

基于社交信任聚类的混合推荐算法*

朱敬华^{1,2}, 王超¹, 马胜超¹

¹(黑龙江大学 计算机科学与技术学院, 黑龙江 哈尔滨 150080)

²(黑龙江省数据库与并行计算重点实验室, 黑龙江 哈尔滨 150080)

通讯作者: 朱敬华, E-mail: zhujinghua@hlju.edu.cn



摘要: 推荐系统能够有效地解决信息过载问题,其中,协同过滤(collaborative filtering,简称 CF)是推荐系统广泛采用的技术之一.然而传统的 CF 技术存在可扩展性差、数据稀疏和推荐结果精度低等问题.为了提高推荐质量,将信任关系融合到推荐系统中,采用聚类(FCM)方法,对信任关系进行聚类.利用信任类预测用户间的隐性信任,最后将信任关系与用户-项目关系线性融合进行推荐.在 Douban 和 Epinions 数据集上的实验结果表明,与传统的基于 CF、基于信任和用户项目聚类的推荐算法相比,该算法能够大幅度地改进推荐质量,提升算法的时间效率.

关键词: 社交信任;聚类;协同过滤;数据稀疏;推荐

中文引用格式: 朱敬华,王超,马胜超.基于社交信任聚类的混合推荐算法.软件学报,2018,29(Suppl.(1)):21-31. <http://www.jos.org.cn/1000-9825/18003.htm>

英文引用格式: Zhu JH, Wang C, Ma SC. Hybrid recommendation algorithm based on social trust clustering. Ruan Jian Xue Bao/Journal of Software, 2018,29(Suppl.(1)):21-31 (in Chinese). <http://www.jos.org.cn/1000-9825/18003.htm>

Hybrid Recommendation Algorithm Based on Social Trust Clustering

ZHU Jing-Hua^{1,2}, WANG Chao¹, MA Sheng-Chao¹

¹(School of Computer Science and Technology, Heilongjiang University, Harbin 150080, China)

²(Heilongjiang Province Key Laboratory for Database and Parallel Computing, Harbin 150080, China)

Abstract: Recommender system can solve the information overload problem effectively, and collaborative filtering (CF) is one of the techniques that is widely used in recommendation system. However, the traditional CF technology has problems such as poor scalability, sparse data, and low accuracy of recommendation results. In order to improve the quality of recommendations, this article integrates the trust relationship into the recommendation system in which the trust relationship is clustered by using the clustering (FCM) method. Using the trust cluster to predict implicit trust between users, the trust relationship is finally combined with the user-item relationship to give recommendations. The experimental results on the data set of Douban and Epinions show that compared with traditional CF algorithm, trust based recommendation algorithm and recommendation algorithm for user item clustering, the presented algorithm can greatly improve the recommendation quality and time efficiency.

Key words: trust relationship; clustering; collaborative filtering; data sparseness; recommendation

随着移动互联网技术的快速发展,推荐系统的应用越来越广泛,成为解决信息过载的重要技术.推荐系统是一个个性化系统,它能够根据用户的浏览记录、用户评分等历史数据,挖掘用户偏好,帮助用户从大量数据中筛选出感兴趣的文章、电影、音乐等信息.目前广泛应用的推荐算法是协同过滤推荐算法 CF(collaborative filtering)^[1].然而,协同过滤推荐算法也有其不足之处:(1) 数据稀疏.例如,一个包含海量商品的电商平台,由于用

* 基金项目: 国家自然科学基金(6110048); 黑龙江省自然科学基金(F2016034)

Foundation item: National Natural Science Foundation of China (6110048); Natural Science Foundation of Heilongjiang Province, China (F2016034)

收稿时间: 2018-05-01; 采用时间: 2018-08-30

户购买的商品只是所有商品的一小部分,不同用户购买商品的重叠性较低,导致具有相似偏好的用户数量较少.(2) 可扩展性差.随着用户和项目数量的成倍增加,算法通常无法在线性时间内给出推荐结果.(3) 冷启动问题.当注册一个新用户时,由于没有任何关于该用户的历史记录,从而无法进行个性化推荐.

由于传统协同过滤算法存在的一些不足,人们就考虑将社交信任融入推荐算法中^[2].现有的基于信任的推荐系算法一般只能利用显式的用户信任关系,很多隐式的用户信任关系没有被挖掘出来.本文提出通过 FCM 聚类方法^[16],对用户信任关系进行聚类分析,能够挖掘用户之间的隐式关系.利用聚类结果来预测用户之间的信任值,最后将信任值和用户相似性线性融合,进行评分预测,从而给出推荐.

