计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.190300056

基于评分偏好和项目属性的协同过滤算法

朱 磊 胡沁涵 赵 雷 杨季文
苏州大学计算机科学与技术学院 江苏 苏州 215006
(2247684006@qq.com)

摘 要 针对传统的协同过滤推荐由于数据稀疏性导致物品间相似性计算不准确、推荐准确度不高的问题,文中提出了一种基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法,通过改进物品相似度度量公式来提高推荐的准确度。首先考虑到不同用户的评分习惯存在差异这一客观现象,引入评分偏好模型,通过模型计算出用户对评分类别的偏好,以用户对评分类别的偏好来代替用户对物品的评分,重建用户-物品评分矩阵;其次基于时间效应,引入时间权重因子,将时间因素纳入评分相似度计算中;然后结合物品的属性,将物品属性相似度和评分相似度进行加权,完成物品最终相似度的计算;最后通过用户偏好公式来计算用户对候选物品的偏好,依据偏好对用户进行 top-N 推荐。在 MovieLens-100K 和 MovieLens-Latest-Small 数据集上进行了充分实验。结果表明,相比已有的经典的协同过滤算法,所提算法的准确率和召回率在 MovieLens-100K 数据集上提高了 9%~27%,在 MovieLens-Latest-Small 数据集上提高了 16%~28%。因此,改进的协同过滤算法能有效提高推荐的准确度,有效缓解数据稀疏性问题。

关键词:评分偏好;时间权重;物品属性;协同过滤;相似度
中图法分类号 TP311

Collaborative Filtering Algorithm Based on Rating Preference and Item Attributes

ZHU Lei,HU Qin-han,ZHAO Lei and YANG Ji-wen
Department of Computer Science and Technology,Soochow University,Suzhou,Jiangsu 215006,China

Abstract Aiming at the impact of data sparsity of traditional collaborative filtering algorithm resulting in inaccuracy of item similarity,this paper proposed an improved collaborative filtering algorithm based on user rating preference model by incorporating time factor and item attributes. The algorithm improves the accuracy by modifying item similarity formula. Firstly,a preference model is introduced by considering the differences of user's rating habits. A user-item rating matrix is rebuilt by replacing user's rating of item with the preference for rating class. Then time weight function is designed and put into rating similarity based on time effect. What's more,item similarity is calculated by incorporating item attributes similarity and rating similarity. Finally,top-N recommendation is completed after calculating user preference for item by the user preference formula. The experiment results suggest that the precision and recall of the proposed algorithm is increased by 9%~27% on the MovieLens-100K dataset and 16%~28% on the MovieLens-Latest-Small dataset than classical approaches. Therefore,the improved algorithm can improve recommendation accuracy and mitigate the problem of data sparsity effectively.

Keywords Rating preference,Time weight,Item attributes,Collaborative filtering,Similarity

1 引言

随着互联网技术的发展,数据信息呈爆炸式增长,人们面临着严重的信息过载问题[1]。搜索引擎和推荐系统是解决信息过载的两种常用工具。用户如果有明确的目标或需求,利用搜索引擎检索数据是比较方便、有效的方式。用户在浏览新闻、网上购物和观看电影等情况下往往更希望被动地了解自己感兴趣的内容,推荐系统提供的个性化推荐能较好地满足用户的这一需求。

传统的个性化推荐方法主要有基于内容的推荐[2-3]、协同过滤推荐[4-8]和混合推荐[9-10],其中最经典的是协同过滤推荐。协同过滤推荐通过收集用户行为数据和分析用户偏好,来为用户推荐潜在感兴趣的物品,被广泛应用于电子商务、邮件过滤和社交网络等领域。

到稿日期:2019-03-15 返修日期:2019-07-1 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家自然科学基金项目(61572335),江苏高校优势学科建设工程资助项目
This work was supported by the National Natural Science Foundation of China (61572335),Priority Academic Program Development of Jiangsu Higher Education Institutions.
通信作者:胡沁涵(huqinhan@suda.edu.cn)

在现实应用中，传统的协同过滤算法面临一系列问题，包括数据稀疏性问题、冷启动问题和扩展性问题[11-13]。在解决数据稀疏性方面，国内外科研人员进行了深入的研究，提出了各种改进方法和思路[14-21]。这些改进算法的推荐准确度虽然有所提高，但仍有很大的提升空间。

在深入研究相关算法后，本文提出了一种基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法(Collaborative Filtering Algorithm Based on User Rating Preference Model by Incorporating Time Factor and Item Properties，PTP-Item-CF)。该算法从用户评分特性、时间因素和物品属性3个方面来改进相似度度量公式，能有效提高推荐的准确度，缓解数据稀疏性问题。本文在MovieLens-100K和MovieLens-Latest-Smal数据集上进行了实验，结果表明，该算法能有效提高推荐的准确率和召回率。

