

基于信任网络的协同过滤推荐方法

张洪波 王佳蕾 张丽娟 刘志宏

(西安电子科技大学网络与信息安全学院 西安 710071)

摘要 经典的协同过滤推荐系统存在数据稀疏和冷启动问题。利用信任网络能够有效地解决此问题,但性能有待提高。根据“如果 a 信任 b,则 a 与 b 相似度高的概率较大”这一普适规律,提出一种基于信任网络的协同过滤推荐算法。该算法采用惩罚、奖励机制,进一步提高了推荐系统的性能。算法将覆盖率和准确率作为衡量标准,与经典协同过滤算法和已有信任推荐算法进行实验对比,结果表明所提推荐方法的性能更好。

关键词 推荐系统,信任网络,协同过滤,冷启动

中图分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.08.026

Trust Network Based Collaborative Filtering Recommendation Algorithm

ZHANG Hong-bo WANG Jia-lei ZHANG Li-juan LIU Zhi-hong

(School of Cyber Engineering, Xidian University, Xi'an 710071, China)

Abstract The problems of data sparsity and cold start cannot be solved by the classical collaborative filtering recommendation schemes. Although these problems can be solved effectively by exploiting the trust networks of users, the performance of these schemes need to be improved. Based on the ubiquitous phenomenon of “if a trusts b, then the similarity between a and b is relatively high”, this paper proposed a collaborative filtering recommendation algorithm, which exploits a penalty and reward mechanism to further promote its performance. Then it was compared with the classical collaborative filtering algorithms and the existing trust recommendation algorithms in terms of the coverage and accuracy. The results show that the performance of the proposed algorithm is improved.

Keywords Recommendation system, Trust network, Collaborative filtering, Cold start

1 引言

“我该买什么款式的图书? 我该买什么类型的衣服? 我该看什么电影?”身处互联网时代,丰富的信息是人们的“宝藏”,但同时也给用户的选择带来了巨大的障碍。如何解决信息超载问题,帮助用户迅速、有效地从巨大的信息量中找到需要的信息,一直是人们研究的热点。推荐系统是解决信息过载问题的有效技术。目前采用的推荐方法主要包括协同过滤推荐、基于内容的推荐以及混合推荐等^[1]。

协同过滤推荐^[2]的基本思想是,根据已有用户群的行为或意见,预测当前用户最喜欢哪些事物或者对哪些事物感兴趣。如果多个用户在过去有相同的偏好(例如他们看过同一部电影或买过同一款衣服),则他们对其他同类产品很可能也具有相似的偏好。协同过滤算法可以分为 3 种类型:基于用户的最邻近推荐、基于项目的最邻近推荐和基于模型的协同过滤。协同过滤技术的输入只有给定的用户-项目评分矩阵,算法比较简便,存在一些问题,如评分预测的准确度低、数据稀疏以及用户或项目冷启动等^[3]。

为了解决协同过滤存在的问题,研究人员尝试过多种办法,结合社交网络的信息推荐系统是其中之一。Palau 等^[4]从

宏观角度研究了网络结构对推荐效果的影响;Yuan 等^[5]验证了社交网络具有小世界特性,并利用这一特性证明了将信任的最大传递距离设置为 5 较为合适;童向荣等^[6]研究了主观信任传递闭包的相关问题;Hwang 等^[7]分析了推荐系统中的 local 和 global 信任矩阵,并且证明了两者的提高推荐系统的准确率和覆盖率。Golbeck 等^[8]提出采用扩展的广度优先的搜索算法获得推荐者列表的 TidalTrust 算法,该算法以最短路径距离寻找推荐者,并以活跃用户与推荐者间的信任值为权重预测对目标项目的评分。Messa 等^[9]通过对用户和项目进行分类处理,提出“采用向后搜索方法的以评分矩阵、信任网络、最大信任传递距离作为算法的输入”的 MoleTrust 算法。以上 3 种算法都能在一定程度上缓解推荐系统的冷启动及数据稀疏问题,但它们在评分预测的准确度方面仍有待提高。Moradi 等^[10]提出了一种基于可靠性的信任相关的协同过滤算法——RTCF,用以提高基于信任网络的推荐系统的准确性。Jamali 等^[11]提出了 TrustWalker 方法,该算法以 Random Walk 策略搜索信任网络,同时利用 item 间的相似性降低了搜索算法的时间复杂度。Zhao 等^[12]提出一种通过利用社交网络解决隐式评分推荐系统中的冷启动等问题的方法——SBPR(Social Bayesian Personalized Ranking)。Azad-

