基于项目聚类和时间衰减的动态协同过滤算法

刘 旭,李玲娟
(南京邮电大学 计算机学院,江苏 南京 210023)

摘 要:传统协同过滤推荐算法侧重于用户兴趣和项目的关系,目的是向用户推荐符合其兴趣的项目。但忽略了用户兴趣随时间的变化,将不同时间段的项目评分同等对待,降低了推荐的准确率。另一方面,基于项目的协同过滤算法在寻找目标项目的最近邻居时,因需要遍历整个项目空间而导致开销较大。为了解决上述问题,设计了一种基于项目聚类和时间衰减的动态协同过滤推荐算法 ITDCF。该算法适用于基于项目的协同过滤,首先根据用户的评分对项目进行聚类,以快速找出目标项目的最近邻。接着,在计算项目相似度和预测评分阶段都引入时间衰减因子,以客观反映用户兴趣,提高推荐精度。最后,将前 N 个项目推荐给用户。在 MovieLens 数据集上对 Popular、ItemCF、ITDCF 算法的准确率、召回率和 F_1 值的测试结果表明,ITDCF 算法在准确性和效率上都有所提高。

关键词:推荐算法;聚类;协同过滤;时间衰减;基于项目
中图分类号:TP311 **文献标识码**:A **文章编号**:1673-629X(2020)08-0022-05
doi:10.3969/j.issn.1673-629X.2020.08.004

Dynamic Collaborative Filtering Algorithm Based on Item Clustering and Time Decay

LIU Xu,LI Ling-juan
(School of Computer Science,Nanjing University of Posts and Telecommunications,Nanjing 210023,China)

Abstract:The traditional collaborative filtering recommendation algorithm focuses on the relationship between user interests and items, with the aim of recommending users to items that match their interests. However,the change of user interest over time is ignored,and the item scores of different time periods are treated equally,which reduces the accuracy of the recommendation. In addition,the item-based collaborative filtering algorithm leads to a large overhead because it needs to traverse the entire item space when searching for the nearest neighbor of the target item. In order to solve the above problems,we design a dynamic collaborative filtering recommendation algorithm ITDCF based on item clustering and time decay,which is suitable for item-based collaborative filtering. Firstly,the item is clustered according to the user's score to quickly find the nearest neighbor of the target item. Then,the time decay factor is introduced in both the calculation item similarity and the prediction score stage to objectively reflect the user's interest and improve the recommendation accuracy. Finally,the top N items are recommended to the user. The precision,recall and F_1 values of Popular,ItemCF and ITDCF algorithms are tested on the MovieLens dataset. The results show that the ITDCF algorithm has improved accuracy and efficiency.

Key words:recommendation algorithm;clustering;collaborative filtering;time decay;item-based

0 引 言

随着互联网的发展和进步,人们在享受网络 资源极大便利的同时,也受到信息超载的困扰,推荐系统是解决信息过载问题的主流方法。其中最著名的便是协同过滤推荐算法[1]。该算法有基于用户的和基于项目的两种类型,两者都基于用户的历史记录信息,前者结合与目标用户有类似偏好的其他用户兴趣,后者 基于项目间的相似性,把用户可能感兴趣的项目快速呈现。

尽管协同过滤推荐算法在个性化 推荐中有明显的优点:基于用户行为和项目相似性,不需要先验知识,在用户行为比较丰富的时候,推荐效果不错,但是有关用户兴趣随时间变化的事实被忽略了。在基于项目的协同过滤算法中,计算最近邻项目时,将不同时段的项目评分同等 对待,以致寻找到的目标项目的最近邻,有可能不是真正意义上的最近邻居,使推荐结果存在较大偏差。另外,基于项目的协同过滤算法搜索目

收稿日期:2019-10-16 **修回日期**:2020-02-25
基金项目:国家自然科学基金(61302158,61571238)
作者简介:刘 旭(1997-),女,硕士研究生,CCF 会员(E7638G),研究方向为数据挖掘与个性化推荐;通讯作者:李玲娟(1963-),女,教授,研究方向为数据挖掘、信息安全、分布式计算。