2 相关工作

2.1 研究现状

一些学者将传统协同过滤算法和其他模型相结合，以提高算法在数据稀疏情况下的推荐准确度。Chen等[14]从用户评分量级的角度出发，引入平衡因子，结合传统余弦相似度，提出了一种优化的基于用户的协同过滤算法。Parivash等[15]考虑到不同用户的评分数量和用户之间的影响程度不同，因此引入了非对称影响度，并加权考虑了用户的评分频率，提高了推荐的准确度。Liu等[16]将用户兴趣度权重函数与物品属性进行融合，在一定程度上降低了算法的平均绝对误差。Lee等[17]提出了偏好模型的概念，利用偏好模型修正用户-物品评分矩阵，将偏好模型和传统协同过滤算法相结合，有效提高了算法的准确率和召回率。

上述算法虽然在一定程度上提高了推荐的准确度，但将用户不同时间段的评分信息同等对待，忽略了时间效应对用户兴趣的影响。时间信息对用户兴趣的影响主要体现在两个方面[18]：1)用户兴趣是随时间变化的；2)物品是有生命周期的。Gasmi等[19]提出了一种反映用户兴趣动态变化的协同过滤算法，考虑了用户兴趣随时间变化和用户周期性兴趣变化这两方面因素，定义了一个与时间有关的权重函数，改进了评分预测公式。Wu等[20]将logistic函数作为时间权重函数，将其纳入到用户相似度计算公式和评分预测公式中，并设置一个阈值来判断两个用户间的相似度是否受时间因素的影响。Jin等[21]从时间上下文信息、用户兴趣衰退函数和物品相似度3个方面来考虑时间因素的影响，通过加权由非线性时间函数计算的用户兴趣和由最近时间段物品相似度计算的用户兴趣来计算用户对物品的偏好。这些算法在融合时间因素后，在一定程度上提高了推荐的质量。

2.2 传统的协同过滤算法

传统协同过滤算法分为基于用户的协同过滤、基于物品(项目)的协同过滤和基于模型的协同过滤[22]。目前基于物品的协同过滤算法是商业界应用得最多的算法，Amazon，YouTube，Hulu等网站的推荐算法都是基于该算法提出的。下面详细介绍基于物品的协同过滤算法。

基于物品的协同过滤算法基于一个假设：相似的物品得到的用户评分是相似的[23]。其基本思想是：根据用户的历史评分或偏好，计算出物品间的相似度，通过相似物品预测用户对候选物品的偏好或预测用户对候选物品的评分。预测用户对候选物品的偏好并给用户提供个性化的推荐列表称为top-N推荐，预测用户对物品的评分称为评分预测。在实际应用中，top-N推荐更接近系统的实际需求。

基于物品的协同过滤算法进行top-N推荐的基本流程如图1所示。

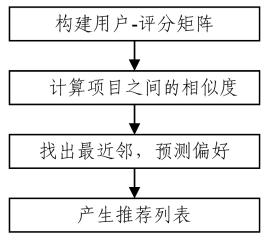

图1 基于物品的协同过滤算法的基本流程

Fig. 1 Basic procedure of collaborative filtering algorithm based on item

步骤1 构建用户-物品评分矩阵。矩阵元素的值可以是具体的评分值，也可以是0或1。矩阵元素的值为具体评分值时代表用户对该物品的实际评分；矩阵元素的值为1或0时代表用户是否对该物品有历史行为，如是否购买过某商品、是否看过某部电影等。

步骤2 计算物品间的相似度。Badrul等[24]提出物品之间的相似度度量方法有余弦相似度、皮尔逊相似度和修正余弦相似度，具体公式如下：

$$sim_{i,j}^{cos}=\frac{\sum_{u\in U_{i,j}} r_{u,i}r_{u,j}}{\sqrt{\sum_{u\in U_{i,j}}(r_{u,i})^2}\sqrt{\sum_{u\in U_{i,j}}(r_{u,j})^2}} \tag{1}$$

$$sim_{i,j}^{pearson}=\frac{\sum_{u\in U_{i,j}}(r_{u,i}-\overline{r_i})(r_{u,j}-\overline{r_j})}{\sqrt{\sum_{u\in U_{i,j}}(r_{u,i}-\overline{r_i})^2}\sqrt{\sum_{u\in U_{i,j}}(r_{u,j}-\overline{r_j})^2}} \tag{2}$$

$$sim_{i,j}^{adcos}=\frac{\sum_{u\in U_{i,j}}(r_{u,i}-\overline{r_u})(r_{u,j}-\overline{r_u})}{\sqrt{\sum_{u\in U_{i,j}}(r_{u,i}-\overline{r_u})^2}\sqrt{\sum_{u\in U_{i,j}}(r_{u,j}-\overline{r_u})^2}} \tag{3}$$

在上述公式中，u表示用户，i和j表示物品，$r_{u,i}$和$r_{u,j}$分别表示用户u对物品i和j的评分，$U_{i,j}$表示同时给物品i和j评分的用户集合，$\overline{r_i}$和$\overline{r_j}$分别表示物品i和j的平均评分，$\overline{r_u}$表示用户u的平均评分。

