计算机研究与发展
Journal of Computer Research and Development
ISSN 1000-1239/CN 11-1777/TP
50(5): 1076-1084, 2013

基于双重邻居选取策略的协同过滤推荐算法

贾冬艳　　张付志
(燕山大学信息科学与工程学院　河北秦皇岛　066004)
(xjzfz@ysu.edu.cn)

A Collaborative Filtering Recommendation Algorithm Based on Double Neighbor Choosing Strategy

Jia Dongyan and Zhang Fuzhi
(School of Information Science and Engineering, Yanshan University, Qinhuangdao, Hebei 066004)

Abstract Collaborative filtering is the most successful and widely used recommendation technology in E-commerce recommender system. It can recommend products for users by collecting the preference information of similar users. However, the traditional collaborative filtering recommendation algorithms have the disadvantages of lower recommendation precision and weaker capability of attack-resistance. In order to solve the problems, a collaborative filtering recommendation algorithm based on double neighbor choosing strategy is proposed. Firstly, on the basis of the computational result of user similarity, the preference similar users of target user are chosen dynamically. Then the trust computing model is designed to measure the trust relation between users according to the ratings of similar users. The trustworthy neighbor set of target user is selected in accordance with the degree of trust between users. Finally, a novel collaborative filtering recommendation algorithm based on the double neighbor choosing strategy is designed to generate recommendation for the target user. Using the MovieLens and Netflix dataset, the performance of the novel algorithm is compared with that of others from both sides of recommendation precision and the capability of attack-resistance. Experimental results show that compared with the existing algorithms, the proposed algorithm not only improves the recommendation precision, but also resists the malicious users effectively.

Key words double neighbor choosing strategy; preference similar user; trust computing model; collaborative filtering; recommender system; similarity

摘　要　协同过滤是电子商务推荐系统中应用最成功的推荐技术之一,但是传统的协同过滤推荐算法存在推荐精度低和抗攻击能力差的缺陷.针对这些问题,提出了一种基于双重邻居选取策略的协同过滤推荐算法.首先基于用户相似度计算的结果,动态选取目标用户的兴趣相似用户集.然后提出了一种用户信任计算模型,根据用户的评分信息,计算得到目标用户对兴趣相似用户的信任度,并以此作为选取可信邻居用户的依据.最后,利用双重邻居选取策略,完成对目标用户的推荐.实验结果表明该算法不仅提高了系统推荐精度,而且具有较强的抗攻击能力.

关键词　双重邻居选取策略;兴趣相似用户;信任计算模型;协同过滤;推荐系统;相似度

中图法分类号　TP311

收稿日期:2011-06-10;**修回日期**:2012-01-05

基金项目:国家"九七三"重点基础研究发展计划基金项目(2005CB321902);河北省自然科学基金项目(F2011203219);教育部高等学校博士学科点专项科研基金项目(20101333110013);河北省高等学校科学技术研究重点项目(ZH2012028)

通信作者:张付志(xjzfz@ysu.edu.cn)

推荐系统(recommender systems)作为一种信息过滤技术,逐渐成为解决互联网上信息过载(information overload)问题的有效工具[1].现有的推荐技术主要包括基于内容的推荐、协同过滤(collaborative filtering)推荐、基于知识的推荐以及混合推荐等,已广泛应用于大型的电子商务系统中,例如 Amazon,eBay 等.

协同过滤是推荐系统中应用最为成功的技术之一[2].其基本思想是基于用户-项目评分数据集,通过收集相似用户的兴趣信息进而为目标用户进行推荐.但是,传统的协同过滤推荐算法存在一个弊端:由于用户评分的极端稀疏性,仅仅依靠用户之间的相似度来选取目标用户的邻居,导致推荐精度较低,并且在面对用户概貌注入攻击(profile injection attacks)时,算法的抗攻击能力较差.本文针对这个问题,提出了将用户相似度和用户间信任度作为邻居选取的双重依据,更好地提高邻居选取的质量,进而提高推荐精度,增强算法的抗攻击能力.

1 相关工作

为了提高系统推荐精度,针对传统的协同过滤推荐算法,国内外学者提出了诸多改进的启发式推荐算法.例如,为了解决数据稀疏情况下推荐精度低的问题,黄创光等人[3]提出了一种不确定近邻的协同过滤推荐算法.但是,在不同应用环境中很难准确地计算出不确定近邻因子,难以达到用户群和项目群推荐结果的平衡.李聪等人[4]提出了一种基于领域最近邻的协同过滤推荐算法,通过对有推荐能力用户的未评分项进行评分预测,以提高最近邻寻找的准确性,但是当系统中存在攻击概貌(attack profiles)时,根据该方法获得的邻居质量较差,从而影响推荐质量.Netflix 大奖获得者提出了基于矩阵分解技术的推荐算法[5],该方法提高了推荐质量,但是并没有考虑攻击概貌对推荐精度的影响.

文献[6]中指出基于模型的推荐算法具有较强的抗攻击能力.Sandvig 等人[7]将数据挖掘技术与协同过滤推荐相结合,提出了一种基于关联规则挖掘的协同过滤推荐算法,该算法以降低推荐精度为代价获得了较强的鲁棒性,并且其推荐覆盖面较小.Mobasher 等人[6]提出了一种基于概率潜在语义分析(probabilistic latent semantic analysis, PLSA)的推荐算法,利用 PLSA 技术完成对用户(或项目)的聚类,从而实现对目标用户的推荐.相比其他聚类算法,该方法推荐精度较高,并且受攻击概貌影响较小.Mehta 等人[8]提出了一种基于奇异值分解(singular value decomposition, SVD)的推荐方法,通过采用 M-estimators 函数降低攻击用户对推荐结果的影响,从而提高系统的抗攻击能力,但是该方法只适用于小规模攻击的情况,当面对大规模攻击时,该方法取得的效果并不理想.