本文可能的贡献如下:

(1) 本文提出了一种基于社交信任聚类的混合推荐算法,将社交信任融入聚类算法中,能够提高算法的准确性和可扩展性.

(2) 利用 FCM 聚类结果进行信任值计算,可以得到所有用户间的信任值.这样,用户之间的信任关系变成精确的数值,信任关系更加直观和准确.

(3) 通过仿真实验验证了所提出算法的推荐性能.

1 相关工作

1.1 传统推荐技术的改进

由于 CF 推荐算法存在推荐准确性低和数据稀疏性等问题,很多学者对其进行了改进.如 Heckel 等人^[1]提出将负面评价引入 CF 推荐算法中,与正面评价相结合共同给出推荐,提高了算法的准确性.Wang 等人^[3]提出一种协同深度学习推荐算法,解决了数据稀疏性等问题.Jing 等人^[4]提出了一种稀疏的概率矩阵分解方法,可以用来解决长尾分布问题.

1.2 社交信任推荐技术

传统推荐算法一般只考虑用户的兴趣,往往并不能获得良好的推荐效果.于是,很多学者提出将社交信任引入推荐算法中,用来改善算法的推荐性能.如陈婷等人^[5]提出了基于信任关系计算的推荐算法,将用户相似性和信任关系进行综合计算.也有学者在矩阵分解模型中融入了社交信任关系.Guo 等人^[2]提出了一种信任关系的矩阵分解模型,考虑了用户-项目关系与用户信任关系对推荐结果的显性和隐性影响.Chaney 等人^[6]设计了一种 SPF 模型,在传统分解方法中引入社交网络信息,目标是缩小基于用户偏好和基于社交关系推荐的差距.

1.3 聚类推荐技术

聚类的应用很广泛,有学者将聚类技术应用到社区检测中.如 Kakkar^[7]等人提出了一种利用联合聚类的方法,能够在社区网络中检测出重叠的社区.Jin 等人^[8]提出了一种在复杂的动态网络中确定社区结构的聚类算法.也有学者将聚类技术应用到朋友推荐中,如 Huang 等人^[9]提出了基于网络相关性和特征聚类的好友推荐算法,将用户、标签和图像特性组合在一起,使用聚类方式能够使推荐结果更准确.

也有学者在传统推荐算法中引入聚类方法^[10-12],能够有效地解决传统推荐算法普遍存在的问题.如 Du 等人^[13]将社交影响力加入到聚类算法中,用以解决冷启动问题.有些学者利用聚类相关的技术对 CF 算法加以改进,如 Wu 等人^[14]提出一种用户项目聚类的可扩展方法来改进传统的推荐算法.然而,现有的大多数方法只考虑了对用户-项目的聚类,并没有考虑对用户之间的信任进行聚类.Dubois 等人^[15]将用户间的信任关系进行聚类,并将聚类结果融入 CF 算法中进行推荐.然而,Dubois 等人所使用的信任值是数据集中直接给出的,并没有进行更准确的信任值计算.本文将用户间的关注关系进行聚类,并利用聚类结果进行信任值计算,将信任值与 CF 算法线性结合,具有更准确的推荐效果.

2 基于社交信任聚类的混合推荐

2.1 推荐算法框架

算法主要有以下几个步骤.首先根据用户-项目评分矩阵,计算用户之间的相似性,将每个用户相似度最高的前 k 个用户,放入集合 S_U 中.同时根据用户之间的关注关系,对用户关系矩阵进行聚类,得到用户与类的隶属度矩阵.根据隶属度矩阵进行信任值计算,计算所有用户之间的信任值.将每个用户信任值最高的前 k 个用户,放入集合 B_U 中.最后将用户相似性和信任值线性结合得出推荐.本文算法的主要流程如图 1 所示.

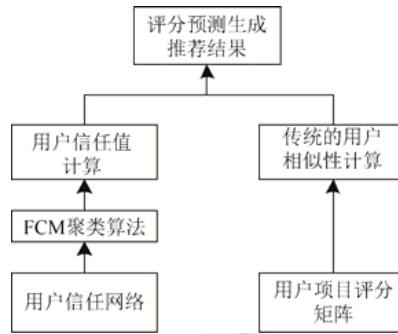


Fig.1 Algorithm flowchart

图 1 算法流程图

2.2 用户关系数据初始化

假设给出了 n 个用户之间的关注信息,并将其生成用户关系矩阵 $R^{n \times n}$.如果用户 u_i 关注了用户 u_j ,则令 $x_{ij}=1$.如果用户 u_i 未关注用户 u_j ,则令 $x_{ij}=0$.假设用户 A 关注了用户 B ,称 A 信任 B ,反之则是 B 被 A 信任.矩阵 $R^{n \times n}$ 行与列的关系就是信任与被信任关系.此时,矩阵中的行用户和列用户分别表示信任用户和被信任用户.图 2 所示为初始给出的用户关系网络(实线表示直接关注关系,虚线表示传递关系).

