

基于正相关和负相关最近邻居的协同过滤算法

徐 怡^{1,2}, 唐一民¹, 王 冉¹

(1.安徽大学 计算机科学与技术学院, 安徽 合肥 230601; 2.计算智能与信号处理教育部重点实验室(安徽大学), 安徽 合肥 230039)

摘 要:协同过滤算法是应用最广泛和最成功的推荐算法之一。针对传统协同过滤算法在预测评分时仅考虑正相关最近邻居, 没有考虑负相关最近邻居对评分预测的影响而导致的预测结果准确性及多样性较低的问题, 提出一种基于正相关和负相关最近邻居的协同过滤算法。该算法首先计算用户之间的相似度, 再通过用户评分与其平均评分等信息计算出用户之间的变异系数, 利用变异系数修正相似度的值, 从而缓解因为用户共同项目数不足而导致的相似度计算结果可信度较低的问题。然后分别对与目标用户相似度为正及与目标用户相似度为负的用户进行排序, 并利用动态加权参数 α 及训练得到的阈值 β 分别选取正相关最近邻居和负相关最近邻居, 基于选取的正相关最近邻居和负相关最近邻居分别进行预测评分。最后, 将基于正相关最近邻居和负相关最近邻居的预测评分进行加权, 作为最终的预测评分。在MovieLens数据集上利用3种评价标准进行对比实验, 结果表明本文算法有效地提高了推荐的准确性和多样性。

关键词:协同过滤; 正相关邻居; 负相关邻居

中图分类号: TP391

文献标志码: A

文章编号: 2096-3246(2018)05-0189-07

Collaborative Filtering Algorithm Based on Positive Correlation and Negative Correlation Nearest Neighbors

XU Yi^{1,2}, TANG Yimin¹, WANG Ran¹

(1.School of Computer Sci. and Technol., Anhui Univ., Hefei 230601, China;

2.Key Lab. of Intelligent Computing & Signal Processing(Anhui Univ.), Ministry of Education, Hefei 230039, China)

Abstract: Collaborative filtering algorithm is one of the most widely and successfully used recommendation algorithm. Traditional collaborative filtering algorithm only considers the positive correlation nearest neighbors while ignore the influence of negative correlation nearest neighbors when predicting item rating, which has caused problems like low accuracy and low diversity. In order to resolve these problems, based on positive correlation and negative correlation neighbors a collaborative filtering algorithm is proposed. Firstly, the algorithm computes the similarities and Coefficient of Variation between users and sorts the neighbors according to the similarities which were modified by the Coefficient of Variation. Secondly, the positive nearest neighbors and the negative nearest neighbors are selected based on dynamic weight value α and threshold value β , then item ratings are predicted based on the positive nearest neighbors and the negative nearest neighbors respectively. Finally, the last prediction result is acquired by combining predicting ratings based on the positive nearest neighbors and the negative nearest neighbors with weight. The experiments are tested on the MovieLens with three metrics, and the result show that this approach is achieved great improvement of the accuracy and diversity of recommendation effectively.

Key words: collaborative filtering; positive correlation nearest neighbors; negative correlation nearest neighbors

随着互联网的不断飞速发展, 数据指数级的增长带来了信息过载等一系列问题, 于是推荐系统由此诞生并得到飞速发展。如国外的亚马逊、ebay等,

国内的京东、阿里巴巴等都在推荐系统上做出了各种尝试^[1-2]。其广泛应用于电子商务、音乐、视频及新闻等各种领域。目前主要的推荐算法有基于协同过

收稿日期: 2017-09-17

基金项目: 国家自然科学基金资助项目(61402005); 安徽省自然科学基金资助项目(1308085QF114); 安徽省高等学校省级自然科学基金资助项目(KJ2013A015); 安徽大学计算智能与信号处理教育部重点实验室课题项目资助; 国家留学基金委员会资助项目

作者简介: 徐 怡(1981—), 女, 副教授, 博士。研究方向: 智能信息处理; 粗糙集理论; 个性化推荐。E-mail: xuyi1023@126.com

网络出版时间: 2018-08-30 15:40:00 网络出版地址: <http://kns.cnki.net/kcms/detail/51.1773.TB.20180830.1539.019.html>

滤的推荐、基于关联规则的推荐、基于内容的推荐、基于图的推荐及混合推荐等。其中协同过滤算法是最经典且最成功的推荐算法之一^[3-5],该算法主要分为基于用户的协同过滤算法和基于项目的协同过滤算法。基于用户的协同过滤算法的主要思想为参照与目标用户相似的其他用户,计算相似度并找出最近邻居,最后根据最近邻居的评分预测出目标用户对目标项目的评分;基于项目的协同过滤算法则是从项目角度出发,找出与其最相似的项目集合,最后预测评分^[6-7]。传统的协同过滤算法的优点是无需考虑内容信息和一些复杂的概念,缺点是推荐精度不高,以及存在稀疏性、扩展性、冷启动等一些问题^[8-9]。