步骤3 找出物品的最近邻，预测用户偏好。遍历用户所有的历史物品，找出与每个历史物品最相似的k个物品作为候选集，预测用户对候选物品(排除用户历史物品)的偏好。

步骤4 生成推荐列表。按用户偏好得分对候选集进行排序，获取用户对物品偏好最高的n个物品，并将其推荐给用户。

偏好预测不同于评分预测，偏好预测的目的是计算用户对候选物品的偏好程度，其并不代表用户对该物品的预测评分。用户对物品的偏好预测可以通过式(4)计算：

$$p_{u,j}=\sum_{i\in N_u \text{ and } j\in S_{i,k}} sim_{j,i}\cdot r_{ui} \tag{4}$$

其中，$p_{u,j}$表示用户 u 对物品 j 的偏好，N_u 表示用户 u 的历史物品集合，$S_{i,k}$表示与物品 i 相似的 k 个物品集合，$sim_{j,i}$表示物品 i 和 j 的相似度，r_{ui}表示用户 u 对物品 i 的评分。

3　改进的协同过滤算法

3.1　评分偏好模型

现实生活中，用户对物品的打分习惯存在差异[25]，有的用户对物品打分比较严格，习惯于打低分，有的用户对物品打分相对宽松，习惯于打高分。如果这两类用户对某一物品的打分相同，那么打分相对严格的用户可能更加偏爱这一物品。

考虑到用户评分习惯的影响，即用户评分的分布情况，引入用户评分偏好模型。对于离散型评分数据，假设评分类别集合为$\{C_1,C_2,\cdots,C_k\}$，若 $C_i>C_j$，则表示评分类别 C_i 大于评分类别 C_j。例如，$\{C_1,C_2,C_3,C_4,C_5\}$表示评分类别$\{1,2,3,4,5\}$，则评分类别 $C_2>C_1$。

用户 u 对评分类别 C_i 的偏好得分可以通过式(5)计算：

$$pref(u,C_i)=\partial\cdot\sum_{C_j\in\{C_1,\cdots,C_k\}}\frac{pref>(C_i,C_j)}{|R_u|}+\beta\cdot\frac{pref=(C_i)}{|R_u|} \tag{5}$$

其中，$pref(u,C_i)$表示用户 u 对评分类别 C_i 的偏好得分；$pref>(C_i,C_j)$表示用户 u 评分为 C_j 的数量，其中 $C_j<C_i$；$|R_u|$表示用户 u 的评分数量；$pref=(C_i)$表示用户 u 评分为 C_i 的数量。根据文献[17,26]，∂ 取值为 1，β取值为 0.5。

计算出用户评分类别偏好矩阵后，使用用户评分类别偏好得分代替用户-物品矩阵中的原始得分，得到修正的用户-物品评分矩阵，该矩阵元素的值并不代表用户对物品的实际评分，仅代表用户对物品的偏好。

下面举例说明修正的用户-物品评分矩阵的计算过程。假设原始用户-物品评分矩阵如表 1 所列。其中共有 2 个用户(u 和 v)和 6 个物品($I_1,I_2,\cdots,I_6$)，用户评分范围为$\{1,2,3,4,5\}$。由式(5)计算得到用户评分类别偏好矩阵，如表 2 所列。使用用户对评分类别的偏好得分代替用户原始评分，得到修正的用户-物品评分矩阵，如表 3 所列。

表 1　原始用户-物品评分矩阵

Table 1　Original user-item rating matrix

	I_1	I_2	I_3	I_4	I_5	I_6
u	1	1	2	4	3	5
v	2	1	3	4	5	5

表 2　用户评分类别偏好矩阵

Table 2　User-rating class preference matrix

	1	2	3	4	5
u	0.167	0.417	0.583	0.750	0.917
v	0.083	0.250	0.417	0.583	0.833

表 3　修正的用户-物品评分矩阵

Table 3　Adjusted user-item rating matrix

	I_1	I_2	I_3	I_4	I_5	I_6
u	0.167	0.167	0.417	0.750	0.583	0.917
v	0.250	0.083	0.417	0.583	0.833	0.833

可以看出，用户 u 和 v 虽然对物品 I_6 的评分都为 5，但根据修正的用户-物品评分矩阵可以认为用户 u 较用户 v 更偏爱物品 I_6。

3.2　时间权重因子

用户兴趣的阶段性使用户在短期内交互的物品具有更高的相关性。时间信息对于推荐算法有重要影响，主要体现在两个方面：用户在短期内感兴趣的物品具有更高的相似度；用户最近的行为更能反映用户当前的兴趣。本文从第一个方面出发，引入时间权重因子，其定义为[27]：

$$t_{i,j}=e^{-|t_{u,i}-t_{u,j}|} \tag{6}$$

其中，$t_{i,j}$为时间权重因子，$t_{u,i}$和 $t_{u,j}$分别表示用户 u 对物品 i 和 j 的评分时间。可以看出，用户对两个物品评分的时间间隔越短，时间权重就越大。