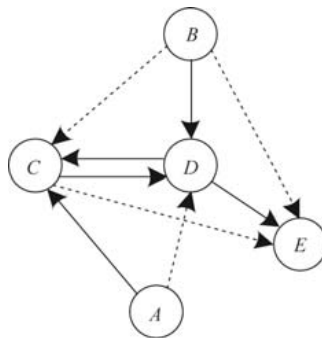


Fig.2 User relationship diagram

图 2 用户关系图

根据用户关系网络,可以生成用户关系矩阵 $R^{5 \times 5}$.

$$R^{5 \times 5} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

由于原始矩阵是比较稀疏的,为了解决矩阵稀疏性问题,假设在用户关系的传递路径中,将每个顶点到其他顶点的路径长度不超过 2 的顶点,作为起始顶点的关注用户。

例如,图 2 中有一条传递路径 A-C-D-E,除了 C 作为 A 的信用户外,也可以将 D 作为 A 的信用户,此时, $x_{AD}=1$ (图 2 中用虚线表示)。而 A 到 E 的路径长度超过 2,所以 E 不作为 A 的关注用户。

假设初始矩阵为 $M_R(M_R=R^{n \times n})$,只要求出 1,2,3 阶传递关系,并在每阶传递关系求完后与上一阶联合再进行下一阶的求解。具体操作为:首先对原始矩阵 M_R 进行布尔幂运算得到 $M_R^{[2]}$,再进行布尔幂运算得到 $M_R^{[3]}$,最后将这 3 个矩阵取并集。

$$M_{R^*} = M_R \vee M_R^{[2]} \vee M_R^{[3]},$$

M_{R^*} 是最终结果。最终得到的关系矩阵为

$$R^{5 \times 5} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}.$$

本文的目的就是将矩阵 $R^{n \times n}$ 分成 c 个类。其中,信用户属于多个类,被信用户也可以属于多个类,并且一个类中包含至少一个信用户和一个被信用户(信用户和被信用户不能是同一个用户)。

2.3 用户相似性计算

给出包含 n 个用户对 m 个项目的评分数据。数据中包括部分用户对部分项目的评分,首先将用户对项目评分数据生成用户-项目评分矩阵 $Y^{n \times m}$ 。

$$Y^{n \times m} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1m} \\ y_{21} & y_{22} & \cdots & y_{2m} \\ \vdots & \vdots & & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{nm} \end{bmatrix}.$$

y_{ij} 表示用户 u_i 对项目 r_j 的评分,根据用户-项目评分矩阵,本文使用修正余弦相似度(adjusted cosine similarity)来计算用户间的兴趣相似性。

$$\text{sim}(u_i, u_j) = \frac{\sum_{r \in I} (y_{i,r} - \bar{y}_i)(y_{j,r} - \bar{y}_j)}{\sqrt{\sum_{r \in I} (y_{i,r} - \bar{y}_i)^2} \sqrt{\sum_{r \in I} (y_{j,r} - \bar{y}_j)^2}} \quad (1)$$

$y_{i,r}$ 表示用户 u_i 对项目 r 的评分, I 表示项目的集合,其中, $r \in I$, \bar{y}_i, \bar{y}_j 分别表示用户 u_i 和 u_j 对所有已评分项目评分的平均值。当完成用户相似性计算后,把每个用户相似性最高的前 k 个用户,放入集合 S_U 中。

2.4 信任关系聚类

信任关系聚类结果得到的是用户与类之间的隶属度矩阵 $W^{n \times c}$,其中, $W_{ij} \in [0,1], i \in (1,2,3, \dots, n), j \in (1,2,3, \dots, c), W_{ij}=0$ 表示用户 i 不属于某个类 j 。当 $W_{ij} \neq 0$ 时, W_{ij} 表示用户 i 与类 j 的关联程度。