针对推荐精度不高的问题,研究人员提出了很多改进方法。文献[10]提出一种基于时间段划分的协同过滤算法,传统的协同过滤算法认为用户的兴趣是静态的,没有考虑用户兴趣的转变,且用户有长时间的兴趣和短时间的兴趣。对此,本文引入了时间因子,将用户评分历史划分为几个周期,分析在这些周期中用户兴趣的分布,并量化用户兴趣,通过设置时间窗口发现用户最近的兴趣。

文献[11]提出一种不确定近邻的协同过滤推荐算法,该算法自适应地选择目标用户的最近邻居作为推荐群,提高了推荐质量。文献[12]提出一种基于时间上下文的协同过滤算法,考虑到用户的兴趣会发生改变,故加入时间上下文,在不需要额外信息的前提下,进一步缓解稀疏性。除此之外,二部图算法也得到了广泛的应用^[13-14],且与协同过滤相结合提高推荐精度。文献[15]提出一种结合二部图投影与排序的协同过滤,该算法将用户和项目看成节点,用户对项目的评分则可用边权表示,利用二部图投影和随机游走对节点进行排序,最终形成推荐。

上述方法虽然在一定程度上提高了推荐质量,但还是具有不足的地方。比如,在相似度计算时只依赖共同评分项目,但当共同评分项目个数较少时,计算出来的相似度可信度会非常低。另外,在预测评分时也均只考虑了正相关最近邻居,忽略了负相关最近邻居对预测评分的价值,若两个用户之间的负相关程度较高,说明这两个用户的评分呈现较高的相反趋势,可从另一个角度对评分进行预测。

基于此,为进一步提高推荐精度,提出一种基于正相关和负相关最近邻居的协同过滤算法(PNCF)。在相似度计算时,通过用户之间变异系数的差异反映出用户评分特点的接近程度,从而对相似度计算结果进行修正,弥补了只依赖共同评分项目计算相似度带来的不足;在邻居选取时根据一定的选取规则分别选取正相关最近邻居和负相关最近邻居,并

在评分预测的同时考虑这两类不同的最近邻居,将基于正相关最近邻居和负相关最近邻居的预测评分进行加权,作为最终的预测评分。利用GroupLens提供的标准数据集MovieLens^[16]进行算法验证,实验表明本文算法有效地提高了推荐的准确度和多样性。

1 传统的协同过滤算法

1.1 构建评分矩阵

协同过滤算法所需要的输入数据 $I = (U, P, R)$,其中: $U = \{u_1, u_2, u_3, \dots, u_n\}$ 为用户id构成的集合, $|U| = n$; $P = \{p_1, p_2, p_3, \dots, p_m\}$, P 为项目id构成的集合, $|P| = m$; R 为 U 和 P 构成的 $n \times m$ 的矩阵,若评分为显式的,则 R 存储的为评分值的大小, R_{ij} 表示 i 用户对 j 项目的评分值,其值的大小代表评分尺度^[17];若评分为隐式的,则 R 取值为0或1,0表示用户没有对项目产生过行为(购买、点击等),1表示产生过行为。

1.2 相似度计算

相似度的计算是协同过滤算法中非常重要的一部分,利用第1.1节中构建的用户-项目评分矩阵可计算用户之间的相似度,常用的计算相似度的方法有Pearson相关系数和修正的余弦相似度。Pearson相关系数的公式为:

$$\text{sim}(i, j) = \frac{\sum_{p \in P_{ij}} (r_{ip} - \bar{r}_i)(r_{jp} - \bar{r}_j)}{\sqrt{\sum_{p \in P_{ij}} (r_{ip} - \bar{r}_i)^2} \sqrt{\sum_{p \in P_{ij}} (r_{jp} - \bar{r}_j)^2}} \quad (1)$$

式中, i, j 为用户, P_{ij} 为 i 用户和 j 用户共同评分的项目集合, r_{ip} 和 r_{jp} 分别为 i 用户对 p 项目的评分和 j 用户对 p 项目的评分, \bar{r}_i 和 \bar{r}_j 分别为 i 用户对项目的平均评分和 j 用户对项目的评分。Pearson相关系数取值为 $[-1, 1]$ 。