根据修正的用户-物品评分矩阵，结合时间权重因子，得到基于用户评分的物品相似度计算公式[28]：

$$r\_sim_{i,j}=\frac{\sum_{u\in U_{i,j}} t_{i,j}(r_{u,i}-\overline{r_i})(r_{u,j}-\overline{r_j})}{\sqrt{\sum_{u\in U_i}(r_{u,i}-\overline{r_i})^2}\sqrt{\sum_{u\in U_j}(r_{u,j}-\overline{r_j})^2}} \tag{7}$$

其中，$r_sim_{i,j}$表示物品 i 和 j 的评分相似度，$t_{i,j}$为时间权重因子，$r_{u,i}$和 $r_{u,j}$分别表示用户 u 对物品 i 和 j 偏好评分，$\overline{r_i}$和$\overline{r_j}$分别表示物品 i 和 j 的偏好平均评分，U_i 和 U_j 分别表示对物品 i 和 j 评分的用户集合，$U_{i,j}$表示同时给物品 i 和 j 评分的用户集合。

3.3　物品属性相似度

文献[11]指出综合考虑物品评分数据和属性特征能够在一定程度上缓解数据稀疏性导致的相似度度量不准确的问题。在用户评分数据较少的情况下，可以利用物品属性构造物品属性相似度，通过物品属性相似度和评分相似度的加权，减小仅有评分相似度带来的误差。标签属性是常用的物品属性，下面以标签属性为例进行介绍。

假设物品标签集合为 $T=\{t_1,t_2,t_3,\cdots,t_k\}$，$k$ 为标签总个数。物品标签属性可以用一个 $n\times k$ 的矩阵来表示，n 为物品个数，k 为标签个数。矩阵元素的值为 0 或 1，1 代表该物品具有这个标签，0 代表该物品不具有此标签。则物品属性相似度的计算公式为：

$$p\_sim_{i,j}=\frac{|N_i\cap N_j|}{\sqrt{|N_i||N_j|}} \tag{8}$$

其中，$|N_i|$和 $|N_j|$分别表示物品 i 和物品 j 具有的标签数，$|N_i\cap N_j|$表示物品 i 和物品 j 都具有的标签数。

综上所述，得到最终的物品相似度计算公式为：

$$sim_{i,j}=\delta\cdot p\_sim_{i,j}+(1-\delta)\cdot r\_sim_{i,j} \tag{9}$$

该公式表示最终物品相似度是评分相似度和物品属性相似度的加权平均值。其中，δ 为平衡因子，取值为$[0,1]$，通过实验选择最优值。

标签属性只是物品属性的一种，针对不同的数据集需要选择合适的物品属性进行物品属性相似度的计算。

3.4　相似度的归一化

文献[29]提出在基于物品的协同过滤算法中，将物品相似度矩阵按最大值归一化，可以提高推荐的准确率。文献[18]通过实验证明，相似度矩阵的归一化不仅能提高推荐的

准确度，还可以提高推荐的覆盖率和多样性。在得到物品相似度矩阵后，可以按式(10)进行归一化：

$$sim'_{ij}=\frac{sim_{ij}}{max} \tag{10}$$

其中，sim_{ij} 表示物品 i 和 j 的相似度，max 表示相似度矩阵的最大值。

3.5 预测偏好得分

根据上文的分析，评分相似度的计算是以修正的用户-物品评分矩阵为基础的，因此预测用户偏好应该基于用户对物品的偏好得分，即修正的用户-物品评分矩阵中元素的值，预测用户对物品的偏好公式如式(11)所示：

$$p_{u,j}=\sum_{i\in N_u \text{ and } j\in S_{i,k}} sim_{j,i}\cdot r_{ui} \tag{11}$$