为了减少数据稀疏性对算法的影响,提升算法的可扩展性,本文使用了模糊 c 均值聚类(fuzzy c -means,简称 FCM)算法^[16]。该算法使用隶属度确定用户与各个类的关联程度。FCM 的目标是最小化全局代价函数 J 。如公式(2)所示。

$$J(W_{ij}, c_j) = \sum_{j=1}^c \sum_{i=1}^n (W_{ij})^p d(x_i, c_j)^2 \quad (2)$$

其中, W_{ij} 表示用户 x_i 与类 c_j 的隶属度, p 是自由参数,它能够控制类与类之间的混合程度。当 $p=0$ 时,每个用户只属于一个类。当 $p>0$ 时,则允许每个用户属于多个类,这里取 $p=2$ 。 $d(x_i, c_j)$ 表示第 i 个用户与第 j 个聚类中心的欧几里德距离。FCM 的目的就是将用户关系矩阵 $R^{n \times n}$ 分成 c 个模糊类,求出能够使得全局代价函数最小的聚类中心。

$$\sum_{j=1}^c W_{ij} = 1 \tag{3}$$

在 FCM 初试化时,用(0,1)之间的随机数初试化矩阵隶属度 W_{ij} ,并且每个用户与所有类的隶属度之和为 1. 而使得全局代价函数最小的必要条件为

$$c_j = \frac{\sum_{i=1}^{n+m} (W_{ij})^p x_i}{\sum_{i=1}^n (W_{ij})^p} \tag{4}$$

和

$$W_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d(x_i, c_j)}{d(x_i, c_k)} \right)^{2/(p-1)}} \tag{5}$$

FCM 是一个迭代过程,对于所有的 i 和 $j, i \in (1, 2, 3, \dots, n), j \in (1, 2, 3, \dots, c)$, 由于初始的聚类中心不同,全局代价函数可能不同,所以在每次迭代时都使用不同的聚类中心来初始化算法.当 J 相对于上次迭代的数值改变量小于某个阈值 q 时,算法停止.

聚类结果示意图.如图 3 所示,灰色区域表示在信任网络中有信任关系的用户,白色区域则表示在信任网络中没有信任关系的用户.第 2.5 节将会介绍如何计算信任值.首先对聚类结果所得到的用户-类隶属度矩阵进行迭代更新,求出最优的用户-类隶属度矩阵.再利用最优隶属度矩阵计算出有信任关系和没有信任关系用户之间的信任值.图 3 中两个方框所包含的用户是两个不同的类,各个类之间是可重叠的,一个用户可同时属于多个类.

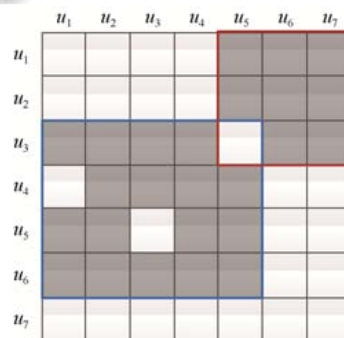


Fig.3 Clustering diagram

图 3 聚类示意图

FCM 算法的主要步骤见表 1.

Table 1 Fuzzy-C-Means

表 1 模糊 c 均值聚类

Algorithm. Fuzzy-C-Means.	
input:	User-Relationship matrix、 Threshold q 、 Number of clusters c 、 Free parameters p ;
output:	The affiliation matrix of the user and cluster.
1. initialize:	$t=1$, random generate W^0
2. Repeat	
3.	Calculate c cluster centers $C_j, j \in (1, 2, \dots, c)$;
4. For	$(j=1, j <= c, j++)$
5.	Compute c_j^t ;
6. End for	
7. For	$(i=1, i <= n, i++)$
8. For	$(j=1, j <= c, j++)$
9.	Compute Euclidean distance $d(x_i, c_j^t)$;
10.	Compute W_{ij}^t ;
11. End for	
12. End for	
13.	Compute cost function J^t ;
14.	$t++$;
15. Until	$ J^{t-1} - J^{t-2} \leq q$

2.5 信任值预测

对于给定的用户关系矩阵 $R^{n \times n}$,假设 U 是用户集合, $U = \{u_1, u_2, u_3, \dots, u_n\}$.第 2.4 节使用 FCM 聚类方法可以获

$$K(u_i, r_q) = \alpha \frac{\sum_{u_j \in B_U} (T_{ij}) \times (y_{jq} - \bar{y}_j)}{\sum_{u_j \in S_U} (T_{ij})} + (1 - \alpha) \frac{\sum_{u_j \in S_U} \text{sim}(u_i, u_j) \times (y_{jq} - \bar{y}_j)}{\sum_{u_j \in S_U} \text{sim}(u_i, u_j)} + \bar{y}_i \quad (12)$$