另外,还有些学者将用户间的信任关系引入到推荐过程中,以解决传统的协同过滤推荐算法存在的问题,并且提出了诸多信任计算模型.例如,Pitsilis 等人[9-10]从人的主观逻辑思维角度分析用户之间的信任关系,并且基于不确定概率理论提出了一种信任计算模型.由于不确定度是根据用户之间的共同评分计算的,所以在用户评分数据极端稀疏的情况下,得到的用户信任度不准确.Donovan 等人[11]从用户概貌级和项目级两个方面对用户之间的信任关系进行度量,并提出了一种改善系统推荐精度的信任计算模型.但是,当系统中存在攻击概貌时,推荐质量下降.针对传统的协同过滤推荐算法在邻居选取方面存在的局限性,Kwon 等人根据信息来源可靠性理论,提出了一种多维信任计算模型[12],从专业技能(expertise)、可信赖性(trustworthiness)和相似度(similarity)3 个方面对信任进行了分析和度量,并且采取三者加权求和的方式实现可信邻居的选取.但是,该模型仅考虑了用户评分的异构性,当系统中存在攻击概貌时,仍然存在脆弱性.近几年,信任感知推荐算法逐渐成为该领域研究的一个热点.Jamali 等人[13]提出了随机游走模型 TrustWalker,通过执行多次随机游走,融合返回的多个评分,得到目标用户对目标项目的预测评分,但是该模型受评分数据稀疏性影响较大.Ma 等人[14]提出了利用矩阵分解技术进行推荐的方法 RSTE,通过对用户和项目潜在特征的学习,得到对目标用户的推荐.并且,他们还提出了融合社会环境信息的推荐方法[15],尝试着将 RSTE 方法应用到基于隐式信任关系的推荐中[16].但是,该推荐方法受用户之间的直接信任信息的稀疏性较大.

针对以上方法中存在的问题,在已有研究的基础上,提出了一种基于双重邻居选取策略的协同过滤推荐算法(collaborative filtering recommendation algorithm based on double neighbor choosing strategy, CF-DNC),通过用户相似度和信任度的双重过滤,得到目标用户的最佳邻居集,进而完成对目标用户的推荐.本文的创新点主要体现在以下几个方面:

1) 提出了一种兴趣相似用户集选取算法,基于用户相似度的计算,动态选取目标用户的兴趣相似用户;

2) 提出了一种用户信任计算模型,根据目标用户与兴趣相似用户的信任关系,筛选出目标用户的可信邻居用户集;

3) 基于双重邻居选取策略,提出了一种新的协同过滤推荐算法,综合利用可信邻居用户的评分信息,实现对目标用户的推荐;

4) 在 MovieLens 数据集和 Netflix 数据集上分别进行了算法对比实验,相比现有的算法,所提出的算法不仅提高了推荐精度,而且具有较强的抗攻击能力.

2 基于双重邻居选取策略的推荐算法(CF-DNC)

2.1 相关定义

在协同过滤推荐系统中,用户评分数据库中包括 m 个用户组成的集合 $U=\{u_1,u_2,\cdots,u_m\}$ 和 n 个项目组成的集合 $I=\{i_1,i_2,\cdots,i_n\}$,用户-项目评分数据集可以用 $m\times n$ 阶矩阵 $\boldsymbol{R}$ 表示. $R_{i,j}(1\leqslant i\leqslant m,1\leqslant j\leqslant n)$ 表示用户 u_i 对项目 i_j 的评分,如果用户 u_i 对项目 i_j 没有进行评分,则记作 $R_{i,j}=\varnothing$. 用户 u_i 的评分集合表示为 $R(u_i)=\{R_{i,1},R_{i,2},\cdots,R_{i,n}\}$,用户对项目 i_j 的评分集合表示为 $R(i_j)=\{R_{1,j},R_{2,j},\cdots,R_{m,j}\}$.

定义 1. δ 算子:

$$\delta(H,\alpha)=\begin{cases}1, & \alpha\in H,\alpha\neq\varnothing;\\ 0, & \alpha\in H,\alpha=\varnothing.\end{cases}$$

其中,H 表示某一用户的评分集合或对某一项目的评分集合;α 表示集合 H 中的某个元素.

定义 2. η 算子:

$$\eta(H)=\{\alpha\mid\delta(H,\alpha)=1\},$$

显然,$\eta(H)\subseteq H$.

定义 3. Θ 算子:

$$\Theta(H_1,H_2)=\{\alpha\mid\delta(H_1,\alpha)=1,\delta(H_2,\alpha)=1\},$$

其中,H_1,H_2 表示的含义与 H 相同,显然,$\Theta(H_1,H_2)\subseteq\eta(H_1),\Theta(H_1,H_2)\subseteq\eta(H_2)$.