标项目的最近邻居时,需要遍历整个项目空间,费时又费力[2]。

文中以基于项目的协同过滤算法为研究对象,设计了一种基于项目聚类和时间衰减的动态协同过滤算法(dynamic collaborative filtering algorithm based on item clustering and time decay, ITDCF)。首先根据用户的评分对项目进行聚类,接下来在计算相似度时引入时间衰减因子反映用户兴趣的变化,在预测评分时也考虑时间衰减。最后,生成 Top-N 推荐列表,提高了推荐的准确性和效率。

1　相关研究

1.1　协同过滤算法

协同过滤(collaborative filtering, CF)算法的基本思想是:基于过去的行为或现有用户组的意见来预测数据,并使用与当前用户或当前项目类似的邻居数据来生成推荐结果[3]。该算法的输入是用户-项目评分矩阵,输出数据一般分为两类:当前用户对项目偏好的预测值和 Top-N 的推荐项目列表。其基本步骤包括:数据的收集与处理、生成用户-项目评分矩阵、计算相似度、产生最近邻、预测评分和产生 Top-N 推荐[3]。

基于用户的协同过滤算法(UserCF)[4]的主要思想是:首先,输入评分数据集和当前用户 ID,以查找出与当前用户有相似偏好的其他用户,这些用户被称为最近邻居;然后,利用邻居用户预测项目的评分;对所有项目的评分按照从大到小进行排序,将评分居前的 N 个项目推荐给当前用户。

基于项目的协同过滤算法(ItemCF)[4]的核心思想是:首先构建一个项目相似度矩阵来描述两个项目之间的相似性;找出与当前项目相似的 k 个最近邻项目,然后根据 k 个最近邻计算当前用户没有看到的每一项目 i 的用户评分;最后,将用户对所有项目的评分从大到小排序,向当前用户推荐所有项目中得分最高的前 N 项。

尽管 UserCF 算法在推荐领域得到了广泛的应用,但也面临着很多挑战。像电商网站,一方面,项目的数量比较稳定,但用户数目更新频率较高,当用户数量远大于项目数量时,计算用户间的相似性将越来越耗时并占用更多内存[5]。另一方面,基于用户的算法生成的推荐结果可解释性较差[6]。而 ItemCF 算法时间复杂度相对较低,并能根据用户的历史行为做出推荐解释,可以比较容易令用户信服。

1.2　基于项目的协同过滤算法

基于项目的协同过滤算法(ItemCF)的基本步骤是:处理数据、生成用户-项目评分矩阵、计算项目相似度、产生最近邻、预测评分和产生 Top-N 推荐。

该算法利用式(1)所示的余弦相似度计算两个项目之间的相似度。

$$\mathrm{sim}(i,j)=\frac{|N(i)\cap N(j)|}{\sqrt{|N(i)||N(j)|}} \tag{1}$$

其中,$N(i)$ 是评价了项目 i 的用户,$N(j)$ 是评价了项目 j 的用户,$|N(i)|$ 是评价了项目 i 的用户数,$|N(j)|$ 是评价了项目 j 的用户数,$|N(i)\cap N(j)|$ 是同时评价了项目 i 和 j 的用户数。相似性的范围为[0,1],值越接近 1,越相似。

此外,基于所获取的 k 个最近邻项目后,用式(2)预测当前用户对目标项目的兴趣。

$$p(u,i)=\sum_{j\in N(u)\cap S(i,k)}\mathrm{sim}(i,j)\,r_{uj} \tag{2}$$

其中,$N(u)$ 是用户 u 评价过的项目集合,$S(i,k)$ 包含了和项目 i 最相似的 k 个项目,项目 j 属于和用户评价过的最相似的项目的集合,$\mathrm{sim}(i,j)$ 是项目 i 和 j 的相似度,r_{uj} 是用户 u 对项目 j 的兴趣。根据数据集的类型,这里的 $r_{uj}=1$。

虽然 ItemCF 算法在项目数明显少于用户数的场合,相对于 UserCF 算法有一定的优势,但是如果项目过多,计算项目之间的相似度矩阵的代价偏高,而且算法也忽略了用户兴趣随时间变化对推荐效果产生的影响[7]。