$K(u_i, r_q)$ 表示用户 u_i 对未评分项目 r_q 的预测评分, $i \in (1, 2, 3, \dots, n)$, $q \in (1, 2, 3, \dots, m)$. α 表示用户信任值和相似度所占比重的权值, $\alpha \in [0, 1]$, $\alpha=1$ 表示评分预测只和信任值有关, $\alpha=0$ 表示评分预测只和相似度有关. B_U 表示各个用户信任值最高的前 k 个用户集合, S_U 是每个用户兴趣相似度最高的前 k 个用户集合. y_{jq} 表示与 u_i 相似的用户 u_j 对项目 r_q 的评分, \bar{y}_j 表示 u_j 对所有已评分项目评分的平均值, \bar{y}_i 也类似. 公式(12)可以求出用户对未评分项目的预测评分, 依据预测评分, 给出推荐结果.

3 实验与结果分析

本节主要通过实验来验证之前假设, 并进行了对比分析.

(1) 分析聚类数量 c 和权重 α 对算法性能的影响, 找出使得推荐性能最优的 c 和 α 的取值.

(2) 与基于用户的(user-based)协同过滤推荐算法、基于用户信任的(user-trust)推荐算法和对用户项目聚类的推荐算法进行了实验对比分析.

3.1 实验数据

本文使用 Douban 和 Epinions 数据集进行实验. 数据集中主要有用户对部分项目的评分以及与朋友的关系. 首先对数据预处理, 去除好友数量少于 3、对项目评分次数少于 5 的用户. 预处理后的 Douban 数据中包含 23 367 个用户对 4 580 部电影的评分, 共有 176 452 条评分信息以及 158 135 条用户关系信息. Epinions 数据集主要包含 401 230 个用户对 125 160 件物品的评分, 共有 504 645 条评分信息以及 385 230 条用户关系信息. 实验中将数据集分为 5 部分, 使用 5 折交叉验证方式, 每次将其中 4 部分用于训练, 剩下的一部分用于测试, 将 5 次实验的平均值作为实验结果.

3.2 评价标准

实验中本文使用 4 种经典评价指标: Precision、Recall、 $F1$ 值和平均误差 MAE(mean absolute error). 假设为用户推荐了 N 部电影, 用户观看了 L 部, 则精确率 $\text{Precision}@N=L/N$. 假设用户总共观看了 M 部电影, 这些电影中有些是推荐的, 有些不是, 则召回率 $\text{Recall}@N=L/M$. 本文使用 $F1$ 值对精确率和召回率进行综合评估.

$$F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (13)$$

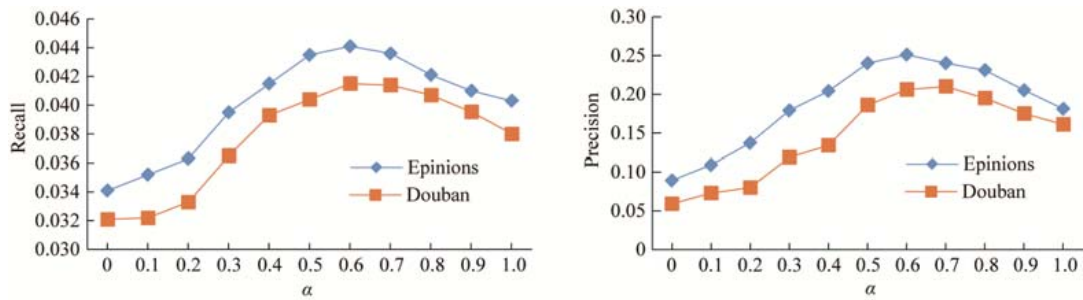
MAE 是评估算法推荐质量的一种标准. 它通过计算实际分值和预测分值的差异来衡量推荐是否准确.

$$MAE = \frac{\sum_{i \in U, j \in I} |p_{ij} - q_{ij}|}{N} \quad (14)$$

p_{ij} 表示用户 u_i 对项目 r_j 的实际评分, q_{ij} 表示用户 u_i 对项目 r_j 的预测评分, N 表示所有评分的数量.