定义 4. 邻居候选集 C,给定一个用户-项目评分矩阵 $\boldsymbol{R}$,目标用户 $u_a\in U$,目标项目 $i_j\in I$,如果 $\exists u_k\in U$,使得 $R_{k,j}\neq\varnothing$,那么就说用户 u_k 是目标用户 u_a 在项目 i_j 上的一个邻居候选用户,则目标用户 u_a 的邻居候选集 $C(u_a)$ 表示为

$$C(u_a)=\{u_k\mid R_{k,j}\in\eta(I_j),u_k\in U\}.$$

定义 5. 兴趣相似用户集 S,给定一个相似度阈值 T_{sim},目标用户 $u_a\in U$,对于 $\forall u_x\in C(u_a)$,如果 u_x 与目标用户 u_a 的相似度 $sim_{a,x}>T_{sim}$,那么就说 u_x 是目标用户 u_a 的一个兴趣相似用户,则目标用户 u_a 的兴趣相似用户集 $S(u_a)$ 表示为

$$S(u_a)=\{u_x\mid sim_{a,x}>T_{\mathrm{sim}},a\neq x,u_x\in C(u_a)\}.$$

定义 6. 可信邻居用户集 T,对于目标用户 $u_a\in U$,u_a 对 $S(u_a)$ 中的所有用户的信任度从大到小排列为 $t_{a,1},t_{a,2},\cdots,t_{a,s}$,$s=|S(u_a)|$,则对应的前 l 个用户 $u_1,u_2,\cdots,u_l$ 就是目标用户 u_a 的可信邻居用户,记为 $T(u_a)=\{u_1,u_2,\cdots,u_l\}$.

2.2 兴趣相似用户集的动态选取

在推荐过程中,选取与目标用户相似度较大的用户作为邻居是保障推荐质量的关键. 但是在传统的协同过滤推荐算法 KNN[17] 中,如果选择的 k 近邻中包括一些与目标用户相似度非常小的用户,则会导致系统推荐精度降低. 因此,为了减小兴趣偏好差异较大的用户对目标用户推荐的影响,现有的工作中通常设定一个阈值对用户相似度进行界定,选取相似度大于该值的用户作为目标用户的邻居. 但是,在不同的环境中,推荐系统无法根据数据的变化而自适应地进行调整,扩展性较差. 为此,我们提出目标用户的兴趣相似用户集动态选取方法. 该方法根据目标用户 u_a 与所有邻居候选用户的相似度均值来设定相似度阈值,其计算公式如下:

$$T_{\mathrm{sim}}(u_a)=\frac{\sum_{i=1}^{k}sim_{a,i}}{k},k=|C(u_a)|,\tag{1}$$

$$sim_{a,i}=\frac{\sum_{i_k\in I_{a,i}}(R_{a,k}-\bar{R}_a)(R_{i,k}-\bar{R}_i)}{\sqrt{\sum_{i_k\in I_{a,i}}(R_{a,k}-\bar{R}_a)^2}\sqrt{\sum_{i_k\in I_{a,i}}(R_{i,k}-\bar{R}_i)^2}},\tag{2}$$

其中,$sim_{a,i}$ 表示目标用户 u_a 与邻居候选用户 $u_i(u_i\in C(u_a))$ 的相似度,采用 *Pearson* 相关系数公式计算得到;$R_{a,k}$ 和 $R_{i,k}$ 分别表示 u_a 和 u_i 对项目 i_k 的评分;$\bar{R}_a$ 和 $\bar{R}_i$ 分别表示 u_a 和 u_i 的平均评分;$I_{a,i}$ 表示 u_a 和 u_i 的共同评分项目集. 如果用户 u_i 与目标用户 u_a 的相似度满足 $sim_{a,i}\geqslant T_{sim}(u_a)$,则 u_i 是目标用户 u_a 的兴趣相似用户.

目标用户的兴趣相似用户集(*preference similar users*, PSU)选取算法如下:

算法 1. 兴趣相似用户集选取算法 $PSU(u_a)$.

输入:目标用户 u_a,目标项目 i_j,用户-项目评分矩阵 $\boldsymbol{R}$;

输出:目标用户 u_a 的兴趣相似用户集 $S(u_a)$.

Begin

① $S(u_a)\leftarrow\varnothing$; sum$\leftarrow$0;

② $C(u_a)\leftarrow\{u_k \mid R_{k,j}\in\eta(I_j), u_k\in U\}$;

③ *for each* $u_i\in C(u_a)$ *do*

④ $sim_{a,i}\leftarrow$similarity(u_i,u_a);

⑤ sum$\leftarrow$sum$+sim_{a,i}$;

⑥ *end for*

⑦ $T_{sim}\leftarrow\dfrac{\text{sum}}{|C(u_a)|}$;

⑧ *for each* $u_i\in C(u_a)$ *do*

⑨ *if* $sim_{a,i}>T_{sim}$ *then*

⑩ $S(u_a)\leftarrow S(u_a)\cup u_i$;

⑪ *end if*

⑫ *end for*

⑬ *return* $S(u_a)$;

End

算法 1 主要包括 3 部分:第 1 部分为行①~②,主要进行变量初始化,并找出目标用户的邻居候选集 $C(u_a)$;第 2 部分为行③~⑥,主要完成集合 $C(u_a)$中的每一个用户与目标用户的相似度计算;第 3 部分为行⑦~⑬,主要计算用户相似度阈值,并且查找出所有满足条件的用户,作为目标用户的兴趣相似用户集.

下面通过一个例子来说明兴趣相似用户集的动态选取过程. 表 1 给出了 *Alice* 和其他 6 个用户对项目 $i_1,i_2,\cdots,i_6$ 的评分信息以及利用 *Pearson* 相关系数计算出的这 6 个用户与 *Alice* 的相似度. 假设 *Alice* 为目标用户,项目 i_5 为目标项目,表中"?"表示我们要计算出用户 *Alice* 对项目 i_5 的预测评分 P_{Alice,i_5}.