2　ITDCF 算法

文中为进一步提升 ItemCF 算法效率而设计的基于项目聚类和时间衰减的动态协同过滤推荐算法 ITDCF 包括:项目聚类、时间加权、产生最近邻、预测评分、产生推荐等步骤。

2.1　项目聚类

针对基于项目的协同过滤算法在寻找目标项目的最近邻居时,因需要遍历整个项目空间而导致耗时大的问题[7],ITDCF 算法先对项目进行聚类,选择评分数量达到一定标准的项目为初始簇类中心,可以认为这些初始中心就是最受用户喜欢的项目;然后遍历所有项目,计算其与初始中心的相似度,并归入相似度最高的簇中,从而使目标项目最近邻居的计算从全局空间转到簇内空间,大大降低了计算量,提升推荐的时效性。

项目聚类的具体处理过程如下:

输入:类簇的数目 K 和数据集合。

输出:K 个簇。

步骤 1:选择初始中心点。

根据构建的用户-项目评分矩阵 $\boldsymbol{R}$,从项目集合中检索所有 n 个项目,计算每个 item 的评分数量 V。将所有的 V 值按降序排列,选择前 K 个 V 值大的项目

作为初始的聚类中心，得到 K 个聚类簇 $\{C_1, C_2, \cdots, C_k\}$。

步骤 2：利用余弦相似度公式计算 item 分别到初始化聚类中心的距离。

步骤 3：按照与聚类中心距离最近（相似度最高）的原则，将各个 item 分配到最邻近的簇中，并记录下 item 与每个聚类中心的相似度。

步骤 4：算法结束，输出聚类结果。

聚类时，以评价数量最多的 K 个 item 作为初始的聚类中心，减少了迭代次数，提高了算法效率。在这里，需要通过尝试来确定最优的 K 值。

采用余弦相似度公式，将项目间的相似度作为评判距离的标准，相似度越大则距离越近。采用余弦相似度公式还可以避免由于各属性衡量单位的差异性而导致的“相似不相同”问题[8]。

2.2 时间加权

考虑到用户兴趣会随着时间而改变[9]，ITDCF 算法将时间信息加入到推荐算法中，以在特定时间向用户推荐其最感兴趣的项目。ITDCF 算法对基于项目的协同过滤推荐算法的改进 有两个方面：计算相似度时加入时间衰减因子和预测评分时加入时间衰减因子。其中时间衰减函数选择的是艾宾浩斯曲线。

艾宾浩斯曲线是由德国心理学家艾宾浩斯发现的，描述了人类大脑忘记新事物的规律[10]。将之应用到推荐系统上就是，用户对越久远的项目越不感兴趣，在向用户进行推荐时，将他以前看到过的项目进行适当地降权。但是如果用户最近又对其中一个项目进行了操作，那么在推荐时就增加这类物品的权重。这样得到的最近邻项目更能符合用户的兴趣。

时间加权[11]的具体任务是：拟合艾宾浩斯曲线，编写加权函数。

随着时间的变化，人的兴趣会逐渐衰减，但衰减趋势为先快后慢[12]。所以指数函数在一定程度上更符合兴趣的衰减特性。拟合所依赖的带参数指数函数为：

$$W_t = Ae^{Bx} + Ce^{Dx} \tag{3}$$

艾宾浩斯遗忘曲线的最终拟合公式为：

$$W_t = 32.6e^{-0.016\frac{|T_{ui}-T_{uj}|}{60}} + 37.7e^{-284*\frac{|T_{ui}-T_{uj}|}{60}} \tag{4}$$

其中，T_{ui} 表示用户 u 对项目 i 操作的时间，T_{uj} 表示用户 u 对项目 j 操作的时间，单位为 min。

2.3 产生最近邻

在加入时间权重后，认为项目 i 和 j 相似是因为在同一时间内，二者同时被多个用户选择过[13]。考虑到艾宾浩斯遗忘曲线对用户偏好的影响，用户选择时间间隔越短，项目间的相似度越高，文中采用余弦相似函数[14]来判断两个项目之间的相似度，得到计算项目 i 和 j 相似度的改进公式为：

$$\mathrm{Sim}(i,j) = \frac{\sum_{u\in N(i)\cap N(j)} w_t}{\sqrt{|N(i)||N(j)|}} \tag{5}$$