到稿日期:2017-01-06 返修日期:2017-04-08 本文受 111 基地(B16037),国家自然科学基金(U1405255)资助。

张洪波(1991—),男,硕士生,主要研究方向为信任管理、推荐系统、物理层安全,E-mail:284477545@qq.com;王佳蕾(1992—),女,硕士生,主要研究方向为信任管理、推荐系统、物理层安全,E-mail:1192363902@qq.com;张丽娟(1991—),女,硕士生,主要研究方向为信任管理、推荐系统、物理层安全,E-mail:838894914@qq.com;刘志宏(1968—),男,博士,副教授,主要研究方向为密码学、信息安全、网络编码、复杂网络、传感器网络等。

jalal 等^[13]利用博弈论的思想来提高推荐系统的准确性及覆盖率。

通过用户已有的社交关系和信任的可传递性能够有效地解决数据稀疏和冷启动等问题,但已有 TARS(Trust-Aware Recommendation System)只考虑了用户间的信任关系,没有考虑到两个彼此信任的用户可能对 Item 有不同的评判标准这一情况,因此推荐算法的准确性受到影响。评分预测的准确性是衡量推荐系统好坏的重要指标,已有算法的预测准确性仍有待提高。为了进一步提升推荐系统的预测准确性,本文提出一种基于动态信任网络的协同过滤算法(Dynamic Trust-aware Collaborative Filtering, DTCF)。通过信任网络寻找与活跃用户相似度较高的用户,再根据用户间的相似度调整信任网络,最后以覆盖率和准确度作为衡量标准将所提方法与经典协同过滤算法和已存在的信任推荐算法进行实验对比。结果表明,DTCF 算法的性能较已有算法有了进一步的提升。

本文第 2 节主要介绍信任的定义、传递、聚合及信任值的动态更新等内容;第 3 节介绍 DTCF,主要涉及推荐者列表的获取算法、评分预测算法和信任网络动态更新算法等;第 4 节为实验部分;最后总结全文。

2 动态信任模型

2.1 信任的定义、度量以及表示方法

本文涉及的信任是指用户能够在多大程度上依赖其他用户给出的推荐做出评分的预测。

信任有多种度量方法,比较常见的有二元值法、多元离散值法和连续值法^[14]。本文以[0,1]间的连续数值表示用户之间的信任程度。信任值为 0 则表示完全不信任,信任值为 1 则表示完全信任,信任值为[0.5,1]则表示用户之间存在一定程度的信任。

用户间的信任关系可以采用加权有向图表示。一个信任图可表示为 $G(V, E, T)$ 。其中, V 表示节点集合,代表社交网络中的用户集合; E 表示网络中的边,代表用户间的信任关系; T 表示边的权重,代表用户间实际的信任值。本文只考虑用户间的信任关系, G 中的边的权重大于或等于 0.5。