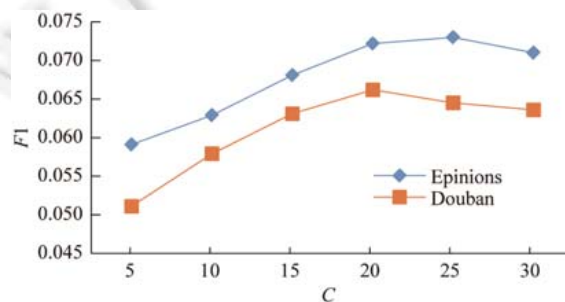
3.3 参数的选择

(1) 权重 α 的选择. 权重 α 是控制评分预测中信任值和用户相似度所占比重, $0 \leq \alpha \leq 1$. 当 α 为 0 时表示算法只基于用户兴趣相似度进行推荐, α 为 1 时表示算法只基于用户信任进行推荐. 为了寻找使得算法推荐性能最好的 α , 本文在 Douban 和 Epinions 数据集中使用 Recall 和 Precision 两种评价指标分别对 α 取值为 0~1 之间的算法性能进行评估, 实验结果如图 5 所示.

Fig.5 The effect of weight α on recall rate and precision图 5 α 的取值对召回率和精确率的影响

从图 5 中可以看出,开始时随着 α 的增大,召回率和精确率也在递增,当 α 取值在[0.5,0.7]区间时召回率和精确率都达到了最大.所以,这里 α 取 0.65.在召回率和精确率结果中都出现了拐点,因为在公式(12)中 α 代表用户信任关系在评分预测中所占比重.随着 α 的逐渐增大,信任关系对评分预测的影响也越来越大,这也说明用户更愿意采纳依据信任关系进行的推荐,当 α 取值在[0.6,0.7]之间时,算法性能最好.当 α 大于 0.7 时,算法性能有一定程度的下降,说明只依赖信任好友的推荐并不能取得最好的推荐效果,要将用户相似性和信任关系相互结合进行推荐.

(2) 聚类个数 c 的选择.由于不同数据集中聚类个数对推荐性能的影响有所不同.本文在使用 $F1$ 值评价标准时分别在 Douban 数据集和 Epinions 数据集进行实验,目的是要找出使得算法在各个数据集上推荐性能最好的 c 的取值.

Fig.6 The effect of the number of clusters c on the $F1$ value图 6 聚类个数 c 对 $F1$ 值的影响

由实验结果可知,在 Douban 数据集,当 c 的数量在[5,20]之间时, $F1$ 值逐渐增大.当 c 的数量在 20 时, $F1$ 值最大.当 c 的数量大于 20 时, $F1$ 值的大小趋于平稳.说明当 c 的数量达到某个值时, c 的增加并不能更好地提高推荐性能,反而使推荐性能有一定程度的下降.说明随着聚类数 c 的增加,一些类中只有少量的用户,并不能准确地对用户进行分类,就会导致推荐性能的下降.所以,在 Douban 数据集进行实验时, c 取 20 左右,算法能达到最佳性能.而在 Epinions 数据集进行实验时, c 取 25 左右,算法能达到最佳性能.

3.4 算法推荐性能对比

接下来通过实验验证本文提出的基于社交信任聚类的混合推荐算法(TCoC)的性能,并与下列 3 种推荐算法进行对比实验.(1) User-based CF:基于用户的协同过滤推荐算法.(2) User-trust 算法:传统的基于信任的推荐算法.(3) SCC 算法:使用聚类技术对用户-项目评分矩阵聚类的推荐算法^[17].

在 Douban 和 Epinions 数据集上,使用 $F1$ 和 MAE 评价指标,分别在推荐项目个数 N 为 5、10、20、40 的情况下,对以上 4 种算法的推荐性能进行了对比实验.两个数据集上的实验结果如图 7 和图 8 所示.

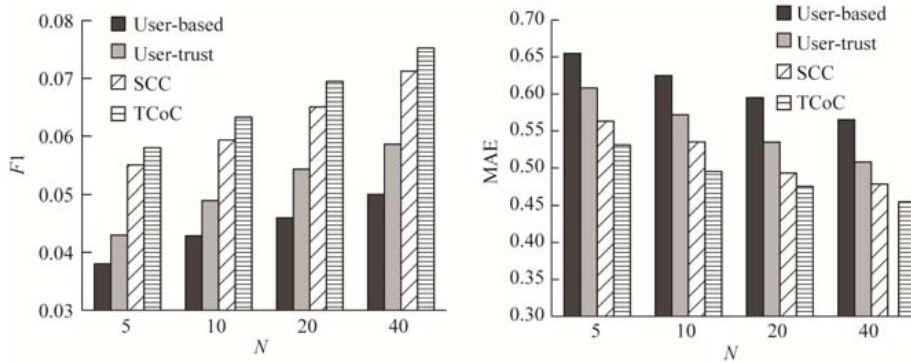


Fig.7 Comparison of $F1$ values and mean absolute deviation of the four algorithms in Epinions dataset
图7 Epinions 数据集上4种算法的 $F1$ 值和 MAE 对比