修正的余弦相似度^[19]公式为:

$$\text{sim}(i, j) = \frac{\sum_{p \in P_{ij}} (r_{ip} - \bar{r}_i)(r_{jp} - \bar{r}_j)}{\sqrt{\sum_{p \in P_i} (r_{ip} - \bar{r}_i)^2} \sqrt{\sum_{p \in P_j} (r_{jp} - \bar{r}_j)^2}} \quad (2)$$

式中, i, j 为用户, P_{ij} 为 i, j 共同评分的项目集合, P_i 和 P_j 分别为 i 用户已评分的项目集合和 j 用户已评分的项目集合, r_{ip} 和 r_{jp} 分别为 i 用户对 p 项目的评分和 j 用户对 p 项目的评分, \bar{r}_i 和 \bar{r}_j 分别为 i 用户和 j 用户对项目的平均评分。修正的余弦相似度取值也为 $[-1, 1]$ 。

1.3 评分预测

计算出目标用户与其他用户的相似度后,按相

似度值从大到小进行排序得到邻居集合 Q ,根据相似度阈值或最近邻居个数选取集合 Q 中的一部分邻居作为目标用户的最近邻居集合 N ,其中 $N = \{n_1, n_2, n_3, \dots, n_k\}$ 。基于得到的邻居集合对目标用户未评分的项目进行评分预测,按照预测的评分大小进行排序,并取top N 推荐给用户。其中评分预测公式^[20]为:

$$pred(i, p) = \bar{r}_i + \frac{\sum_{j \in N} sim(i, j) \cdot (r_{jp} - \bar{r}_j)}{\sum_{j \in N} sim(i, j)} \quad (3)$$

式中, $pred(i, p)$ 为预测用户 i 对 p 项目的预测评分, \bar{r}_i 和 \bar{r}_j 分别为用户 i 和 j 的平均评分, N 为用户 i 的最近邻居集合。

2 基于正相关和负相关最近邻居的协同过滤算法

2.1 相似度计算

引入变异系数 (coefficient of variation) 对相似度的计算结果进行修正, 变异系数用以衡量数据的离散程度, 其值越大表明离散程度越大; 反之, 离散程度越小。

对于一个用户, 通过对项目的评分计算变异系数的值。变异系数的公式^[21]:

$$CV_i = \frac{\sqrt{\frac{1}{|P_i|} \sum_{p \in P_i} (r_{ip} - \bar{r}_i)^2}}{\bar{r}_i} \quad (4)$$

式中, CV_i 为用户 i 的变异系数, P_i 为用户 i 的评分项目集合, \bar{r}_i 为用户 i 对项目的平均评分, r_{ip} 为用户 i 对项目 p 的评分。

计算出所有用户的变异系数之后, 对于任意两个用户计算相似度, 可利用其变异系数的差值对相似度进行修正, 差值越小表明这两个用户评分的离散程度越接近, 由此引入平衡因子 λ , 计算公式:

$$\lambda(i, j) = 1 - |CV_i - CV_j| \quad (5)$$

式中, 当用户 i 和 j 的变异系数差值的绝对值越小, $\lambda(i, j)$ 值越大; 反之越小。

最后将平衡因子 λ 加入传统的相似度计算方法, 得到新的相似度计算公式 $simCV$:

