计算机科学
COMPUTER SCIENCE

一种基于时效近邻可信选取策略的协同过滤推荐方法

韩志耕, 范远哲, 陈耿, 周婷

引用本文

韩志耕, 范远哲, 陈耿, 周婷. 一种基于时效近邻可信选取策略的协同过滤推荐方法[J]. 计算机科学, 2023, 50(11A): 220800199-11.

HAN Zhigeng, FAN Yuanzhe, CHEN Geng, ZHOU Ting. Time-effective Nearest Neighbor Trusted Selection Strategy Based Collaborative Filtering Recommendation Method [J]. Computer Science, 2023, 50(11A): 220800199-11.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220800199

一种基于时效近邻可信选取策略的协同过滤推荐方法

韩志耕[1,2]　范远哲[1,2]　陈　耿[3]　周　婷[1,2]
1 南京审计大学计算机学院/智能审计学院　南京 211815
2 江苏省审计信息工程重点实验室　南京 211815
3 南京审计大学会计学院　南京 211815
(hanzgnit@126.com)

摘　要　传统协同过滤推荐通常基于数据是静态的假设,在数据稀疏时存在推荐精度低下的问题。为解决该问题,一些研究尝试向推荐策略中添加有关用户兴趣变化、推荐能力可信度等补充信息,但对误导或干扰推荐的恶意用户兴趣策略变化和推荐能力波动等异常情况欠缺考虑,系统抗攻击性、推荐稳定性与可信性均难以得到保证。通过引入兴趣时效相似度和推荐信度重估两个概念,提出了一种基于时效近邻可信选取策略的协同过滤推荐方法。该方法充分考虑了影响目标用户近邻筛选质量的用户兴趣异常变化和推荐能力波动两个关键因素,构建了包含时效近邻筛选、可信近邻选取和评分预测3个策略的推荐流程。在MovieLens数据集和亚马逊video game数据集上,利用平均绝对误差,平均预测增量,攻击用户查准率、查全率和调和平均等评估指标,对所提策略与其他6种基准策略进行了比较。结果显示,新策略在推荐精度、抗攻击力和攻击者识别力上均有明显的性能提升。

关键词:协同过滤;时效近邻;可信近邻;推荐精度;抗攻击;攻击者识别
中图法分类号　TP391

Time-effective Nearest Neighbor Trusted Selection Strategy Based Collaborative Filtering Recommendation Method

HAN Zhigeng[1,2],FAN Yuanzhe[1,2],CHEN Geng[3] and ZHOU Ting[1,2]
1 School of Computer Science/School of Intelligence Audit,Nanjing Audit University,Nanjing 211815,China
2 Jiangsu Key Laboratory of Audit Information Engineering,Nanjing 211815,China
3 School of Accounting,Nanjing Audit University,Nanjing 211815,China

Abstract　The traditional collaborative filtering(CF) recommendation is usually based on the assumption that the data is static. When the data is sparse,it usually leads to low recommendation accuracy. With this in mind,some studies try to add supplementary information such as changes in user interest and the trustiness of recommendation ability to their strategies. However,most of them lack of consideration for the abnormal situations that mislead or interfere with the recommendation,such as malicious changes in user interests and fluctuations in the recommendation ability,and are difficult to ensure the anti-attack,recommendation stability and reliability of the recommendation system. By introducing interest time-effective similarity and re-evaluation on re-commendation trust degree,this paper proposes a time-effective nearest neighbor trusted selection strategy based collaborative filtering recommendation method. It takes into account two key factors,that is,the abnormal change of user interest and the fluctuation of user recommendation ability,which affect the quality of target user's neighbor filtering,and construct a recommendation process that includes three strategies,that is,time-effective nearest neighbor selection,trusted nearest neighbor selection and ra-ting prediction. Based on MovieLens dataset and Amazon video game dataset,and with the metrics such as mean absolute error (MAE),average prediction shift(APS),and attacker identification of precision ratio,recall ratio and F1 means,the performance of the proposed strategy and other six baselines are compared. The results show that our strategy outperform the baselines in recommendation accuracy,anti-attack and attacker identification.