2.2 信任的传递

信任具有一定程度的可传递性(朋友的朋友还可能是朋友),但信任的传递随着传播路径的增长而逐级衰减,而信任的可知度不会随信任的传递而增加^[15]。由于以乘法运算作为用户间的信任值传递符合信任逐级递减的特性,本文与大多数涉及信任传递的研究工作相同,采用乘法运算作为用户间信任值衰减的计算方法^[16]。给定信任网络 $G(V, E, T)$, 则:

$$T(u, r) = T(u, i) \times T(i, u), i \in N_u \tag{1}$$

其中, N_u 表示节点 u 的直接邻居集合, $u, i, v \in V, T(u, v)$ 表示 u 对 v 的信任值。

在图 1 的信任网络中,用户 u_1 对 u_4 的信任值为 $0.9 \times 0.8 = 0.72$,用户 u_1 对 u_5 的信任值为 $0.6 \times 0.7 = 0.42$ 。

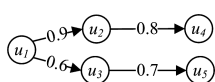


图 1 信任传递示例

Fig. 1 Example of trust transfer

2.3 信任的聚合

在实际信任网络中,两个节点间的信任传递路径通常不止一条,因此需要考虑如何处理多路径上的信任的聚合问题。常用的信任聚合算法有累加、最大值、平均值、加权以及加权平均等。考虑到计算复杂度以及信任路径存在闭环等问题,本文采用最大值法,即将多条路径中最终信任值最大的路径设为有效信任传递路径。

2.4 信任网络的动态更新

确定信任的聚合与传递计算方法后,就确定了静态信任模型。在现实社交网络中,人与人的信任关系及信任程度会逐渐发生变化。为了描述信任网络随用户交互的动态变化,提出一种带有奖惩机制的高效更新模型,此部分内容将在第 3 节中详细描述。

3 基于动态信任网络的协同过滤算法(DTCF)

本文提出的基于动态信任网络的协同过滤算法(DTCF)采用的推荐过程如图 2 所示,共由 6 个部分组成,其中评分矩阵(记录各个用户的评分信息)和信任矩阵(记录用户间的信任关系)作为算法的输入。假设推荐系统要预测用户 u 对项目 a 的评分,工作流程分为 3 个步骤:获取推荐者列表、预测评分以及更新信任网络。

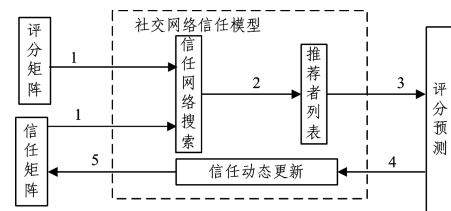


图 2 DTCF 框架

Fig. 2 Framework of DTCF

3.1 推荐者列表的获取

获取算法的描述如算法 1 所示。首先,以用户 u 为源节点对信任网络进行广度优先搜索。搜索过程中,当搜索到用户 v 时,如果 v 对项目 a 有评分记录且 u 对 v 的信任值大于或等于 0.5,则将其加入到推荐列表中;当用户 u 对用户 w 的信任值小于 0.5 时,则停止对 w 的所有子节点的搜索。

算法 1 推荐者列表的获取算法

Input:信任网络及用户对 Items 的评分矩阵 rate

Output:推荐者列表 recommenders

说明: $ts[u][v]$ 表示在当前路径下 u 对 v 的信任值, $tm[u][v]$ 表示所有已遍历过的 u 到 v 的路径中, u 对 v 的最大信任值。 $rate[u][a]$ 表示 u 对项目 a 的评分,若以前没有对 a 有过评分记录,则值为 -1。

Initialization:

以 u 为源节点对信任网络进行广度搜索

while 待遍历序列不为空 do

 对于搜索过程中遇到的所有节点 v :

 If $ts[u][v] < 0.5$ then

 不遍历 v 的子节点;

 end

 if $ts[u][v] \geq 0.5$ and $rate[v][a] \neq -1$ then

 if $v \notin recommenders$ then

 将 v 放入 recommenders 中;

 将 v 的子节点放入待遍历序列中;

 else

```

if tm[u][v] < ts[u][v] then
    tm[u][v] = ts[u][v];
    将 v 的子节点放入待遍历序列中;
end
end
end
end