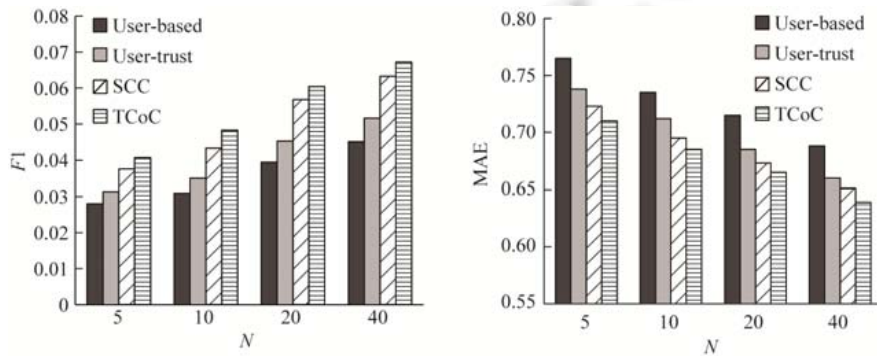


Fig.8 Comparison of $F1$ values and mean absolute deviation of the four algorithms in Douban dataset
图8 Douban 数据集上4种算法的 $F1$ 值和 MAE 对比

由图7和图8可知,随着 N 值的增大,TCoC 算法的精确率虽有一定程度的下降,但 TCoC 算法在2种评价指标下的推荐性能都表现最佳.实验结果也表明,根据用户信任进行聚类再结合用户相似性进行推荐算法,与 User-trust 算法、User-based、SCC 算法相比推荐效果更好.

本文在 Douban 数据集上进行了实验,在将推荐个数设置为5、10、20、40时,对 TCoC 算法和 User-trust、User-based、SCC 算法的运行时间进行了对比.如图9所示.

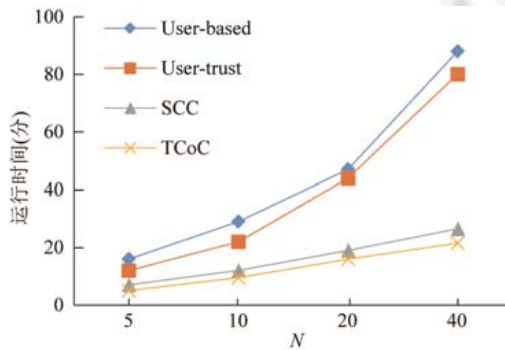


Fig.9 Comparison of running time of the four algorithms
图9 4种算法运行时间对比

从运行时间的对比实验结果中可以看出, User-trust 和 User-based 算法的运行时间较长, 并且随着推荐数量增加, 算法运行时间呈指数型增长. TCoC 和 SCC 算法的运行时间远少于前两种算法, 大致相等, 随着推荐数量的增加, 算法运行时间呈线性增长. 而 TCoC 算法推荐效果要好于 SCC, 所以 TCoC 算法的整体性能更好.

4 结论和未来工作

传统的基于用户和项目的协同过滤算法存在数据稀疏和冷启动问题, 融合社交信任能够有效地解决这一问题, 提高了推荐质量. 现有的基于信任的推荐算法重点研究直接信任的预测和间接信任的传递, 但时间开销比较大, 不适合大规模社交网络. 本文提出一种基于社交信任聚类的混合推荐算法, 将信任关系聚类计算与用户相似性计算相结合, 在真实数据集上的实验结果表明, 本文方法提高了推荐结果的质量, 并降低了推荐算法的时间, 算法具有较好的可扩展性.

未来, 将研究结合用户之间不信任关系辅助推荐, 以更好地改善算法的推荐质量.