Table 1 Ratings of Users and the Similarity between Alice and the Other 6 Users

表 1 用户的评分信息以及 Alice 与其他 6 个用户的相似度

User	i_1	i_2	i_3	i_4	i_5	i_6	The Similarity with Alice
Alice	5	1	2	3	?	3	—
u_1	4	1	2	2	3	2	0.9540
u_2	4	3		3	3	2	0.5000
u_3	1	3	2	1	3	1	−0.8292
u_4	3	1	3		1	2	0.6625
u_5	4	3		1	2	3	0.3244
u_6	5		4		4	2	0.5000

从表 1 可以看出,*Alice* 的邻居候选集 $C(Alice)=\{u_1,u_2,u_3,u_4,u_5,u_6\}$,根据邻居候选集中的用户与 *Alice* 的相似度情况可得 $T_{sim}(Alice)=0.3520$,则 $S(Alice)=\{u_1,u_2,u_4,u_6\}$.

2.3 用户信任计算模型

在传统的协同过滤推荐过程中,用户相似度是邻居选取的主要依据. 但是,由于用户评分数据较少,用户相似度计算存在较大的偶然因素,不能准确地度量用户间的相似性[3]. 因此,在选取的兴趣相似用户集 $S(u_a)$基础上,本文将用户之间的信任度作为邻居选取的第 2 重依据. 我们假设在用户间过去的交互行为中,如果某用户提供的可靠推荐次数越多,则其信任度越大. 例如用户 u_a,u_b 和用户 u_c,在过去的推荐历史中,如果 u_a 和 u_b 对 u_c 推荐的总次数均为 10,但是通过 u_c 的反馈信息可知,u_c 对 u_a 推荐的结果仅有 1 次满意,而 u_c 对 u_b 推荐的结果满意的次数为 9,显然 u_c 对 u_b 的信任度要远远大于 u_a,并且在以后的交互行为中,u_c 更倾向于采纳 u_b 的推荐结果.

基于以上讨论,用户信任计算模型的构造方法如下:

本文在 *O'Donovan* 提出的项目级信任计算模型的基础上,提出了一种改进的用户信任计算模型. 利用推荐系统评价中的"*leave-one-out*"方法,将 $\forall u_k\in S(u_a)$作为推荐用户,针对项目集 $I_{a,k}=\{i_j \mid R_{a,j}\in\Theta(U_a,U_k), i_j\in I\}$中的每一个项目 i_j,根据式(3)对目标用户 u_a 进行评分预测.

$$P_{a,j}=\bar{R}_a+\frac{(R_{k,j}-\bar{R}_k)\times sim_{a,k}}{|sim_{a,k}|},\tag{3}$$

其中,$P_{a,j}$表示推荐用户 u_k 对目标用户 u_a 在目标项目 i_j 上的预测评分;$R_{k,j}$表示 u_k 对 i_j 的评分;$\bar{R}_a$ 和 $\bar{R}_k$ 分别表示 u_a 和 u_k 的平均评分;$sim_{a,k}$表示 u_a 和 u_k 的相似度.

根据预测评分与实际评分的偏差,目标用户 u_a 对推荐用户 u_k 的预测能力估算如下:

$$sat_{a,k}^{j}=\begin{cases}1, & |P_{a,j}-R_{a,j}|\leqslant\varepsilon;\\ 0, & \text{else.}\end{cases}\tag{4}$$

式(4)中,$sat_{a,k}^{j}$表示目标用户 u_a 对推荐用户 u_k 在项目 i_j 上的预测能力估计值;$P_{a,j}$表示的含义与式(3)中相同;$R_{a,j}$表示 u_a 对 i_j 的实际评分;本文中,选取常数 $\varepsilon=1.2$.

假设 $t_{a,k}$表示目标用户 u_a 对推荐用户 u_k 的信任度,计算公式如下:

$$t_{a,k}=\frac{\sum_{i=1}^{|I_{a,k}|}sat_{a,k}^{j}}{|I_{a,k}|}. \quad (5)$$

基于上述信任计算模型,给出用户信任度计算(*user trust computing*, *UTC*)的算法如下:

算法 2. 用户间的信任度计算算法 UTC(u_a, u_k).

输入:用户 u_a, u_k,用户-项目评分矩阵 $\boldsymbol{R}$;

输出:用户 u_a 对用户 u_k 的信任度 $t_{a,k}$.

Begin

① $I_{a,k}\leftarrow\{i_j \mid R_{a,j}\in\Theta(U_a,U_k), i_j\in I\}$;

② *if* $I_{a,k}=\varnothing$ *then*

③ $t_{a,k}\leftarrow 0$;

④ *else*

⑤ *for each* $i_j\in I_{a,k}$ *do*

⑥ $p_{a,j}\leftarrow$Predict_CF(u_k, u_a);

⑦ $sat_{a,k}^{j}\leftarrow$satisfy(u_a, u_k);

⑧ *end for*

⑨ $t_{a,k}\leftarrow$trust(u_a, u_k);

⑩ *end if*

⑪ *return* $t_{a,k}$.

End

算法 2 主要包括两部分:第 1 部分为行①,找出用户 u_k 与用户 u_a 的共同评分项目集 $I_{a,k}$;第 2 部分为行②~⑪,通过统计用户 u_a 对用户 u_k 在项目集 $I_{a,k}$ 中每个项目上的预测能力估计值,进而计算得到用户 u_a 对用户 u_k 的信任度.

对于前面讨论的例子,根据上述信任计算模型可以计算出目标用户 *Alice* 对 S(*Alice*)中每一个用户的信任度:$t_{Alice,u_1}=1.0$, $t_{Alice,u_2}=0.25$, $t_{Alice,u_4}=0.5$, $t_{Alice,u_6}=0$.