其中，$r_{u,i}$ 为用户 u 对物品 i 的偏好评分，其余参数同式(4)。

3.6 算法描述

基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法如算法 1 所示。

算法 1 基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法

输入：用户-物品评分矩阵 **R**(m,n)，物品属性矩阵 **P**(n,k)，物品近邻个数 K，推荐列表长度 N

输出：各个用户的推荐列表

步骤 1 根据用户-物品评分矩阵 **R**(m,n)和式(5)计算用户评分类别偏好矩阵。

步骤 2 用步骤 1 得到的用户对评分类别的偏好得分来代替原始用户评分，从而得到修正的用户-物品评分矩阵。

步骤 3 根据式(7)计算融合时间权重因子的用户评分相似度矩阵 **R_Sim**(n,n)，然后归一化 **R_Sim**(n,n)。

步骤 4 根据式(8)计算物品属性相似度矩阵 **P_Sim**(n,n)，然后归一化 **P_Sim**(n,n)。

步骤 5 根据式(9)计算最终物品的相似度矩阵 **Sim**(n,n)，然后归一化 **Sim**(n,n)。

步骤 6 遍历用户的历史物品集合，找出每个历史物品最相似的 K 个物品(去掉历史物品)作为候选集，并根据式(11)计算用户对候选集中物品的预测偏好得分。

步骤 7 从候选集中选出得分最高的 N 个物品作为该用户的推荐物品集。

步骤 8 重复步骤 6 和步骤 7，生成所有用户的推荐物品集。

4 实验结果与分析

4.1 数据集

本文实验采用 MovieLens-100K 和 MovieLens-Latest-Small 这两个真实数据集。MovieLens 数据集是一个关于电影评分的数据集，由明尼苏达大学 GroupLens 研究小组提供，包含用户对电影的评分信息、用户信息、电影信息和标签信息等数据。MovieLens 数据集有多个版本，本文采用的是两个数据量较小的版本：1) MovieLens-100K 数据集，包含 943 个用户对 1 652 部电影的 100 000 个评分，稀疏度为 93.7%；2) MovieLens-Latest-Small 数据集，包含 610 个用户对 9724 部电影的 100837 个评分，稀疏度为 98.3%。下面分别将 MovieLens-100K 数据集和 MovieLens-Latest-Small 数据集记为 M_1 和 M_2。

实验采用随机划分的方式将 80% 的数据作为训练集，20% 的数据作为测试集。实验中取 5 次实验的算术平均值作为最终结果。

4.2 评价指标

评价推荐算法性能的指标有多个，包括准确度、新颖性、覆盖率和多样性等，其中预测准确度是最重要的指标。top-N 推荐的准确度指标为准确率(Precision)和召回率(Recall)，本文采用这两个指标作为评价指标。

准确率和召回率的定义如下：

$$precision=\frac{\sum_{u\in U}|R(u)\cap T(u)|}{\sum_{u\in U}|R(u)|} \tag{12}$$

召回率的定义为：

$$recall=\frac{\sum_{u\in U}|R(u)\cap T(u)|}{\sum_{u\in U}|T(u)|} \tag{13}$$

在式(12)和式(13)中，U 表示测试集中的所有用户集合，$R(u)$ 为根据训练集为用户推荐的列表，$T(u)$ 为用户在测试集上的行为列表。

4.3 结果分析

为了验证本文提出的基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法(PTP-Item-CF)的性能，将该算法与传统基于物品的协同过滤算法(IBCF)、文献[7]提出的基于用户兴趣和物品属性的协同过滤算法(UIIP-CF)和文献[6]提出的使用偏好模型的协同过滤算法(PreItemCF)在数据集 M_1 和 M_2 上进行比较。

1)平衡因子 δ 对 PTP-Item-CF 算法的影响

实验中，推荐列表长度为 10，物品最近邻数为 20，通过设置不同的 δ 值来测试其对算法准确度的影响。首先将 δ 值从 0.1 依次递增到 0.9，可以发现 δ 大于 0.2 时，算法的准确度会迅速减小。因此，将 δ 从 0 按更小的步长依次递增到 0.2 进行实验，寻找最优的 δ 值。图 2 和图 3 分别给出了 δ 在 M_1 和 M_2 数据集上的结果。

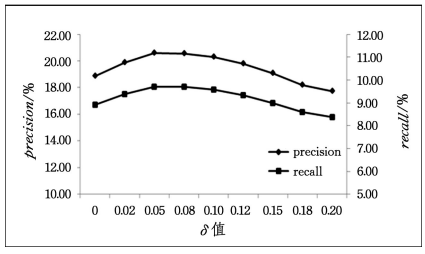

图 2 M_1 数据集上平衡因子 δ 对算法的影响

Fig. 2 Influence of δ on proposed algorithm on M_1 dataset

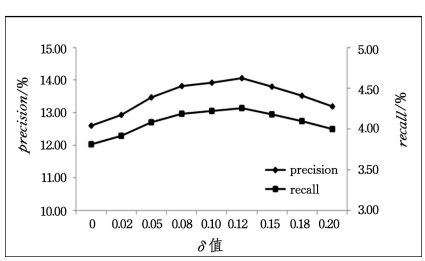

图 3 M_2 数据集上平衡因子 δ 对算法的影响

Fig. 3 Influence of δ on proposed algorithm on M_2 dataset

从图 2 和图 3 可以看出，PTP-Item-CF 算法的准确率和召回率随 δ 值的增大先增大后减小。在 M_1 数据集上，δ 取值为 0.05 左右时，算法的准确率和召回率达到最高，比 δ 取 0 时提高了 9.02%。在 M_2 数据集上，δ 取值为 0.12 左右时，算法的准确率和召回率达到最高，比 δ 取 0 时提高了11.54%。由此可以说明，使用本文提出的相似度比只使用偏好模型和时间权重因子得到的评分相似度具有更好的推荐效果。

另外，从实验结果可以看出，在 M_1 和 M_2 数据集上 δ 的最优值都较小，即物品属性相似度所占的比重较小。这是由于将电影主题标签作为物品属性较为粗糙，主题标签只有 19 种，计算得到的物品属性相似度集中在一些固定的值上，这些值比大多数的评分相似度大得多。因此，在物品属性相似度占比较小时，平衡因子才能起到更好的调节作用。