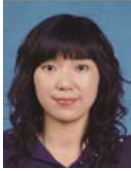
References:

- [1] Heckel R, Ramchandran K. The sample complexity of online one-class collaborative filtering. In: Proc. of the Int'l Conf. on Machine Learning. 2017. 1452–1460.
- [2] Guo G, Zhang J, Yorke-Smith N. TrustSVD: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In: Proc. of the 29th AAAI Conf. on Artificial Intelligence. AAAI Press, 2015. 123–129.
- [3] Wang H, Wang N, Yeung DY. Collaborative deep learning for recommender systems. In: Proc. of the KDD. 2014. 1235–1244. [doi: 10.1145/2783258.2783273]
- [4] Jing L, Wang P, Yang L. Sparse probabilistic matrix factorization by Laplace distribution for collaborative filtering. In: Proc. of the Int'l Conf. on Artificial Intelligence. AAAI Press, 2015. 1771–1777.
- [5] Chen T, Zhu Q, Zhou MX, Wang S. Trust-Based recommendation algorithm in social network. Ruan Jian Xue Bao/Journal of Software, 2017, 28(3):721–731 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/5159.htm> [doi: 10.13328/j.cnki.jos.005159]
- [6] Chaney AJB, Blei DM, Eliassi-Rad T. A probabilistic model for using social networks in personalized item recommendation. In: Proc. of the ACM Conf. on Recommender Systems. ACM, 2015. 43–50. [doi: 10.1145/2792838.2800193]
- [7] Kakkar S, Beniwal S. Discovering overlapping community structure in networks through co-clustering. In: Proc. of the Int'l Conf. on Inventive Computation Technologies. Coimbatore: IEEE, 2017. 1296–1300. [doi: 10.13140/RG.2.2.28132.17282]
- [8] Jin H, Yu W, Li SJ. A clustering algorithm for determining community structure in complex networks. Physica A Statistical Mechanics & Its Applications, 2018, 492. [doi: 10.1016/j.physa.2017.11.029]
- [9] Huang S, Zhang J, Lu S, Hua XS. Social friend recommendation based on network correlation and feature co-clustering. In: Proc. of the ICMR 2015, the 5th ACM on Int'l Conf. on Multimedia Retrieval. Shanghai: ICMR, 2015, 18(2):315–322. [doi: 10.1145/2671188.2749325]
- [10] Heckel R, Vlachos M, Parnell T, Duenner C. Scalable and interpretable product recommendations via overlapping co-clustering. In: Proc. of the IEEE Int'l Conf. on Data Engineering. San Diego: IEEE, 2017. 1033–1044. [doi: 10.1109/ICDE.2017.149]
- [11] Xu B, Bu J, Chen C, Cai D. An exploration of improving collaborative recommender systems via user-item subgroups. In: Proc. of the WWW. 2012. 21–30. [doi: 10.1145/2187836.2187840]
- [12] Liang CY, Leng YJ. Collaborative filtering based on information-theoretic co-clustering. Proc. of the Int'l Journal of Systems Science, 2014, 45(3):589–597. [doi: 10.1080/00207721.2012.724109]
- [13] Du X, Liu H, Jing L. Additive co-clustering with social influence for recommendation. In: Proc. of the 11th ACM Conf. ACM, 2017. 193–200. [doi: 10.1145/3109859.3109883]
- [14] Wu Y, Liu X, Xie M, Ester M, Yang Q. CCCF: Improving collaborative filtering via scalable user-item co-clustering. In: Proc. of the WSDM 2016, the 9th ACM Int'l Conf. on Web Search and Data Mining. ACM, 2016. 73–82. [doi: 10.1145/2835776.2835836]
- [15] Dubois T, Golbeck J, Kleint J, Srinivasan A. Improving recommendation accuracy by clustering social networks with trust. In: Proc. of the ACM RecSys 2009 Workshop on Recommender Systems & the Social Web. 2009.

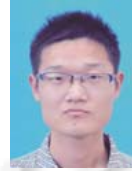
- [16] Bezdek JC, Ehrlich R, Full W. FCM: The fuzzy *c*-means clustering algorithm. *Computers & Geosciences*, 1984,10(2):191–203. [doi: 10.1016/0098-3004(84)90020-7]
- [17] Liao CL, Lee SJ. A clustering based approach to improving the efficiency of collaborative filtering recommendation. *Electronic Commerce Research & Applications*, 2016,18:1–9. [doi: 10.1016/j.elerap.2016.05.001]

附中文参考文献:

- [5] 陈婷,朱青,周梦溪,王珊.社交网络环境下基于信任的推荐算法. *软件学报*,2017,28(3):721–731. <http://www.jos.org.cn/1000-9825/5159.htm> [doi: 10.13328/j.cnki.jos.005159]



朱敬华(1976—),女,黑龙江齐齐哈尔人,博士,教授,CCF 专业会员,主要研究领域为推荐系统社交网挖掘,无线传感网数据管理,不确定数据挖掘.



马胜超(1993—),男,硕士,主要研究领域为推荐系统.



王超(1993—),男,硕士,主要研究领域为数据挖掘.

www.jos.org.cn

www.jos.org.cn