2.4 推荐算法 CF-DNC

传统的协同过滤推荐算法仅仅依靠用户相似度的大小选取目标用户的邻居用户集,具有片面性[9-12],尤其是面对用户概貌注入攻击时,系统推荐精度明显降低. 因此,本文将用户相似度和信任度作为邻居的双重选取依据,并提出基于双重邻居选取策略的协同过滤推荐算法CF-DNC,其核心思想如下:

1) 针对目标项目 i_j,选取出目标用户 u_a 的邻居候选集 C(u_a);

2) 利用算法 PSU(u_a)动态生成目标用户的兴趣相似用户集 S(u_a);

3) 根据建立的用户信任计算模型,计算目标用户对兴趣相似用户集 S(u_a)中每个用户的信任度,选取信任度最大的 *Top*-l 个用户作为目标用户的可信邻居用户集 T(u_a);

4) 根据可信邻居用户的评分信息,利用式(6)计算出目标用户 u_a 在目标项目 i_j 上的预测评分.

$$P_{a,j}=\bar{R}_a+\frac{\sum_{u_k\in T(u_a)}(R_{k,j}-\bar{R}_k)\times t_{a,k}}{\sum_{u_k\in T(u_a)}|t_{a,k}|}, \quad (6)$$

其中,T(u_a)表示 u_a 的邻居集;$t_{a,k}$ 表示目标用户 u_a 对邻居用户 u_k 的信任度;$R_{k,j}$ 表示 u_k 对目标项目 i_j 的评分;$\bar{R}_a$ 和 $\bar{R}_k$ 分别表示 u_a 和 u_k 的平均评分.

根据上述算法思想,给出基于双重邻居选取策略的协同过滤推荐算法如下:

算法 3. 协同过滤推荐算法 CF-DNC.

输入:用户-项目评分矩阵 $\boldsymbol{R}$;

输出:用户 u_a 对项目 i_j 的预测评分 $P_{a,j}$.

Begin

① T(u_a)$\leftarrow\varnothing$;

② S(u_a)$\leftarrow$PSU(u_a);

③ *for each* $u_k\in$S(u_a) *do*

④ $t_{a,k}\leftarrow$UTC(u_a, u_k);

⑤ *end for*

⑥ *sort* 目标用户 u_a 对 S(u_a)中每个用户的信任度;

⑦ T(u_a)$\leftarrow$信任度最大的前 l 个用户;

⑧ $P_{a,j}\leftarrow$Predict_CF-DNC (u_a, i_j);

⑨ *return* $P_{a,j}$.

End

算法 3 主要包括 3 部分:第 1 部分为行①~②,主要完成变量初始化,并获得目标用户的兴趣相似用户集;第 2 部分为行③~⑤,主要完成目标用户对兴趣相似用户的信任度计算功能;第 3 部分为行⑥~⑨,根据选取的可信邻居用户的评分信息,计算得到目标用户 u_a 在目标项目 i_j 上的预测评分 $P_{a,j}$.

对于前面讨论的例子,根据上述推荐算法,随着选取的邻居的不同,计算得到的预测评分 P_{Alice,i_5} 结果对比如表 2 所示:

Table 2 Comparison of Predicting Results

表 2 预测评分结果对比

Neighbors	u_1	u_1 u_4	u_1 u_2 u_4
P_{Alice,i_5}	3.4667	2.9111	2.8952

3 实验结果及分析

3.1 数据集

本文实验采用了以下 2 个数据集:

1) MovieLens[18] 站点(http://movielens.umn.edu/)提供的数据集. 该数据集中,用户对自己看过的电影进行评分,评分范围为 1～5,"1"表示"不喜欢","5"表示"非常喜欢",包括 943 个用户对 1 682 部电影的大约 100 000 次评分,该数据集的稀疏度为 93.7%.

2) Netflix 站点(http://www.netflix.com/)提供的数据集. 该数据集中,包括 480 189 个用户对 17 770 部电影的 103 297 638 次评分,评分范围和 MovieLens 数据集相同. 在实验中,我们选取了 2 000 个用户对 4 000 部电影的大约 413 292 次评分信息,稀疏度为 94.83%.

实验中,我们将每个数据集分为两部分:随机选取 10%作为测试集、剩下的 90%作为训练集.

3.2 评价指标

为了评价算法的推荐精度,本文采用平均绝对偏差(mean absolute error, *MAE*)指标来度量,其值可以通过计算项目的预测评分与用户对项目的实际评分之间的偏差得到. 显然,*MAE* 值越小,算法的推荐精度越高. *MAE* 的计算公式如下[2]:

$$MAE = \frac{\sum_{j=1}^{n} | p_j - r_j |}{n}, \tag{7}$$

其中,p_j 为系统对目标用户在项目 i_j 上的预测评分;r_j 为目标用户对项目 i_j 的实际评分;n 表示预测的次数.

另外,为了评价推荐算法的抗攻击能力,本文选取 MAE 和预测偏差(*prediction shift*)两个评价指标来度量. 预测偏差表示项目受攻击前后系统预测的偏差,预测偏差越小,算法的抗攻击能力越强,其计算公式如下[19]:

$$PredShift(u,i) = \frac{1}{n}\sum p'(u_k,i_j) - p(u_k,i_j), \tag{8}$$

式(8)中,$p(u_k,i_j)$和 $p'(u_k,i_j)$分别表示项目 i_j 受攻击前后,推荐算法对用户 u_k 在该项目上的预测评分;n 表示预测的次数.

