合模式电子商务推荐技术的研究[D]. 苏州:苏州大学,2010.

[11] 韩 丽. 社交网络中的信任推荐和好友搜索过滤算法[D]. 秦皇岛:燕山大学,2012.

[12] 高全力,高 岭,杨建锋,等. 融合影响因子的加权协同过滤算法[J]. 计算机工程,2014,40(8):38-42.

[13] 嵇晓声,刘宴兵,罗来明. 协同过滤中基于用户兴趣的相似性度量方法[J]. 计算机应用,2010,30(10):2618-2620.

[14] 严冬梅,鲁城华. 基于用户兴趣度和特征的优化协同过滤推荐[J]. 计算机应用研究,2012,29(2):497-500.

[15] 党 博,姜久雷. 基于评分差异度和用户偏好的协同过滤算法[J]. 计算机应用,2016,36(4):1050-1053.

[16] 朱丽中,徐秀娟,刘 宇. 基于项目和信任的协同过滤推荐算法[J]. 计算机工程,2013,39(1):58-62.

编辑 顾逸斐