```

如图 3 所示,左边为信任网络,右边为项目集合。用户间的边代表用户间的信任关系,用户到项目的有向边代表用户对此项目有评分记录。如果用户间的信任值都为 0.9,系统要预测 u_1 对 I_4 的评分,则获得的推荐者列表为 u_2 和 u_3 。如果用户间的信任值都为 0.6,系统要预测 u_1 对 I_4 的评分,由于此时 u_1 对 u_3 的信任值为 $0.6 * 0.6 = 0.36$ (不信任),则 u_1 获得的推荐者列表只有 u_2 。

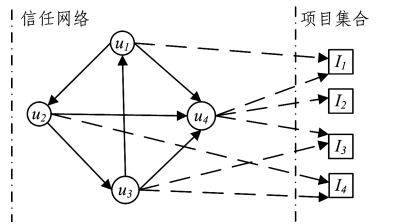


图 3 推荐者列表的获取原理

Fig. 3 Principle of recommender list acquisition

3.2 评分预测

考虑到用户 u 对推荐者列表中的用户拥有不同的信任值,评分的预测值为得到的推荐者列表对 a 的评分的平均值,权重为 u 对推荐者的信任值。信任值越高的用户给出的评分越可信,即:

$$P_{u,a} = \frac{\sum_{v \in \text{recommenders}} T_{u,v} R_{v,a}}{\sum_{v \in \text{recommenders}} T_{u,v}} \quad (2)$$

其中, $P_{u,a}$ 表示用户 u 对项目 a 的预测评分, recommenders 为获得的推荐者列表, $T_{u,v}$ 为用户 u 对用户 v 的信任值, $R_{v,a}$ 为用户 v 对项目 a 的评分。

3.3 信任网络的更新

本文采用一种带有奖惩机制的更新算法对信任网络进行更新。算法描述见算法 2,更新由 4 部分组成:判断用户 u 是否为冷启动用户,判断用户 v 是否为符合要求的推荐者,奖励符合要求的推荐者,以及惩罚不符合要求的推荐者。

1) 判断用户 u 是否为冷启动用户。如果 u 为冷启动用户,则不更新信任网络,结束此过程,否则继续步骤 2)。

2) 判断用户 v 是否为符合要求的推荐者。本文通过利用 threshold1 (评分阈值)及 threshold2 (相似度阈值)来判断某个用户是否为一个符合要求的推荐者。其中, threshold1 用来判断推荐者的本次推荐是否符合要求,当推荐者对项目 a 的评分与活跃用户对项目 a 的评分的差的绝对值小于 threshold1 时,认为本次推荐符合要求; threshold2 用来判断推荐者过去是否与活跃用户对项目的评分足够相似,当两者过去对项目评分的相似度大于 threshold2 时,认为推荐者过去的评分记录符合要求。若一个推荐者给出的推荐符合要求且与活跃用户过去对项目的评分足够相似,则认为该推荐者为符合要求的推荐者。

3) 奖励符合要求的推荐者。如果 v 是一个符合要求的推荐者,则需予以奖励。奖励方式为:若在信任网络中用户 u 和

用户 v 之间不存在边,则添加 u 到 v 的边并设置初始值为 0.5。若 u 和 v 之间存在边,则根据式(3)调整边的权重:

$$T_{i,j} = \begin{cases} 1, & \text{if } T_{i,j} + f(T_{i,j}) \geq 1 \\ T_{i,j} + f(T_{i,j}), & \text{if } T_{i,j} + f(T_{i,j}) < 1 \end{cases} \quad (3)$$

随着用户交互次数的增多,两者间的信任值会逐渐增大,但是增大的速度会逐渐减慢。因此式(3)中,函数 $f(x)$ 应该为一个在区间 $[0.5, 1]$ 中单调递减的函数。符合此要求的函数很多,本文采用以下函数:

$$f(x) = -0.8x^2 + 0.8x \quad (4)$$

4) 惩罚不符合要求的推荐者。惩罚只限于用户 u 和 v 之间存在边的情况。首先判断推荐者过去的评分记录是否符合要求,若不符合要求,则说明两用户的爱好差距很大,直接删除两者间的边;若过去的评分记录符合要求但本次推荐不符合要求,则说明两用户爱好相似,减小两用户间边的信任值即可,此时若边的权重小于 0.5,就删除该边。