3.3 推荐精度对比

为了评价算法的推荐精度,将本文提出的基于双重邻居选取策略的协同过滤推荐算法(CF-DNC)与传统的协同过滤推荐算法(collaborative filtering recommendation algorithm, CF)和 John O'Donovan 信任推荐算法进行了实验对比,另外,我们还对仅采用兴趣相似用户集(collaborative filtering recommendation algorithm based on preference similar users, CF-PSU)的推荐效果和仅采用用户信任计算模型(collaborative filtering recommendation algorithm based on user trust computing, CF-UTC)的推荐效果进行了对比. 采用 MovieLens 和 Netflix 数据集,分别为目标用户选取不同的邻居用户个数,通过计算项目的预测评分和实际评分之间的偏差,得到的推荐精度对比结果如图 1 所示:

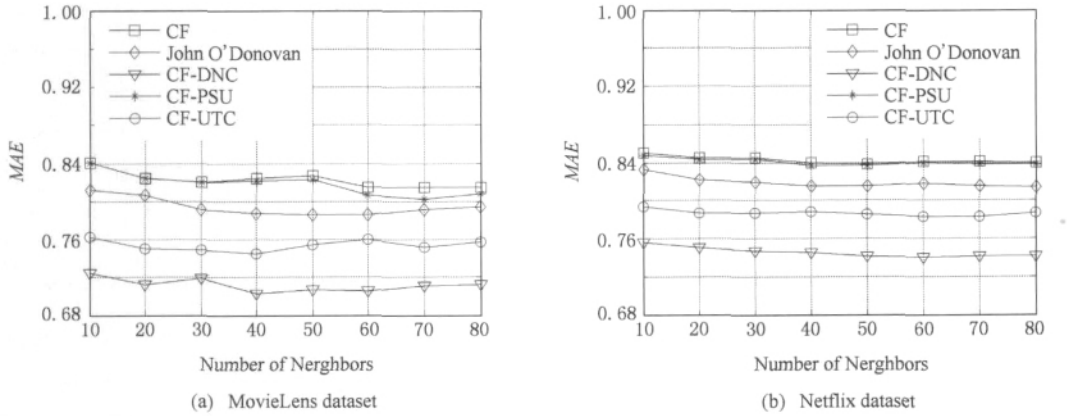

Fig. 1 Comparison of recommendation precision with different datasets.

图 1 不同数据集上推荐精度的对比

从图 1 可以看出,无论采用哪种数据集,CF-UTC 的推荐 *MAE* 值明显小于 CF-PSU 和 John O'Donovan 信任推荐算法. 因此,单独采用信任度计算模块取得的效果优于单独采用相似度计算模块,并且本文提出的信任计算模型取得效果要优于 O'Donovan 提出的信任计算模型. 另外,CF-DNC 和

John O'Donovan 信任推荐算法的推荐 *MAE* 值均小于 CF. 由此可见,将用户间信任关系引入到推荐过程中,可以改善系统推荐质量. 以在 MovieLens 数据集上的对比结果为例,与传统的协同过滤推荐算法相比,CF-DNC 的推荐精度大约提高了 13.47%,与 John O'Donovan 信任推荐算法相比,CF-DNC 的推荐精度大约提高了 10.39%. 由于 O'Donovan 提出的信任计算模型得到的某个用户的信任度是其他用户对该用户推荐能力的综合评价,相当于用户的一个全局描述,而 CF-DNC 算法中采用的信任计算模型得到是用户之间的信任关系,相当于用户的一个局部评价,符合信任的主观性特征. 因此,本文提出的基于双重邻居选取策略的协同过滤推荐算法 CF-DNC 取得的效果最优.

3.4 抗攻击能力对比

为了评价 CF, CF-DNC 以及 John O'Donovan 信任推荐算法的抗攻击能力,我们向原有的数据集中注入混合攻击数据(随机攻击和均值攻击,各攻击类型的用户概貌个数相等). 在推荐过程中,分别采用 MovieLens 和 Netflix 数据集,选取邻居用户的个数为 40,填充规模为 1%,3%,5%,10%,25%,攻击规模选取 1%,2%,5%,10%,随着攻击规模和填充规模的不断增大,3 种推荐算法的推荐精度(*MAE* 值)对比结果分别如表 3 和表 4 所示.

Table 3 Comparison of Recommendation Precision(*MAE*) for Hybrid Attack when Using MovieLens Dataset

表 3 采用 MovieLens 数据集时混合攻击下推荐精度(*MAE*)的对比

Filler Size	Attack Size											
	1%			2%			5%			10%		
	CF	John O'Donovan	CF-DNC	CF	John O'Donovan	CF-DNC	CF	John O'Donovan	CF-DNC	CF	John O'Donovan	CF-DNC
1%	0.8949	0.8497	0.7623	0.9097	0.8572	0.7736	0.8943	0.8724	0.7760	0.9156	0.8845	0.7808
3%	0.8982	0.8442	0.7789	0.9089	0.8490	0.7742	0.9106	0.8545	0.7779	0.9248	0.8627	0.7830
5%	0.9021	0.8309	0.7840	0.9063	0.8521	0.7846	0.8974	0.8586	0.7854	0.9268	0.8635	0.7868
10%	0.8945	0.8288	0.7824	0.9009	0.8310	0.7833	0.9012	0.8637	0.7842	0.9034	0.8650	0.7885
25%	0.8924	0.8634	0.7835	0.8929	0.8684	0.8057	0.8954	0.8789	0.8124	0.9153	0.8838	0.8228

Table 4 Comparison of Recommendation Precision(*MAE*) for Hybrid Attack when Using Netflix Dataset

表 4 采用 Netflix 数据集时混合攻击下推荐精度(*MAE*)的对比

Filler Size	Attack Size											
	1%			2%			5%			10%		
	CF	John O'Donovan	CF-DNC	CF	John O'Donovan	CF-DNC	CF	John O'Donovan	CF-DNC	CF	John O'Donovan	CF-DNC
1%	0.9087	0.8512	0.7831	0.9168	0.8639	0.7908	0.9152	0.8762	0.7935	0.9261	0.8961	0.8035
3%	0.9072	0.8497	0.7919	0.9145	0.8582	0.7923	0.9214	0.8673	0.7944	0.9334	0.8716	0.8024
5%	0.9114	0.8483	0.7934	0.9139	0.8609	0.8004	0.9063	0.8650	0.8057	0.9362	0.8752	0.8069
10%	0.9137	0.8516	0.7905	0.9167	0.8612	0.8026	0.9177	0.8709	0.8061	0.9109	0.8776	0.8127
25%	0.9126	0.8607	0.8017	0.9194	0.8725	0.8131	0.9212	0.8881	0.8210	0.9297	0.8934	0.8314