2）不同物品近邻数下各算法的性能对比

实验中，推荐列表长度为 10，M_1 数据集上平衡因子值为 0.05，M_2 数据集上平衡因子值为 0.12，通过设置不同的物品近邻数 k 来测试其对算法准确度的影响，物品近邻数 k 从 20 依次递增到 220。图 4－图 7 展示了在 M_1 和 M_2 数据集上各算法在不同物品近邻数下的实验结果。

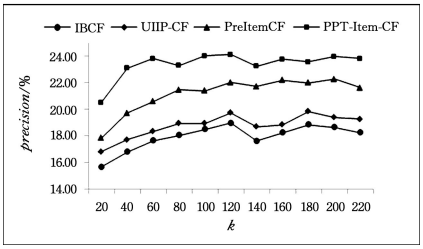

图 4　M_1 数据集上不同近邻数下各算法准确率的对比

Fig. 4　Precision contrast of each algorithm with different nearest neighbors on M_1 dataset

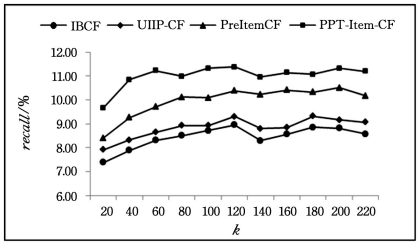

图 5　M_1 数据集上不同近邻数下各算法召回率的对比

Fig. 5　Recall contrast of each algorithm with different nearest neighborson M_1 dataset

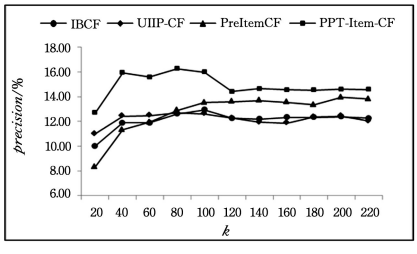

图 6　M_2 数据集上不同近邻数下各算法准确率的对比

Fig. 6　Precision contrast of each algorithm with different nearest neighbors on M_2 dataset

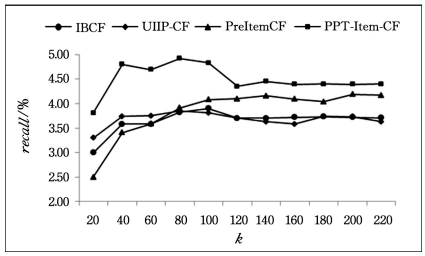

图 7　M_2 数据集上不同近邻数下各算法召回率对比

Fig. 7　Recall contrast of each algorithm with different nearest neighbors on M_2 dataset

从图 4 和图 5 可以看出，在 M_1 数据集上，随着物品近邻数 k 的增大，各算法的准确率和召回率先迅速增大后趋于平缓，当物品近邻数 k 取值为 120 左右时，各算法具有最高的准确率和召回率。从图 6 和图 7 可以看出，在 M_2 数据集上，随着物品近邻数 k 的增大，PTP-Item-CF 算法的准确率和召回率先迅速增大后趋于平缓，然后略有下降，最后保持稳定。PreItemCF 算法在近邻物品数 k 较小时，具有最低的准确率和召回率，但随着 k 的增大，准确率和召回率不断增大，直到 k 大于 100 时，准确率和召回率保持稳定，且高于 IBCF 和 UIIP-CF 算法。IBCF 和 UIIP-CF 算法的准确率和召回率随近邻物品数 k 的增大先迅速增大后趋于平稳，且两者稳定的数值基本相同。当物品近邻数 k 相同时，无论是在 M_1 数据集还是在 M_2 数据集上，本文提出的 PTP-Item-CF 算法相对于其他算法都具有最高的准确率和召回率。

3）不同推荐列表长度下各算法的性能对比

实验中，物品近邻数为 20，M_1 数据集上平衡因子值为 0.05，M_2 数据集上平衡因子值为 0.12，通过设置不同的推荐列表长度 n 来测试其对算法准确度的影响，推荐列表长度 n 从 5 依次递增到 30。图 8－图 11 展示了在 M_1 和 M_2 数据集上各算法在不同推荐列表长度下的实验结果。

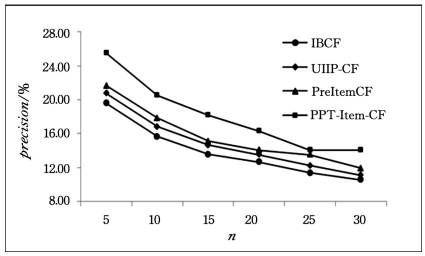

图 8　M_1 数据集上不同推荐长度下各算法准确率的对比

Fig. 8　Precision contrast of each algorithm with different recommend lengths on M_1 dataset