其中，$N(i)$ 是评价了项目 i 的用户，$N(j)$ 是评价了项目 j 的用户，$|N(i)|$ 是评价了项目 i 的用户数，$|N(j)|$ 是评价了项目 j 的用户数，用户 u 属于同时评价了项目 i 和项目 j 的用户集合，新增的 W_t 为时间衰减函数。

因为在计算簇集的时候保存了每个项目和聚类中心的相似度，所以根据记录直接将目标项目划分到相似度最大的类簇中；再取类簇内与目标项目最为相似的前 K 个项目作为最近邻居。

2.4 预测评分

用户最近的行为比用户远期的行为更能反映用户当前的兴趣[15]。通过项目相似性预测用户对项目的兴趣，并考虑用户最近的行为。预测时采用如下公式：

$$p(u,i) = \sum_{j\in N(u)\cap S(i,k)} \mathrm{sim}(i,j)\frac{1}{1+\alpha|t_0 - t_{uj}|} \tag{6}$$

其中，$N(u)$ 是用户 u 评价过的项目集合，$S(i,k)$ 包含了和项目 i 最相似的 k 个项目，项目 j 属于和用户评价过的最相似的项目的集合，t_0 表示当前时间，t_{uj} 代表用户 u 操作项目 j 的时间；对于时间衰减参数 α，不同数据集中的值是不同的[16]，用户的兴趣变化越快，值越大。

最后，根据预测评分公式和得到的 K 个最近邻项目，预测用户的兴趣度，排序后得到 Top-N 推荐。

基于以上步骤，ITDCF 算法的流程可归纳为图 1。

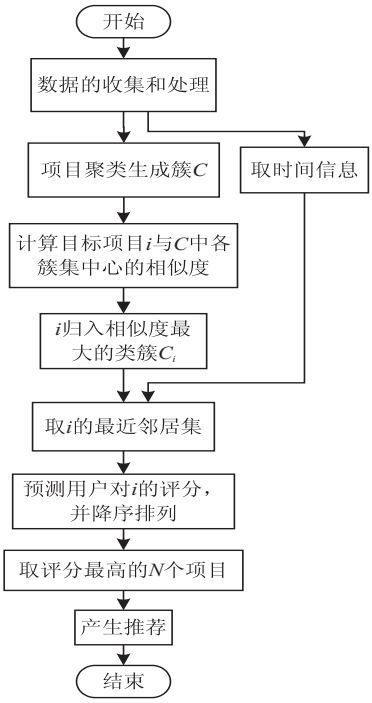

图 1 ITDCF 算法的推荐流程

3　算法性能测试

3.1　实验数据集及划分

MovieLens 数据集[17]是研究人员广泛使用的一种经典数据集,用于测量推荐算法的精度。

文中主要使用 U. data 文件,其中用户的评分值为 1 至 5 分,代表评价从低到高。timestamp 是自 1970 年 1 月 1 日零点后到用户提交评价的时间的秒数。

3.2　实验环境

文中的实验硬件环境如下:Windows7;intel(R) Core(TM) i5-6200U CPU @2.30 GHz 2.40 GHz;4G 内存;64 位操作系统,基于 x64 的处理器。

实验在 JetBrains PyCharm 5.0.3 平台下运行;使用 Python 语言,环境为 python3.6.3;Python 包有:numpy、math、matplotlib.pyplot 等。

3.3　实验方案

按照每个用户选择项目的时间先后对 U. data 的项目进行排序,然后将用户最后选择的项目作为测试集,并把这之前用户对项目的选择记录作为训练集。利用改进后的算法构建用户兴趣模型,向每个用户推荐 N 个物品,并且利用准确率、召回率和 F_1 值对推荐算法的性能进行评价。

为了检验 ITDCF 算法的性能,文中将之与 Popular 算法和基于项目的协同过滤算法 ItemCF 进行对比,其中的 Popular 算法是按照物品的流行度高低向用户推荐当天最受欢迎物品的算法。