算法 2 信任网络的更新算法

Input: 推荐者列表 recommenders , 评分矩阵 rate 及信任网络

Output: 调整过的信任网络

说明: $\text{ts}[u][v]$ 表示 u 对 v 的信任值, $\text{rate}[u][a]$ 表示 u 对项目 a 的评分,若以前没有对 a 有过评分记录,则值为 -1。Pearson(u, v) 为计算 u 与 v 相似度的方法。

```

for v ∈ recommenders do
    if u 不是冷启动用户 then
        if Pearson(u, v) > threshold2 then
            if u 与 v 对 a 的评分比较接近 then
                if u 与 v 之间没有边 then
                    在 u 与 v 间加边,设置初始值为 0.5;
                else
                    tr[u][r] = 1.8 * ts[u][r] - 0.8 * ts[u][r]2;
                end
            else
                if u 与 v 间有边 then
                    ts[u][r] = ts[u][r] - 0.3;
                    if ts[u][r] < 0.5 then
                        删除 u 与 v 的边;
                    end
                end
            end
        end
    end
end
else
    if u 与 v 间有边 then
        删除 u 与 v 的边;
    end
end
end
end
end
end

```

本文提出的 DTCF 算法利用信任网络能很好地解决冷启动问题。其次,对于非冷启动用户,该算法根据已有的信任关系获得一个推荐者列表集。根据存在“ a 信任 b , 则 a 与 b 相似度高的概率比较大”这一规律,推荐者列表中存在大量与活跃用户相似度较高的用户。通过奖惩机制修改现存的信任关系,经过多次迭代后,删除 u 的邻居节点集合中与 u 相似度较低的用户并新加入一些与 u 相似度较高的用户。因此,当用户再通过推荐系统获得推荐者列表时,推荐者列表往往是与 u 的相似度较高且 u 比较信任的用户,因此预测准确度较高。

算法的时间复杂度为 $O(k^{d_{\max}})$, 其中 k 为信任网络的平

均度数, d_{max} 为信任的最大传递距离。 $threshold2$ 作为控制加减边的阈值, 直接影响着 k 的大小, d_{max} 则由初始信任值决定。 $threshold2$ 越小、初始信任值越大, 则算法的复杂度越高。

4 实验与结果分析

本文采用包含信任网络数据和评分数据的 Epinions 数据集进行实验^[17]。该数据集从一个消费者评价网站提取, 包含 49290 个用户及用户间的信任关系、139738 种商品和 664824 条用户评论信息。其中有一半左右的冷启动用户(评分次数少于 5 次), 是一个非常稀疏的数据集, 因此 DTCF 算法对数据稀疏、冷启动问题的适应能力能够得到体现。

实验环境: WIN7, Intel Core i5-3470 CPU 3.20GHz, 4GB 内存。利用 Java 实现 DTCF, 同时实现经典的基于 Pearson 相关系数的协同过滤算法以及其他几种推荐算法。

评估标准: 采用 leave-one-out 思想, 将原有的用户评论信息分为 M 份(本文设置为 16 份), 隐藏其中一份数据信息, 以剩下的 $M-1$ 份数据为依据, 预测用户的评分信息并与隐藏的数据信息中的实际评分进行比较, 然后计算出平均绝对误差 MAE(Mean Absolute Error)及均方差 MSE(Mean Squared Error)。