从表 3 和表 4 可以看出,无论采用 MovieLens 数据集还是 Netflix 数据集,在同一填充规模下,随着攻击规模的增大,3 种推荐算法的推荐 *MAE* 值基本上呈上升趋势,可见,系统中攻击用户越多,系统受影响越大,从而推荐质量越差. 并且,CF-DNC 算法的推荐 *MAE* 值要低于 John O'Donovan 信任推荐算法和传统的协同过滤推荐算法. 以采用 MovieLens 数据集时的推荐精度为例,与传统的协同过滤推荐算法相比,CF-DNC 算法的推荐精度提高了 13.12%;与 John O'Donovan 信任推荐算法相比,CF-DNC 算法的推荐精度提高了 8.44%. 由此可以证明,本文提出的基于双重邻居选取策略的协同过滤推荐算法 CF-DNC 具有更强的抗攻击能力.

在混合攻击下,分别采用 MovieLens 和 Netflix 数据集,当填充规模分别为 3%,5%和 10%时,采用 3 种推荐算法的预测偏差对比结果分别如图 2 至图

4 所示.

从图 2～图 4 可以看出:在同一填充规模下,不管在哪种数据集下,3 种推荐算法的预测偏差随着攻击规模的增大而增大,因此攻击用户越多,推荐质量越差. 另外,在同一填充规模和攻击规模下,与 John O'Donovan 信任推荐算法和传统的协同过滤推荐算法 CF 相比,CF-DNC 算法的预测偏差明显较小. 因此,CF-DNC 算法具有较强的抗攻击能力.

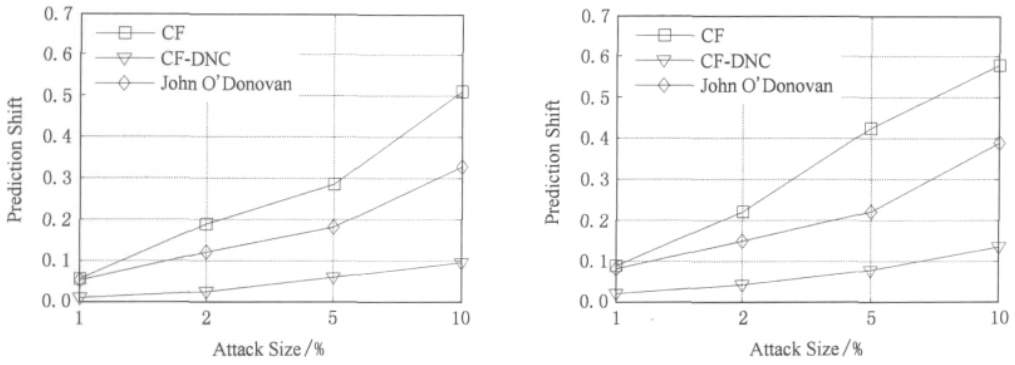

Fig. 2 Comparison of prediction shift with 3% filler size.

图 2 3%填充规模时预测偏差的对比

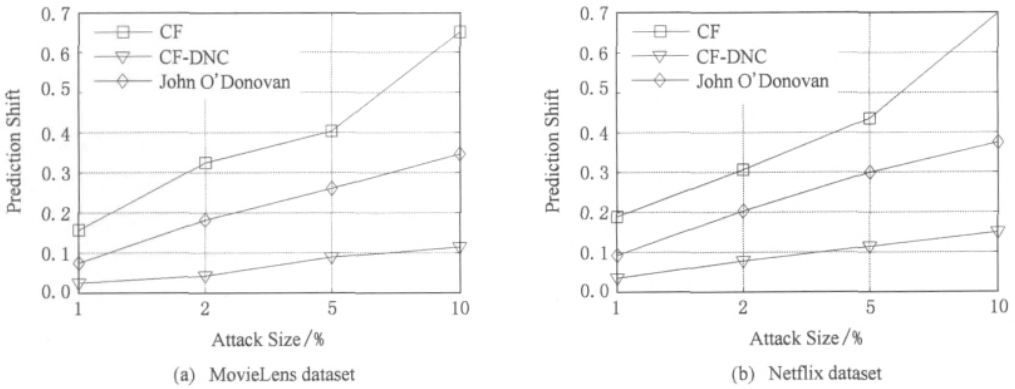

Fig. 3 Comparison of prediction shift with 5% filler size.

图 3 5%填充规模时预测偏差的对比

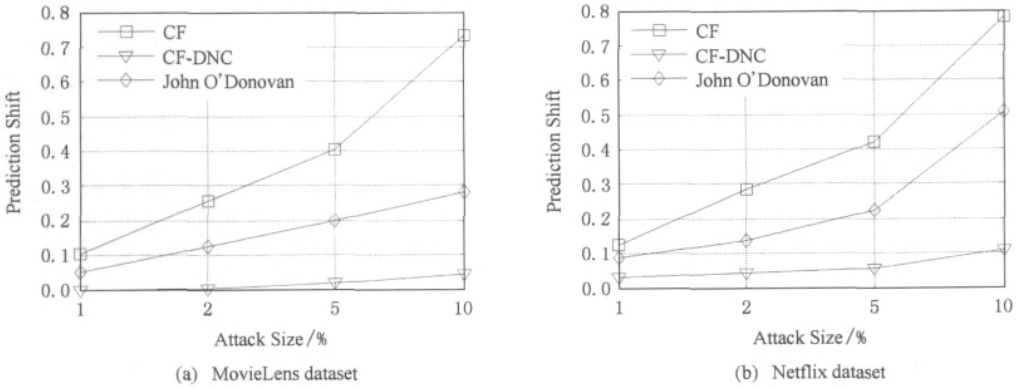

Fig. 4 Comparison of prediction shift with 10% filler size.