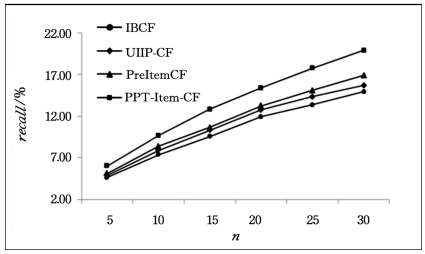

图 9　M_1 数据集上不同推荐长度下各算法召回率的对比

Fig. 9　Recall contrast of each algorithm with different recommend lengths on M_1 dataset

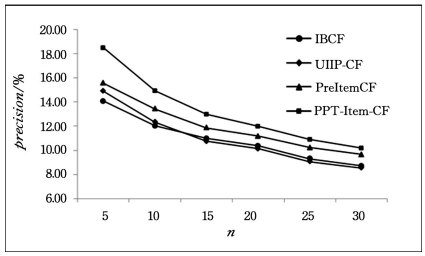

图 10 M_2 数据集上不同推荐长度下各算法准确率的对比

Fig. 10 Precision contrast of each algorithm with different recommend lengths on M_2 dataset

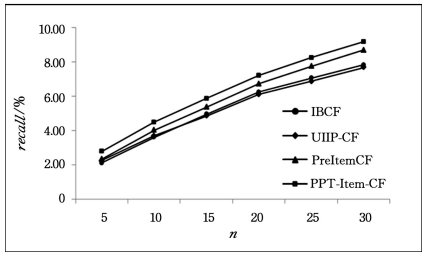

图 11 M_2 数据集上不同推荐长度下各算法召回率的对比

Fig. 11 Recall contrast of each algorithm with different recommend lengths on M_2 dataset

从图 8—图 11 中可以看出，无论是在 M_1 数据集上还是在 M_2 数据集上，随着推荐列表长度 n 的逐渐增大，各算法的准确率不断降低，召回率不断增大。当推荐列表长度一定时，本文提出的 PTP-Item-CF 算法相比其他算法具有最高的准确率和召回率。

表 4 列出了在 top-10 推荐下，各算法在 M_1 和 M_2 数据集上准确率、召回率的最优值。

表 4 各算法在不同数据集上准确率、召回率的最优值

Table 4 Optimal precision and recall of each algorithm on different datasets

（单位：%）

算法	M_1		M_2	
	准确率	召回率	准确率	召回率
IBCF	18.97	8.95	12.93	3.90
UIIP-CF	19.73	9.31	12.70	3.85
PreItemCF	22.00	10.38	13.95	4.19
PTP-Item-CF	24.11	11.37	16.26	4.92

从表 4 可以看出，本文提出的 PTP-Item-CF 算法在 M_1 数据集上的准确率和召回率相比 IBCF 算法提高了 27.10%，相比 UIIP-CF 算法提高了 22.20%，相比 PreItemCF 算法提高了 9.60%；在 M_2 数据集上相比 IBCF 算法提高了 25.75%，相比 UIIP-CF 算法提高了 28.03%，相比 PreItemCF 算法提高了 16.56%。

实验结果表明，本文提出的基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法（PTP-Item-CF）能有效提高推荐的准确度，缓解数据稀疏性问题，从而提高推荐质量。

结束语 本文提出了一种基于用户评分偏好模型、融合时间因素和物品属性的协同过滤算法，从用户评分特性、时间因素和物品属性 3 个方面来改进相似度度量。实验结果表明，所提算法能够提高推荐的准确率和召回率，缓解稀疏性问题。但是，该算法还未充分挖掘用户-物品评分矩阵的潜在信息，如未考虑不同物品之间的影响度，下一步工作将考虑物品之间的非对称影响度，将其融入到相似度计算中，以进一步提高推荐的质量。另外，将考虑把物品属性相似度替换成用户属性相似度，将算法思想应用到基于用户的协同过滤算法中。

参 考 文 献

[1] ISINKAYE F O, FOLAJIMI Y O, OJOKOH B A. Recommendation systems: Principles, methods and evaluation[J]. Egyptian Informatics Journal, 2015, 16(3): 261-273.

[2] MOONEY R J, ROY L. Content-based book recommending using learning for text categorization[C]// Proc of the 5th ACM Conference on Digital libraries. New York: ACM Press, 2000: 195-204.

[3] PHELAN O, MCCARTHY K, BENNETT M, et al. Terms of a feather: Content-based news recommendation and discovery using twitter[C]// Proc of the 33rd European Conference on IR Research. Berlin: Springer, 2011: 448-459.

[4] ZHU J, HAN L X, GOU Z N, et al. A fuzzy clustering - based denoising model for evaluating uncertainty in collaborative filtering recommender systems[J]. Journal of the Association for Information Science and Technology, 2018, 69(9): 1109-1121.

[5] SU X, KHOSHGOFTAAR T M. A survey of collaborative filtering techniques[J]. Advances in Artificial Intelligence, 2009, 2009(4): 1-19.

[6] ZHENG L S, YANG S Q, HE J, et al. An optimized collaborative filtering recommendation algorithm[C]// Proc of the 2nd International Conference on Cloud Computing and Internet of Things. Piscataway, NJ: IEEE. 2016: 89-92.