$$MAE = \frac{1}{|E^p|} \sum_{(u,a) \in E^p} |r_{u,a} - pr_{u,a}| \quad (5)$$

$$MSE = \frac{1}{|E^p|} \sum_{(u,a) \in E^p} (r_{u,a} - pr_{u,a})^2 \quad (6)$$

其中, E^p 为隐藏数据, $r_{u,a}$ 表示用户 u 对项目 a 的评分, $pr_{u,a}$ 表示用户 u 对项目 a 的预测评分。

衡量一个推荐算法性能的另一个关键指标为覆盖率 (Coverage Ratio), 即用户能够通过推荐算法得到预测值的商品数与商品总量的比值。如果一个推荐算法的准确度很高, 但是能够预测的商品数量较少, 则此推荐算法的应用范围将受到限制。覆盖率 cov 的计算式为:

$$cov = \frac{\text{可推荐项目集合数量}}{\text{项目集合总数量}} \quad (7)$$

此外, 算法的运行效率也是衡量其性能的一个重要指标。本文通过预测的平均耗时来衡量算法的运行效率, 计算式为:

$$\text{平均预测时间} = \frac{\text{算法的运行时间}}{\text{测试集大小}} \quad (8)$$

实验步骤: 首先, 通过计算数据集 Epinions 信任网络中每个用户的从 1 到 6 跳节点的平均相似度来验证“ a 信任 b , 则 a 与 b 相似度高的概率比较大”这一规律。

不同信任程度的相似度对比结果如图 4 所示, 纵轴表示网络中每个节点的平均相似度之和(本文采用 Pearson 相关系数作为衡量用户间相似度的标准, 如果两个用户间的相似度无法计算, 则认为其相似度为 0), 横轴代表跳数。

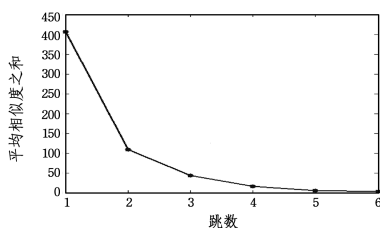


图 4 不同信任程度的相似度对比

Fig. 4 Similarity contrast of different trusted extent

从图 4 可以看出, 随着跳数的增加(也即信任度的下降), 用户间的相似度会逐渐降低, 即“ a 越信任 b , 则 a 与 b 相似度高的概率越大”。

接着, 通过改变信任网络更新的 $threshold2$ 值, 观察其对整个推荐算法性能的影响。为了更清楚地观察, 暂时设定初始信任值为 0.7, 实验结果如图 5 所示。

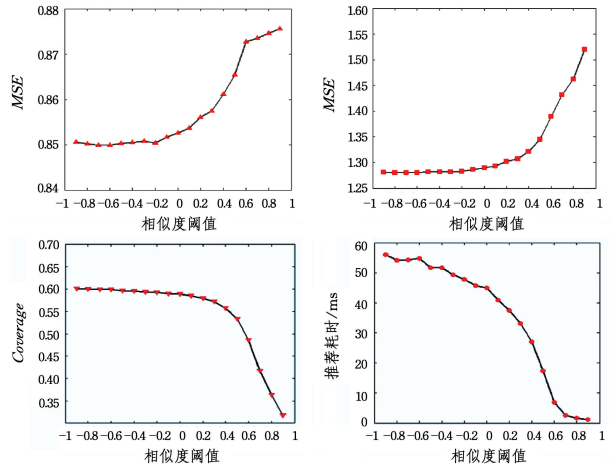


图 5 相似度阈值对推荐算法性能的影响

Fig. 5 Effect of similarity threshold on the performance of recommended algorithm