Keywords　Collaborative filtering,Time-effective nearest neighbor,Trusted nearest neighbor,Recommendation accuracy,Anti-attack,Attacker recognition

---

基金项目:国家自然科学基金项目(72072091);江苏省高校自然科学基金项目(21KJA520002,22KJA520005);江苏省研究生科研与实践创新计划项目(KYCX23_2345);审计信息工程与技术协同创新中心项目
This work was supported by the National Natural Science Foundation of China(72072091),Natural Science Foundation of Colleges and Universities of Jiangsu Province(21KJA520002,22KJA520005),Postgraduate Research & Practice Innovation Program of Jiangsu Province(KYCX23_2345) and Audit Information Engineering and Technology Collaborative Innovation Center Project.
通信作者:范远哲(mg2009105@stu.nau.edu.cn)

1 引言

近年来，随着社会生产与人们生活大规模向信息空间迁移，信息过载所引发的信息迷航现象日趋严重。为帮助人们从急剧膨胀的数据中筛选出有用信息，推荐系统[1]应运而生。当前基于内容的推荐[2]、协同过滤推荐[3]、基于关联规则的推荐[4]、基于效用的推荐[5]、基于知识的推荐[6]以及混合推荐[7]等技术，已广泛应用于各类电子商务平台，用于提升用户粘度，降低服务跳出率、助力用户构建信息茧房。据统计，Netflix 上 80%被观看的电影、YouTube 上 60%的视频点击量和 Google 上 38%的新闻点击量均源自于推荐；Amazon 上 30%的商品页面浏览量、20%～30%的商品销售量，以及 35%的销售收入，均由个性化推荐所贡献。

作为电子商务推荐系统中应用最成功的推荐技术之一，协同过滤（Collaborative Filtering，CF）推荐的基本思想是相似的用户可能会喜欢同一种物品，相似的物品可能会被同一个用户所喜欢。CF 推荐具备无需领域知识、对推荐对象无特殊要求、易解释、便于工程实现等优势，但传统 CF 推荐由于基于数据是静态的假设，在数据稀疏时存在推荐精度低下的问题[8-9]。为提高 CF 推荐精度，一些研究尝试在推荐策略中添加有关用户兴趣变化的描述信息[10-13]，然而这些研究对误导或干扰推荐的兴趣异常变化欠缺考虑，恶意用户可以通过迎合目标用户的兴趣对推荐系统实施概貌注入攻击。另外，在推荐系统中，信度被用于衡量主体对他方推荐能力的信任程度。在推荐策略中增加对用户间信任关系的描述，也是提高推荐精度的常见做法[14-17]。然而如何对用户推荐能力进行可靠、一致性的动态评估尚未有提及，导致在信度波动情境下系统的推荐可信性无法得到保证。为弥补上述问题，通过对文献[18]所提策略进行改进，提出了一种基于时效近邻可信选取策略的协同过滤推荐方法。

主要贡献包括 4 个方面：

1）提出了基于时效近邻可信选取策略的 CF 推荐方法框架。该框架充分考虑用户兴趣异常变化和推荐信度波动两个时变因素，顺序利用时效近邻筛选、可信近邻选取和评分预测 3 个步骤，完成向目标用户的推荐，缓解了数据稀疏情况下传统 CF 推荐精度低、抗攻击能力弱等问题。

2）提出了一种基于兴趣时效相似度的时效近邻选取策略。通过赋予传统的兴趣相似度时效性，使得评分相同但评分时间不同的用户在兴趣相似度上有所差异，缓解了传统基于用户兴趣相似度的近邻选取策略因为目标用户兴趣随时间变化所导致的近邻选取不准确问题。

3）提出了基于推荐信度的可信近邻选取策略。综合利用近邻推荐信度的原始值、历史值、当前变化率和未来趋势 4 方面因素对推荐信度进行重估，使得恶意用户无法利用策略评分来改善他人对其推荐能力的信服程度，缓解了已有基于信度的近邻选取策略因为信度评估的可靠性存疑所带来的目标用户近邻选取不可信的问题。

4）基于 MovieLens 数据集和亚马逊 video game 数据集，采用平均绝对误差、平均预测增量、攻击用户查准率、查全率和调和平均这 5 个指标对所提策略进行了性能评估，验证了其在推荐精度（包括精度稳定性）和抗攻击力（包括攻击者识别力）上较基准策略均有显著提升。