给定时间 T,项目 i 最近的流行度 $p_i(T)$ 可以定义为:

$$p_i(T)=\sum_{(u,i,t)\in \mathrm{Train},t<T}\frac{1}{1+\alpha(T-t)} \tag{7}$$

其中,三元组(u,i,t)表示用户 u 在 t 时刻评价了项目 i,Train 是测试集数据集,α 是时间衰减参数。

通过试验取最佳类簇数 7;设定目标项目的最近邻项目的阈值参数从 0 增加到 45,增加的间隔为 5;进行 10 次实验;观察召回率、准确率和 F_1 值的变化趋势。

召回率为推荐列表中预测的用户感兴趣的项目与系统中用户真喜欢的所有项目的百分比[18]。计算公式为:

$$\mathrm{Recall}=\frac{\sum_u |R(u,N)\cap T(u)|}{\sum_u |T(u)|} \tag{8}$$

其中,$R(u,N)$ 是给用户 u 提供的长度为 N 的推荐列表,$T(u)$ 是测试集中用户评价过的项目集合。

准确率为推荐列表中用户喜欢的项目在所有被推荐的项目中所占的百分比[18],计算公式为:

$$\mathrm{Precision}=\frac{\sum_u |R(u,N)\cap T(u)|}{\sum_u |R(u,N)|} \tag{9}$$

其中,$R(u,N)$ 表示算法给用户 u 提供的长度为 N 的推荐项目列表,$T(u)$ 为测试集中用户喜欢的物品集合。

召回率 Recall 说明了推荐列表的覆盖性,Precision 说明了推荐列表的准确率,都是衡量推荐性能的重要参数[18]。用 F_1 值来拟合 Precision 与 Recall,其中 P 为 Precision,R 为 Recall。F_1 值越大,表明算法的推荐效果越好。计算公式如下:

$$F_1=\frac{2PR}{P+R} \tag{10}$$

3.4　实验结果分析

Popular 算法、ItemCF 算法和 ITDCF 算法在不同推荐项目数 N 值下的准确率、召回率和 F_1 值分别如图 2 至图 4 所示。

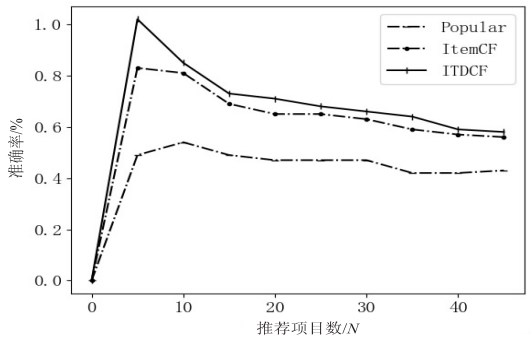

图 2　MovieLens 数据集上各算法在不同 N 值下的准确率

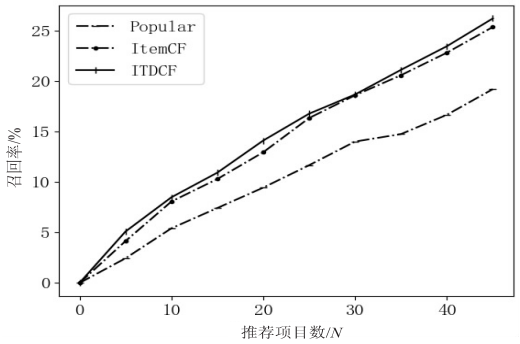

图 3　MovieLens 数据集上各算法在不同 N 值下的召回率

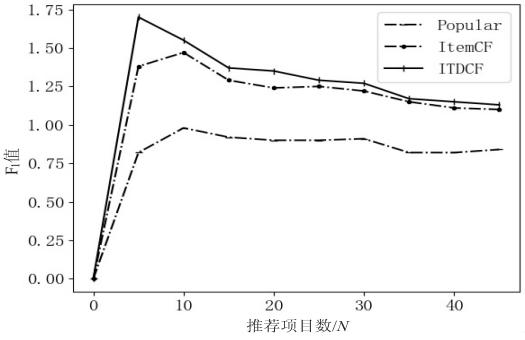

图 4　MovieLens 数据集上各算法在不同 N 值下的 F_1 值

由图 2 可以看出,对 MovieLens 数据集,ITDCF 算法在不同推荐项目数下的准确率都更高,这是由于用户的兴趣随时间而改变,在计算相似度时考虑了用户对项目的遗忘,因此提高了预测项目评分的准确性。同时可以看出选择合适的推荐项目数 N,可以获得最好的推荐效果。