$$simCV(i, j) = \lambda(i, j) \cdot sim(i, j) \quad (6)$$

式中, $sim(i, j)$ 为传统的相似度计算方法, 如第1.2节中提到的Pearson相关系数和修正的余弦相似度, 实验中采用Pearson相关系数。

改进后的用户相似度计算流程见图1。

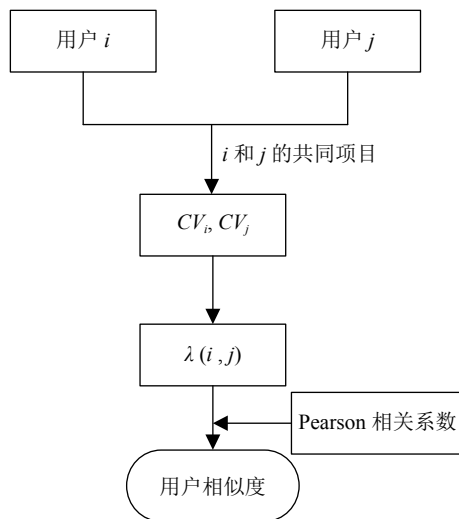


图1 用户相似度计算流程

Fig. 1 User similarity calculation flow chart

2.2 邻居选取

传统的协同过滤算法在选取最近邻居时往往只考虑正相关最近邻居, 即相似度为正的情况。由于Pearson相关系数和修正的余弦相似度取值为 $[-1, 1]$, 其中, 1代表强正相关, -1 代表强负相关。若两个用户之间的相似度取值越接近1, 表示两个用户相似性越大; 若两个用户之间的相似度取值越接近 -1 表示两个用户的差异性越大, 其对项目的评分喜好也越相反。据此可发现, 正相关最近邻居和负相关最近邻居都可为预测评分提供参考价值。因此, 在考虑正相关最近邻居的同时考虑负相关最近邻居, 对基于用户的协同过滤算法进行改进, 提出基于正相关和负相关最近邻居的协同过滤推荐算法。首先, 给出基于正相关最近邻居和负相关最近邻居的邻居选取算法, 如算法1所示; 对于正相关最近邻居的选取, 给定一个数值, 使得从正相关邻居中选取的正相关最近邻居个数等于给定的数值。其次, 对于负相关最近邻居的选取, 一般情况下, 认为只有具有较强负相关性的最近邻居才能从认知的相反角度, 提供推荐的有用信息。为避免选取负相关性不强的最近邻居影响推荐的准确度, 给定负相关最近邻居选取阈值 β , 只有负相关性满足 β 的负相关邻居才会被选择为负相关最近邻居。通常 $\beta \in [0, 1]$, 其合理取值将在第3.3节中进一步讨论。

算法1 基于正相关和负相关最近邻居的邻居选取算法。

输入: 用户-项目评分矩阵 R 、需要选取的正相关最近邻居个数 s 、阈值 β

输出: 正相关最近邻居列表 Pos 和负相关最近邻居列表 Neg

Step 1 根据用户-项目评分矩阵 R 计算出目标用

户 a 与其他用户 x 的相似度 $simCV(a, x)$ 。

Step 2 将 Step 1 中 $simCV(a, x) \geq 0$ 的邻居放入正相关邻居列表 P 中, $simCV(a, x) < 0$ 的邻居放入负相关邻居列表 N 中; 对 P 中的元素按相似度从大到小排序, 对 N 中的元素按相似度从小到大排序。

Step 3 对于给定需要选取的正相关最近邻居个数 s , 取 P 列表中的前 s 个正相关邻居加入正相关最近邻居 Pos 列表中; 同时遍历列表 N 中的负相关邻居, 若目标用户的相似度绝对值大于等于给定的 β 参数且 Neg 列表个数小于等于 s , 则加入负相关最近邻居 Neg 列表中。

Step 4 输出 Pos 列表和 Neg 列表。

基于正相关和负相关最近邻居的邻居选取算法流程如图 2 所示。

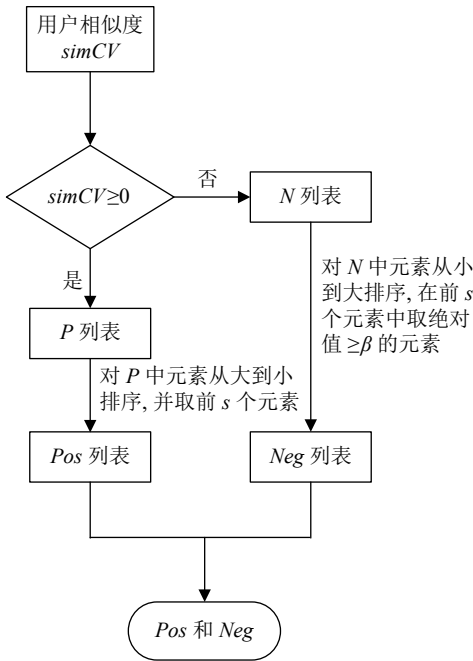


图 2 基于正相关最近邻居和负相关最近邻居的邻居选取算法流程

Fig. 2 Positive correlation and negative correlation nearest neighbors selection flow chart

2.3 预测评分

基于选取的正相关最近邻居和负相关最近邻居分别进行预测评分。最后, 将基于正相关最近邻居和负相关最近邻居的预测评分进行加权, 作为最终的预测评分。对于第 2.2 节得到的 Pos 和 Neg 列表, Pos 列表中的元素全部为正相关最近邻居, 相似度为正, 且值越大, 表明对同一物品的评分越接近; Neg 列表中的元素全为负相关最近邻居, 相似度为负, 且值越小, 表明对同一物品的评分差异越大。

对于 Pos 列表中的正相关最近邻居, 采用的预测评分公式:

$$pred_{Pos}(i, p) = \bar{r}_i + \frac{\sum_{j \in Pos} simCV(i, j) \cdot (r_{jp} - \bar{r}_j)}{\sum_{j \in Pos} simCV(i, j)} \quad (7)$$