2 相关工作

传统 CF 依赖于数据是静态的假设，通常忽略自然生成的数据中所蕴藏的时间相关现象，这些现象的出现会不断地改变着用户与项目之间的潜在关系，影响到最终推荐结果的精度[8]。以用户兴趣变化现象为例，为在推荐中反映用户兴趣变化，Liu[10]提出了一种基于聚类的 CF 推荐算法，该算法利用时间衰减函数对用户评分进行预处理，采用兴趣向量对用户进行描述，使用聚类算法对用户进行聚类，实现了反映用户兴趣变化的多维度画像。Dong 等[11]提出了一种基于用户兴趣变化和评论的 CF 算法，该算法利用主题模型从评论文本中挖掘商品主题特征，利用艾宾浩斯遗忘曲线协助计算用户评论相似度，并结合用户评论相似度和评分相似度获取用户之间的最终相似度。Xu 等[12]基于每个领域的专家都更具说服力以及用户兴趣会随时间推移而变化的事实，提出了一种基于用户可信度和时间上下文的 CF 推荐算法，通过将这两个因素纳入到基于修正的余弦相似度模型，提高了用户间相似度计算的准确度。Li 等[13]依据推荐需要契合用户短期兴趣的事实，提出了一种使用社会信息和动态时间窗口的 CF 算法，该算法依据艾宾浩斯遗忘曲线，借助细分时间窗口来界定用户的短期兴趣，并借助时间函数来区分各短期兴趣的重要度，实现了对用户兴趣变化的及时捕捉。上述方法通过在推荐过程中考虑目标用户的兴趣变化改善了推荐精度，但尚未对恶意用户的兴趣异常变化进行处置。以 CF 推荐中常见的用户概貌注入攻击[19]为例，恶意用户可在目标用户已评分项目上通过刻意迎合目标用户的兴趣来获取近邻身份，之后通过在目标用户未评分项目上实施策略评分，达到干扰或误导推荐的目的。与目标用户兴趣的正常变化不同，这种潜藏在恶意用户策略评分背后的兴趣异常变化如果得不到有效处置，势必会削弱推荐系统的抗攻击力。

与此同时，近年来通过在 CF 中引入信任关系来提高推荐精度和系统鲁棒性也备受关注。Chen 等[14]基于用户更愿意相信朋友推荐的事实，通过将基于记忆的 CF 思想与信任关系融入到概率矩阵分解模型，提出了一种社交网络环境下基于信任的推荐算法，解决了此类算法由于假设用户单一、同质所带来的信任关系挖掘不充分以及相似关系与信任关系无法高效融合的问题，提高了数据稀疏环境下 CF 推荐的精度；但该算法在利用由评分数据产生的相似度和信任关系产生的信任度融合构建的用户偏好模型寻找邻居时，尚未对信任关系的动态性进行关注。Wang 等[15]针对智慧城市中面向用户轨迹行为序列预测的兴趣点推荐，通常基于相似用户的偏好而较少考虑用户间信任关系的问题，将地理位置和时间信息纳入到基于兴趣点偏好的相似度计算模型，给出了具备信任增强的用户相似度计算方法，通过赋予相同时间区间内地理位置相近的用户更高的信任度，改善了兴趣点推荐的精度；不过该方法将评分时间简单地切分为 24 个时区的做法，无法在细粒度层面上可靠地反映用户信度随时间变化的情况。Wang 等[16]针对用户信任与协同过滤对用户关系的处理方式不同所导致的两者难以融合的问题，提出了一种新的基于信任的 CF 算法，该算法依据用户信任度计算相似度，实现了与传统协同过滤计算模式的统一；在此基础上构建了基于用户项目信任记录的协同过滤推荐模型，该模型的特色为在 CF

中融合考虑了用户、项目和信任3个对等维度，实现了推荐结果的多样性、准确性和鲁棒性；但信任度一般会随时间变化而变化，如何保证所使用的信任度在时间跨度上具有一致性和可靠性并没有提及。Cai等[17]提出了一种改进的融合相似度和信任度的CF算法，该算法融合信任度和相似度以计算用户之间的信任值，但对如何保证信度评估的一致性和可靠性也没有提及。