由图 3 和图 4 可以看出,ITDCF 算法的召回率和 F_1 值在整体上也比 Popular、ItemCF 高。

4 结束语

以提高推荐的准确率为目标,基于协同过滤和流行度推荐的思想,引入聚类和进一步考虑兴趣随时间的变化因素,设计了一种基于项目聚类和时间衰减的动态协同过滤推荐算法。该算法根据评分数量选择初始簇类中心,减少了迭代次数,并缩小了寻找目标项目最近邻的候选集;通过在基于项目的协同推荐算法中加入时间衰减因子提高了推荐的精度。MovieLens 数据集上对 ITDCF 算法、Popular 算法和 ItemCF 算法的准确率、召回率和 F_1 值的对比实验结果表明,ITDCF 算法在推荐的准确性和效率方面有所提高。

参考文献:

[1] GOLDBERG D,NICHOLS D,OKI B M,et al. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM,1992,35(12):61-70.

[2] 邓华平. 基于项目聚类和评分的时间加权协同过滤算法[J]. 计算机应用研究,2015,32(7):1966-1969.

[3] 冷亚军,陆 青,梁昌勇. 基于结构相似性的协同过滤推荐算法[J]. 小型微型计算机系统,2015,36(10):2266-2269.

[4] HAMMOU B A,LAHCEN A A. FRAIPA:a fast recommendation approach with improved prediction accuracy[J]. Expert Systems with Applications,2017,87:90-97.

[5] 刘 辉,郭梦梦,潘伟强. 个性化推荐系统综述[J]. 常州大学学报:自然科学版,2017,29(3):51-59.

[6] ZHAO Z D,SHANG M S. User-based collaborative-filtering recommendation algorithms on Hadoop[C]//Third international conference on knowledge discovery and data mining. Phuket,Thailand:IEEE,2010:478-481.

[7] CHOI K, SUH Y. A new similarity function for selecting neighbors for each target item in collaborative filtering[J]. Knowledge-Based Systems,2013,37(1):146-153.

[8] 李 俊,李玲娟. 基于最小生成树的 K-均值算法设计与并行化实现[J]. 南京邮电大学学报:自然科学版,2017,37(5):81-86.

[9] BOBADILLA J,ORTEGA F,HERNANDO A,et al. Recommender systems survey[J]. Knowledge-Based Systems,2013,46:109-132.

[10] WU Y,WANG Y,TANG Z. A collaborative filtering recommendation algorithm based on interest forgetting curve[J]. International Journal of Advancements in Computing Technology,2012,4(10):148-157.

[11] 董立岩,王越群,贺嘉楠,等. 基于时间衰减的协同过滤推荐算法[J]. 吉林大学学报:工学版,2017,47(4):1268-1272.

[12] ZENG L,LING L. An interactive vocabulary learning system based on word frequency lists and Ebbinghaus´ curve of forgetting[C]//Digital media and digital content management (DMDCM). NJ:IEEE,2011:313-317.

[13] 刘 云,王 颖,亓国涛,等. 时间上下文的协同过滤 Top-N 推荐算法[J]. 计算机技术与发展,2017,27(7):79-82.

[14] 王永康,袁卫华,张志军,等. 融合时间隐语义填充和子群划分的推荐算法[J]. 计算机工程与应用,2019,55(16):130-137.

[15] 乔平安,曹 宇,任泽乾. 融合隐语义模型和 K-meansplus 聚类模型的推荐算法[J]. 计算机与数字工程,2018,46(6):1108-1111.

[16] 龚 敏,邓珍荣,黄文明. 基于用户聚类与 Slope One 填充的协同推荐算法[J]. 计算机工程与应用,2018,54(22):139-143.

[17] HARPER F M,KONSTAN J A. The movielens datasets[J]. ACM Transactions on Interactive Intelligent Systems,2016,5(4):1-19.

[18] HERLOCKER J L,KONSTAN J A,TERVEEN L G,et al. Evaluating collaborative filtering recommender systems[J]. ACM Transactions on Information Systems,2004,22(1):5-53.