[7] LINDEN G, SMITH B, YORK J. Amazon.com recommendations: item-to-item collaborative filtering[J]. IEEE Internet Computing, 2003, 7(1): 76-80.

[8] HU J. Application and research of collaborative filtering in e-commerce recommendation system[C]// Proc of the 3rd IEEE International Conference on Computer Science and Information Technology. Piscataway, NJ: IEEE Computer Society, 2010: 686-689.

[9] SILVA E Q D, CAMILO-JUNIOR C G, PA-SCOAL L M L, et al. An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering[J]. Expert Systems with Applications, 2016, 53: 204-218.

[10] ZIEGLER C N, LAUSEN G, SCHMIDT-TH-IEME L. Taxonomy-driven computation of product recommendations[C]// Proc of the 13th ACM Conference on Information and Knowledge Management. New York: ACM Press, 2004: 406-415.

[11] WENG X L, WANG Z J. Research process of collaborative filtering recommendation algorithm[J]. Computer Engineering and Applications, 2018, 54(1): 25-31.

[12] DONG Y, ZHAO C, CHENG W, et al. A personalized recommendation algorithm with user trust in social network[C]// International Conference of Young Computer Scientists, Engineers and Educators. Singapore: Springer, 2016: 63-76.

[13] ADOMAVICIUS G，TUZHILIN A. Toward the next generation of recommender systems：A survey of the state-of-the-art and possible extensions[J]. IEEE Transactions on Knowledge and Data Engineering，2005，17(6)：734-749.

[14] CHEN H，LI Z K，HU W. An improved collaborative recommendation algorithm based on optimized user similarity[J]. The Journal of Supercomputing，2016，72(7)：2565-2578.

[15] PARIVASH P，HWANG D，JUNG J E. Weighted similarity schemes for high scalability in user-based collaborative filtering [J]. Mobile Networks and Applications，2015，20(4)：497-507.

[16] LIU J，WU W Q，LI X，et al. Collaborative filtering algorithm based on user interest and item properties[J]. Computer Applications and Software，2017，34(5)：33-37.

[17] LEE J，LEE D，LEE Y C，et al. Improving the accuracy of top-N recommendation using a preference model[J]. Information Sciences，2016，348(c)：290-304.

[18] XIANG L. Recommender Systems Practice[M]. Beijing：Post & Telecom Press，2012：122-123.

[19] GASMI I，SERIDI-BOUCHELAGHEM H，HOCINE L，et al. Collaborative filtering recommendation based on dynamic changes of user interest[J]. Intelligent Decision Technologies，2015，9(3)：271-281.

[20] WU F，YU L S，FENG M. A collaborative filtering algorithm based on time effect [J]. Computer Engineering & Science，2017，39(11)：2095-2101.

[21] JIN X，ZHENG Q，SUN L. An optimization of collaborative filtering personalized recommendation algorithm based on time context information[C]// 16th International Conference on Informatics and Semiotics in Organisations (ICISO). Singapore：Springer，2015：146-155.

[22] BREESE J E，HECKERMAN D，KADIE C. Empirical analysis of algorithms for collaborative filtering[C]// Proc of the 14th conference on Uncertainty in artificial intelligence. San Francisco：Morgan Kaufmann Publishes Inc，1998：43-52.

[23] LIU J，YONG W，YAN F. An improved collaborative filtering recommendation algorithm[J]. Computer Engineering & Applications，2016，32(9)：3019-3018.

[24] BADRUL S，GEORGE K，JOSEPH K，et al. Item-based collaborative filtering recommendation algorithms[C]// Proc of the 10th International World Wide Web Conference. New York：ACM Press，2001：285-295.

[25] ZENG A，GAO S C，XU X Q. Collaborative filtering algorithm incorporating time factor and user preference properties[J]. Computer Science，2017，44(9)：243-249.

[26] JIN R，SI L，ZHAI C X，et a1. Collaborative filtering with decoupled models for preferences and ratings[C]// Proc of the 12th ACM International Conference on Information and Knowledge Management. New York：ACM Press，2003：309-316.

[27] XIAO W Q，YAO S J，WU S M. Improved top-N collaborative filtering recommendation algorithm[J]. Application Research of Computers，2018，35(1)：105-108，112.

[28] WEI T T，CHEN L，FAN T T，et al. Collaborative filtering recommendation algorithm based on item popularity weighting[J/OL]. Application Research of Computers. [2019-01-23]. http://kns. cnki. net/kcms/detail/51. 1196. TP. 20181224. 1632. 005. html.

[29] KARYPIS G. Evaluation of item-based top-N recommendation algorithms[C]// Proc of the 10th International Conference on Information and Knowledge Management. New York：ACM Press，2001：247-254.

ZHU Lei，born in 1993，postgraduate. His main research interests include recommender systems and intelligent information processing technology.

HU Qin-han，born in 1987，master. His main research interests include machine learning and intelligent information processing technology.