式中, $pred_{Pos}(i, p)$ 为用户 i 对项目 p 的预测评分, \bar{r}_i 、 \bar{r}_j 分别为用户 i 和 j 的平均评分, r_{jp} 为用户 j 对项目 p 的评分, Pos 为正相关最近邻居集合, $sim(i, j)$ 为用户 i 和 j 的相似度。

对于负相关的最近邻居, 由于其与目标用户的评分呈相反趋势, 所以定义如下的预测评分:

$$pred_{Neg}(i, p) = \bar{r}_i - \frac{\sum_{j \in Neg} simCV(i, j) \cdot (r_{jp} - \bar{r}_j)}{\sum_{j \in Neg} simCV(i, j)} \quad (8)$$

式中, $pred_{Neg}(i, p)$ 为用户 i 对项目 p 的预测评分, \bar{r}_i 、 \bar{r}_j 分别为用户 i 和 j 的平均评分, r_{jp} 为用户 j 对项目 p 的评分, Neg 为负相关最近邻居集合, $sim(i, j)$ 为用户 i 和 j 的相似度。

最后设置权重参数 α , 将正、负相关最近邻居得到的预测评分 $pred_{Pos}$ 和 $pred_{Neg}$ 进行加权计算, 作为最终的预测评分。最终的预测评分计算公式为:

$$pred(i, p) = \alpha \cdot pred_{Pos}(i, p) + (1 - \alpha) \cdot pred_{Neg}(i, p) \quad (9)$$

式中, 权重参数 $\alpha \in [0, 1]$ 。

对于 α 的计算, 提出一种根据每个用户正、负相关最近邻居个数的比例动态计算权值 α 的方法:

$$\alpha = \frac{|Pos|}{|Pos| + |Neg|} \quad (10)$$

式中, Pos 为当前目标用户的正相关最近邻居集合, Neg 为当前目标用户的负相关最近邻居集合。

评分预测流程如图 3 所示。

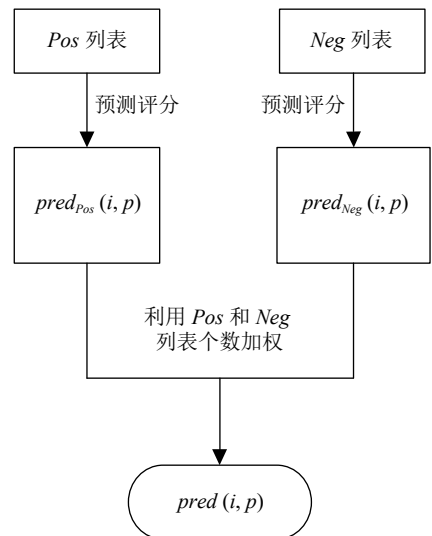


图 3 评分预测流程

Fig. 3 Ratings prediction flow chart

3 实验及结果分析

通过实验比较本文基于正相关和负相关邻居的协同过滤算法(PNCF)、传统的基于用户的协同过滤算法(CF)、基于优化用户相似度的改进协同过滤推荐算法(ICFOS)^[22]的推荐精度。并通过实验给出通常情况下阈值 β 的合理取值范围。

3.1 数据集

实验采用的MovieLens 100 k数据集是GroupLens小组从MovieLens网站(<http://movielens.org>)不同时期收集并整理的评分数据集合,由943名用户对1 682部电影的100 000个评分构成,评分值大小为1~5,评分值越大表明用户对电影的喜爱程度越大,且每个用户至少有20个评分。

3.2 评价标准

采用3种评价标准,分别为均方根误差(root mean squared error, RMSE)、平均绝对偏差(mean absolute error, MAE)^[23]及多样性(diversification)^[24]。MAE和RMSE都是利用预测评分与实际评分的偏差判断预测评分的准确性,其值越大,表明偏差越大,推荐精度越差;反之,表明推荐精度越好,预测越准确。多样性用以衡量用户推荐列表的多样化程度,其值越大,表示多样性越高。

MAE的定义公式:

$$MAE = \frac{\sum_{p \in T_i} |R_{i,p} - P_{i,p}|}{|T_i|} \quad (11)$$

RMSE的定义公式:

$$RMSE = \sqrt{\frac{\sum_{p \in T_i} (R_{i,p} - P_{i,p})^2}{|T_i|}} \quad (12)$$

多样性的定义公式:

$$Diversification = \frac{2 \sum_{i \neq j} (1 - \frac{|L_i \cap L_j|}{|L_i|})}{n(n-1)} \quad (13)$$