通过实验结果可以看出, 当 $threshold2 < 0$ 时, $threshold2$ 的减小对推荐系统的 MAE, MSE 及 Coverage 的影响不大, 但却会造成推荐时间的延长。这是因为当 $threshold2 < 0$ 时, 一些与活跃用户评分不相似的用户由于某次交互给出了符合要求的推荐而成为其邻居节点, 后续交互中会由于给出不符合要求的推荐而被剔除, 最终的邻居节点仍是与活跃用户相似度比较高的用户, 因此 MAE, Coverage, MSE 基本不变, 但由于反复增边、减边, 增加了推荐者获取的复杂度, 因此耗时会增加。当 $threshold2 > 0$ 时, 随着 $threshold2$ 的增大, 可以看到预测的准确度、覆盖率等会下降, 预测的耗时也在减少。这是因为增大 $threshold2$ 表示对邻居节点的要求变得更加严格, 信任网络会变得较稀疏, 此时用户获得的推荐者列表规模会相对较小, 因此预测的准确度、覆盖率以及预测的耗时都会下降。

本文通过改变初始信任值来观察其对推荐算法性能的影响, 如图 6 所示。

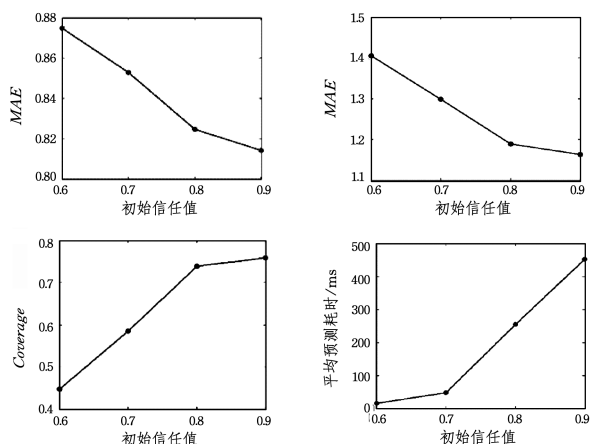


图 6 初始信任值对推荐算法的影响

Fig. 6 Influence of initial trust value on recommendation algorithm

从图6可以看到,随着初始信任值的增大,推荐系统的平均预测时间逐渐增大,MAE和MSE逐渐减小,覆盖率逐渐增大。这是因为随着初始信任值的增大,信任的传递距离会增大,每个用户的可信用户集合的规模会增大,因此平均预测时间和覆盖率都会相应增大,推荐者列表的规模也会相应增大,也即可以获得更多信任用户的推荐,因此MAE,MSE会逐渐减小。

最后,本文将DTCF与已有的经典推荐算法的进行对比,结果如表1所列。

表1 推荐算法的性能对比

Table 1 Performance comparison of recommendation algorithms

	MAE	MSE	覆盖率/%
CF	0.844	1.620	49.52
PMF	0.835	1.599	78.99
SocialMF	0.832	1.594	79.12
TidalTrust	0.830	1.590	80.03
TrustWalker	0.820	1.310	79.86
DTCF	0.814	1.163	75.87

可以看到,经典的CF算法过分依赖系统数据的稀疏程度,如果数据比较稀疏,其覆盖率极低,无法满足用户需求;其余几种基于信任网络的推荐算法在覆盖率方面有显著提升,但预测的准确度还有待提高。DTCF在初期利用了信任网络,有效地解决了覆盖率方面的问题;当用户在系统中的时间不断增长,算法的奖惩机制将发挥作用,原始的信任网络会发生改变,用户间存在的边不仅表示两者之间有信任关系,同时也表示两者之间的相似度比较高,因此预测的准确度也会提高。从表1可以看出,本文所提方法DTCF的MAE和MSE有一定程度的降低,也即评分预测更准确。由于考虑到推荐预测的耗时问题,DTCF在信任的传递距离方面做了一定的限制,导致覆盖率略低于TidalTrust(可以通过设置初始信任值的大小来提升覆盖率,但耗时也会相应增加)。

结束语 本文首先介绍了推荐系统领域的发展现状及目前流行的推荐算法;然后介绍了动态信任模型,包括信任的定义、传递、聚合等基本问题;接着提出了DTCF算法,将用户间的信任与相似度相关联,且随着用户间交互的不断进行,动态地调整信任网络;最后通过实验证明,DTCF算法在基本不影响覆盖率的情况下提高了推荐系统的预测准确度。利用信任网络虽然能够有效解决用户的“冷启动”问题,但却解决不了项目的“冷启动”问题。利用项目的固有属性来计算项目的相似度,能够解决项目的“冷启动”问题,因此未来可以通过将两种方法结合起来进一步提高DTCF的覆盖率。