图 4 10%填充规模时预测偏差对比

4 结论及进一步工作

随着协同过滤推荐算法在电子商务中的广泛应用,提高其推荐精度和抗攻击能力已成为非常重要的研究问题,本文在这方面进行了一些有益的探索. 文中提出了一种基于双重邻居选取策略的协同过滤推荐算法,给出了兴趣相似用户集动态选取算法和

用户信任计算模型.

该算法将用户相似度和信任度作为目标用户邻居的双重选取依据,有效地提高了算法的推荐精度,并且具有较好的抗攻击能力.如何设计有效的推荐算法,为冷启动用户提供更准确的推荐将是下一步的研究工作.

参 考 文 献

[1] Xu Hailing, Wu Xiao, Li Xiaodong, et al. Comparison study of Internet recommendation system [J]. Journal of Software, 2009, 20(2): 350-362 (in Chinese)
(许海玲, 吴潇, 李晓东, 等. 互联网推荐系统比较研究[J]. 软件学报, 2009, 20(2): 350-362)

[2] Ricci F, Rokach L, Shapira B, et al. Recommender Systems Handbook [M]. Berlin: Springer, 2011: 145-186

[3] Huang Chuangguang, Yin Jian, Wang Jing, et al. Uncertain neighbors' collaborative filtering recommendation algorithm [J]. Chinese Journal of Computers, 2010, 33(8): 1369-1377 (in Chinese)
(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)

[4] Li Cong, Liang Changyong, Ma Li, et al. A collaborative filtering recommendation algorithm based on domain nearest neighbor [J]. Journal of Computer Research and Development, 2008, 45(9): 1532-1538 (in Chinese)
(李聪, 梁昌勇, 马丽, 等. 基于领域最近邻的协同过滤推荐算法[J]. 计算机研究与发展, 2008, 45(9): 1532-1538)

[5] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model [C] //Proc of the 14th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2008: 426-434

[6] Mobasher B, Burke R, Sandvig J. Model-based collaborative filtering as a defense against profile injection attacks [C] //Proc of the 21st National Conf on Artificial Intelligence. Menlo Park, CA: AAAI, 2006: 1388-1393

[7] Sandvig J, Mobasher B, Burke R. Robustness of collaborative recommendation based on association rule mining [C] //Proc of the 2007 ACM Conf on Recommender Systems. New York: ACM, 2007: 105-112

[8] Mehta B, Hofmann T, Nejdl W. Robust collaborative filtering [C] //Proc of the 2007 ACM Conf on Recommender Systems. New York: ACM, 2007: 49-56

[9] Pitsilis G, Marshall L. A model of trust derivation from evidence for use in recommendation systems, CS-TR-874 [R]. Newcastle, UK: University of Newcastle Upon Tyne, 2004

[10] Pitsilis G, Marshall L. Modeling trust for recommender systems using similarity metrics [C] //Proc of IFIPTM 2008. Berlin: Springer, 2008: 103-118

[11] O'Donovan J, Smyth B. Trust in recommender systems [C] //Proc of the 10th Int Conf on Intelligent User Interfaces. New York: ACM, 2005: 167-174

[12] Kwon K, Cho J, Park Y. Multidimensional credibility model for neighbor selection in collaborative recommendation [J]. Expert Systems with Applications, 2009, 36(3): 7114-7122

[13] Jamali M, Ester M. TrustWalker: A random walk model for combining trust-based and item-based recommendation [C] // Proc of the 15th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2009: 397-406

[14] Ma H, King I, Michael R L. Learning to recommend with social trust ensemble [C] //Proc of the 32nd Annual ACM SIGIR Conf on Research and Development in Information Retrieval. New York: ACM, 2009: 203-210

[15] Ma H, Zhou T C, Lyu M R, et al. Improving recommender systems by incorporating social contextual information [J]. ACM Trans on Infromation Systems, 2011, 29(2): Article 9

[16] Ma H, King I, Lyu MR. Learning to recommend with explicit and implicit social relations [J]. ACM Trans on Intelligent System and Technology, 2011, 2(3): Article 29

[17] Herlocker J, Konstan J, Borchers A, et al. An algorithmic framework for performing collaborative filtering [C] //Proc of the 22nd Annual Int ACM SIGIR Conf on Research and Development in Information Retrieval. New York: ACM, 1999: 230-237

[18] Miller BN, Albert I, Lam SK, et al. MovieLens unplugged: Experiences with an occasionally connected recommender system [C] //Proc of the Int Conf on Intelligent User Interfaces. New York: ACM, 2003: 263-266

[19] Mehta B, Nejdl W. Attack resistant collaborative filtering [C] //Proc of the 31st Annual Int ACM SIGIR Conf on Research and Development in Information Retrieval. New York: ACM, 2008: 75-82

Jia Dongyan, born in 1983. PhD candidate. Her main research interests include collaborative filtering, trusted computing, and information security(jdy_1983@163.com).

Zhang Fuzhi, born in 1964. Professor and PhD supervisor. His main research interests include intelligent network information processing, network and information security, service-oriented computing, etc.