式中, $R_{i,p}$ 为用户*i*对项目*p*的实际评分, $P_{i,p}$ 为用户*i*对项目*p*的预测评分, T_i 为测试集中用户*i*评分的项目集合, L_i 和 L_j 分别为*i*和*j*用户的推荐列表。

3.3 阈值 β 取值及实验对比

由于基于正相关和负相关最近邻居的改进协同过滤推荐算法在选取负相关最近邻居时需要设置阈值 β ,阈值 β 的不同取值直接影响推荐的准确度。为给出 β 的合理取值,以0.1为步长,设置 β 取值为0、0.1、0.2、...、0.9、1;设置正相关最近邻居个数以10为间隔选取,分别取10、20、...、50。然后运行改进的协同过滤算法,比较对于不同的正相关最近邻居个数,当

β 取不同值时的MAE,以此对 β 参数进行训练,给出 β 的合理取值范围,结果如图4所示。

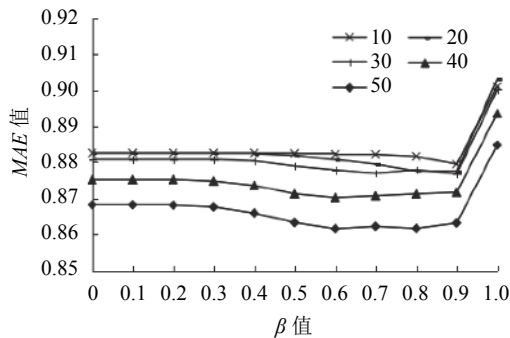


图4 基于不同阈值 β 的MAE实验结果

Fig. 4 MAE with different β

由图4可以看出,当 β 取值为0.8时,MAE值较低,推荐效果较好。故在接下来的实验中,设置阈值 β 取值为0.8。

比较本文基于正相关和负相关最近邻居的改进协同过滤算法(PNCF)、传统的基于用户的协同过滤算法(CF)、基于优化用户相似度的改进协同过滤推荐算法(ICFOS)^[22]的MAE、RMSE及多样性值。在本文算法中,设置阈值 β 为0.8,正相关最近邻居的个数从10到100,以步长10递增变化。

MAE值、RMSE值的对比实验结果分别如图5、6所示。

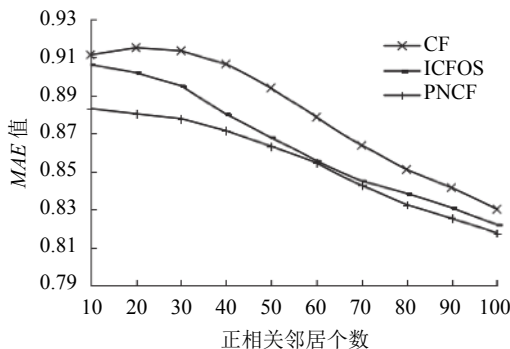


图5 MAE对比结果

Fig. 5 Comparison results of MAE

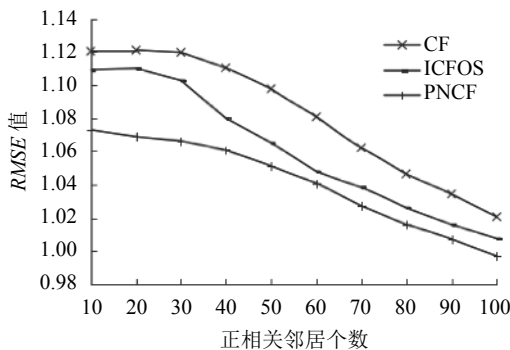
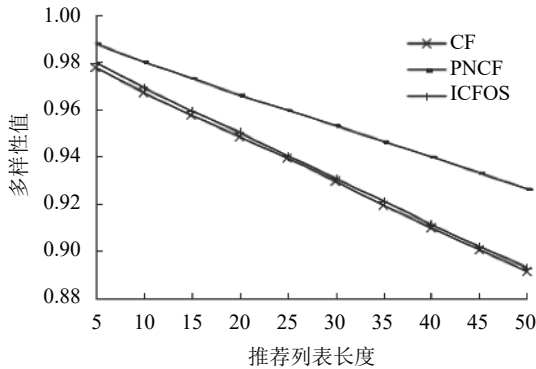