参考文献

- [1] 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.
- [2] Collaborative filtering-wikipedia[EB/OL]. http://en.wikipedia.org/wiki/Collaborative_filtering.
- [3] GOLBECK J. Generating predictive movie recommendations from trust in social networks[C]// International Conference on Trust Management. Springer Berlin Heidelberg, 2006: 93-104.
- [4] PALAU J, MONTANER M, LÓPEZ B, et al. Collaboration analysis in recommender systems using social networks[C]// International Workshop on Cooperative Information Agents. Springer Berlin Heidelberg, 2004: 137-151.
- [5] YUAN W W, GUAN D H, LEE Y K, et al. Improved trust-aware recommender system using small-worldness of trust networks[J]. Knowledge-Based Systems, 2010, 23(3): 232-238.
- [6] TONG X R, ZHANG W, LONG Y. Transitivity of Agent Subjective Trust[J]. Journal of Software, 2012, 23(11): 2862-2870. (in Chinese)
童向荣, 张伟, 龙宇. Agent主观信任的传递性[J]. 软件学报, 2012, 23(11): 2862-2870.
- [7] HWANG C S, CHEN Y P. Using trust in collaborative filtering recommendation[C]// International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems. Springer Berlin Heidelberg, 2007: 1052-1060.
- [8] AVESANI P, MASSA P, TIELLA R. A trust-enhanced recommender system application: Moleskiing[C]// Proceedings of the 2005 ACM Symposium on Applied Computing. ACM, 2005: 1589-1593.
- [9] GUO G B, ZHANG J, THALMANN D. Merging trust in collaborative filtering to alleviate data sparsity and cold start[J]. Knowledge-Based Systems, 2014, 57: 57-68.
- [10] MORADI P, AHMADIAN S. A reliability-based recommendation method to improve trust-aware recommender systems[J]. Expert Systems with Applications, 2015, 42(21): 7386-7398.
- [11] JAMALI M, ESTER M. TrustWalker: a random walk model for combining trust-based and item-based recommendation[C]// 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2009: 397-406.
- [12] TONG Z, MCAULEY J, KING I. Leveraging social connections to improve personalized ranking for collaborative filtering[C]// 23rd ACM International Conference on Conference on Information and Knowledge Management. ACM, 2014: 261-270.
- [13] AZADJALAL M M, MORADI P, ABDOLLAHPOURI A. Application of game theory techniques for improving trust based recommender systems in social networks[C]// 2014 4th International Conference on Computer and Knowledge Engineering (ICCKE). IEEE, 2014: 261-266.
- [14] FENG J Y. Research on Trust Management Technologies in Open Peer-to-Peer Environment[D]. Xi'an: Xidian University, 2011. (in Chinese)
冯景瑜. 开放式P2P网络环境下的信任管理技术研究[D]. 西安: 西安电子科技大学, 2011.
- [15] ZHANG M W, YANG B, YU Y. DS theory based Distributed trust model[J]. Journal of Wuhan University, 2009, 55(1): 41-44. (in Chinese)
张明武, 杨波, 禹勇. 基于DS理论的分布式信任模型[J]. 武汉大学学报, 2009, 55(1): 41-44.
- [16] HU X P, YIN J. Research on Trust Transfer Model [J]. Journal of Southeast University (Philosophy and Social Science), 2013 (4): 46-51. (in Chinese)
胡祥培, 尹进. 信任传递模型研究综述[J]. 东南大学学报(哲学社会科学版), 2013(4): 46-51.
- [17] http://www.trustlet.org/wiki/Epinions_datasets.