图6 RMSE对比结果

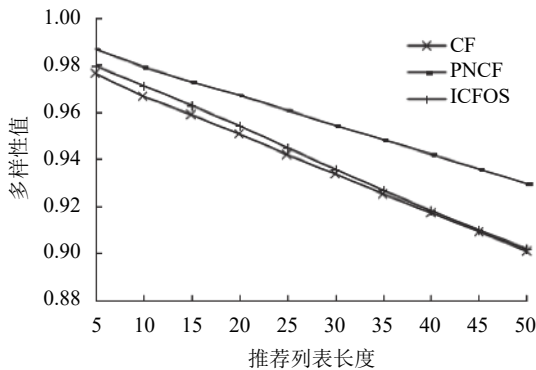
Fig. 6 Comparison results of RMSE

由图5可以看出,本文算法的MAE值,在所有给定的正相关最近邻居个数下,均低于CF算法;在大多数情况下均低于ICFOS算法。由图6可以看出,本文算法的RMSE值,在所有给定的正相关最近邻居个数下,均低于CF算法和ICFOS算法。因此PNCf算法具有更高的推荐准确率。

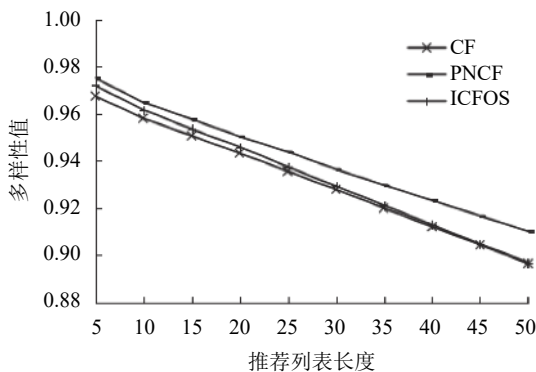
多样性值的对比结果如图7所示。由图7可以看出,当正相关最近邻居个数分别取10、20、50时,本文算法在所有给定的推荐长度 d 下多样性的值均高于CF算法和ICFOS算法。因此,本文算法在提高推荐准确率的同时也提高了推荐的多样性,给用户更加个性化的推荐。



(a) 正相关最近邻居个数为10



(b) 正相关最近邻居个数为20



(b) 正相关最近邻居个数为50

图7 多样性对比结果

Fig. 7 Comparison of Diversification

4 结 论

针对传统协同过滤算法推荐精度不高的问题,提出基于正相关和负相关最近邻居的协同过滤算法。算法对相似度的计算过程进行了修正,综合考虑正相关最近邻居和负相关最近邻居,基于选取的正相关最近邻居和负相关最近邻居分别进行预测评分,然后将两次结果进行加权计算,作为最终的预测评分。实验结果表明本文算法在有效提高推荐准确率的同时,使推荐更加多样化。但本文算法还有进一步改进的空间,由于正相关最近邻居及负相关最近邻居的选取需要相似度值介于 $[-1, 1]$,虽然常用的Pearson相关系数及修正的余弦相似度的值都满足该范围要求,但还有一部分相似度计算方法得到的值介于 $[0, 1]$,这样就无法选取负相关最近邻居。针对这个问题,可考虑利用相关函数将 $[0, 1]$ 的值映射到 $[-1, 1]$ 。下一步研究工作就是利用相关信息构建映射函数,从而解决部分相似度计算方法得到的结果不在 $[-1, 1]$ 的问题,确保负相关最近邻居的选取。

参考文献:

- [1] Meng Xiangwu, Liu Shudong, Zhang Yujie, et al. Research on social recommender system[J]. Journal of Software, 2015, 26(6): 1356–1372. [孟祥武, 刘树栋, 张玉洁, 等. 社会化推荐系统研究[J]. 软件学报, 2015, 26(6): 1356–1372.]
- [2] Zhang Jia, Lin Yaojin, Lin Menglei, et al. An effective collaborative filtering algorithm based on user preference clustering[J]. Applied Intelligence, 2016, 45(2): 230–240.
- [3] Wu Xiaokun, Cheng Bo, Chen Junliang. Collaborative filtering service recommendation based on a novel similarity computation method[J]. IEEE Transactions on Services Computing, 2017, 10(3): 352–365.
- [4] Meng Xiangwu, Hu Xun, Wang Licai, et al. Mobile recommender systems and their applications[J]. Journal of Software, 2013, 24(1): 91–108. [孟祥武, 胡勋, 王立才, 等. 移动推荐系统及其应用[J]. 软件学报, 2013, 24(1): 91–108.]
- [5] Wei Jian, He Jianhua, Chen Kai, et al. Collaborative filtering and deep learning based recommendation system for cold start items[J]. Expert Systems with Applications, 2017, 69: 29–39.
- [6] Bok K, Lim J, Yang H, et al. Social group recommendation based on dynamic profiles and collaborative filtering[J]. Neurocomputing, 2016, 209(C): 3–13.
- [7] Li Hongtao, He Keqing, Wang Jian, et al. A friends recommendation algorithm based on formal concept analysis and random walk in social network[J]. Journal of Sichuan University(Engineering Science Edition), 2015, 47(6): 131–138. [李宏涛, 何克清, 王健, 等. 基于概念格和随机游走的社交网朋友推荐算法[J]. 四川大学学报(工程科学版), 2015,

- 47(6):131–138.]
- [8] 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.
- [9] Su Jahwung, Chang Weiyi, Tseng V S. Effective social content-based collaborative filtering for music recommendation[J]. *Intelligent Data Analysis*, 2017, 21: S195–S216.
- [10] Zhang Yuchuan, Liu Yuzhao. A collaborative filtering algorithm based on time period partition[C]//Third International Symposium on Intelligent Information Technology and Security Informatics. New York: IEEE, 2010: 777–780.
- [11] Huang Chuanguang, Yin Jian, Wang Jing, et al. Uncertain neighbors' collaborative filtering recommendation algorithm[J]. *Chinese Journal of Computers*, 2010, 33(8): 1369–1377. [黄创光, 印鉴, 汪静, 等. 不确定近邻的协同过滤推荐算法[J]. *计算机学报*, 2010, 33(8): 1369–1377.]
- [12] He Liang, Wu Faqing. A time-context-based collaborative filtering algorithm[C]//IEEE International Conference on Granular Computing. New York: IEEE, 2009: 209–213.
- [13] Jiang Bo, Zhang Xiaoxiao, Pan Weifeng, et al. BIGSIR: A bipartite graph based service recommendation method[C]//Services. New York: IEEE, 2013: 363–369.
- [14] Lien D T, Phuong N D. Collaborative filtering with a graph-based similarity measure[C]//International Conference on Computing, Management and Telecommunications. New York: IEEE, 2014: 251–256.
- [15] Liu Qi, Chen Enhong. Collaborative filtering through combining bipartite graph projection and ranking[J]. *Journal of Chinese Computer Systems*, 2010, 31(5): 835–839. [刘淇, 陈恩红. 结合二部图投影与排序的协同过滤[J]. *小型微型计算机系统*, 2010, 31(5): 835–839.]
- [16] Wu Lei, Fang Qing, Jin Zhou. Improved personalized recommendation based on causal association rule and collaborative filtering[J]. *International Journal of Distance Education Technologies*, 2016, 14(3): 21–33.
- [17] Patra B K, Launonen R, Ollikainen V, et al. A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data[J]. *Knowledge-Based Systems*, 2015, 82(C): 163–177.
- [18] Yang Chong, Yu Xiaohui, Liu Yang, et al. Collaborative filtering with weighted opinion aspects[J]. *Neurocomputing*, 2016, 210(C): 185–196.
- [19] Liu Qi, Chen Enhong, Xiong Hui, et al. Enhancing collaborative filtering by user interest expansion via personalized ranking[J]. *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society*, 2012, 42(1): 218–233.
- [20] Schafer J B, Dan F, Herlocker J, et al. Collaborative filtering recommender systems[J]. *Acm Transactions on Information Systems*, 2004, 22(1): 5–53.
- [21] Bedeian A G, Mossholder K W. On the use of the coefficient of variation as a measure of diversity[J]. *Organizational Research Methods*, 2000, 3(3): 285–297.
- [22] Chen Hao, Li Zhongkun, Hu Wei. An improved collaborative recommendation algorithm based on optimized user similarity[J]. *Journal of Supercomputing*, 2016, 72(7): 2565–2578.
- [23] Xie Feng, Chen Zhen, Shang Jiaying, et al. Item similarity learning methods for collaborative filtering recommender systems[C]//IEEE, International Conference on Advanced Information Networking and Applications. New York: IEEE, 2015: 896–903.
- [24] Shang Mingsheng, Zhang Zike, Zhou Tao, et al. Collaborative filtering with diffusion-based similarity on tripartite graphs[J]. *Physica A Statistical Mechanics & Its Applications*, 2009, 389(6): 1259–1264.

(编辑 李轶楠)

引用格式: Xu Yi, Tang Yimin, Wang Ran. Collaborative filtering algorithm based on positive correlation and negative correlation nearest neighbors[J]. *Advanced Engineering Sciences*, 2018, 50(5): 189–195. [徐怡, 唐一民, 王冉. 基于正相关和负相关最近邻居的协同过滤算法[J]. *工程科学与技术*, 2018, 50(5): 189–195.]