Jia等[18]针对传统CF推荐算法存在推荐精度低、应对攻击能力差的问题，提出了一种基于双重邻居选取策略的CF推荐算法。首先利用相似度计算选取目标用户的兴趣相似用户集，然后依据相似用户信度选出目标用户的可信邻居用户，最后依据可信邻居用户完成对目标用户的推荐。该算法虽在一定程度上提高了推荐精度和抗用户概貌注入攻击力，但对目标用户兴趣变化和邻居用户信度波动欠缺考虑，推荐精度和抗攻击力还有提升空间。

本文提出的基于时效近邻可信选取策略的CF推荐方法，通过在目标用户近邻筛选中兼顾用户兴趣异常变化和推荐能力波动这两个因素，采用时效近邻筛选、可信近邻选取和评分预测3个策略，实现了向目标用户的精准推荐，有效缓解了传统CF推荐方法在数据稀疏情况下存在的推荐精度低和抗攻击力差的问题。

3 协同过滤推荐新方法

3.1 背景技术及定义

CF推荐系统可表示为 $CFRS=\langle U,I,R,u_a,i_b,r_{ab}\rangle$，其中 $U=\{u_1,\cdots,u_i,\cdots,u_m\}$ 是 m 个用户 $u_i(1\leqslant i\leqslant m)$ 所组成的用户集；$I=\{i_1,\cdots,i_j,\cdots,i_n\}$ 是 n 个项目 $i_j(1\leqslant j\leqslant n)$ 所组成的项目集；$\boldsymbol{R}=\{r_{ij}\}_{m*n}$ 是 $m*n$ 个评分 r_{ij} 所组成的评分矩阵，此处 r_{ij} 表示 u_i 对 i_j 的评分，若 $r_{ij}=\text{null}$，则表示 u_i 对 i_j 未评分；u_a 是目标用户，i_b 是目标项目。CF推荐的主要任务是依据评分矩阵 $\boldsymbol{R}$，预测 u_a 在 i_b 上的评分 r_{ab}。

在介绍新方法之前，首先给出如下定义。

定义1（评分项目集，Rated Item Set，RIS） 若用户 u_a 对项目 i_j 进行了评分，即 $r_{aj}\neq\text{null}$，则 i_j 是 u_a 的一个评分项目。用户 u_a 的评分项目集 $RIS(u_a)$ 是该用户所有评分项目形成的集合，即 $RIS(u_a)=\{i_j\mid\forall i_j\in I\ \wedge\ r_{aj}\neq\text{null}\wedge\ 1\leqslant j\leqslant n\}$。

定义2（共同评分项目集，Common Rated Item Set，CRIS） 令 $RIS(u_a)$ 和 $RIS(u_i)$ 分别为用户 u_a 和 u_i 的评分项目集，则 u_a 和 u_i 的共同评分项目集 $CRIS(u_a,u_i)$ 可表示为 $CRIS(u_a,u_i)=RIS(u_a)\cap RIS(u_i)$。

定义3（邻居集，Neighbor Set，NS） 令 $CRIS(u_a,u_i)$ 为用户 u_a 与用户 u_i 的共同评分项目集，则 u_a 的邻居集可表示为 $NS(u_a)=\{u_i\mid\forall u_i\in U\ \wedge\ u_i\neq u_a,CRIS(u_a,u_i)\neq\varnothing\}$。其中，$\varnothing$ 表示空集合。

定义4（时效近邻集，Time-effective Nearest Neighbor Set，TeNNS） 令 $NS(u_a)$ 为用户 u_a 的邻居集，$TSD(u_a,u_i)$ 为 u_a 与用户 u_i 的时效相似度（Time-effective Similarity Degree，TSD），$G_{\text{TSD}}(u_a)$ 为 u_a 设定的时效相似度阈值，则 u_a 时效近邻集可表示为 $TeNNS(u_a)=\{u_i\mid\forall u_i\in NS(u_a)\wedge TSD(u_a,u_i)\geqslant G_{\text{TSD}}(u_a)\}$。

定义5（时效近邻信度集，Trust Degree Set of Time-effective Nearest Neighbor，TSTeNN） 令 $TeNNS(u_a)$ 为用户 u_a 的时效近邻集，$RT(u_a,u_i)$ 为用户 u_i 呈现给 u_a 的推荐信度（Recommendation Trust degree，RT），则 u_a 的时效近邻信度集可表示为 $TSTeNN(u_a)=\{RT(u_a,u_i)\mid\forall u_i\in TeNNS(u_a)\}$。

定义6（可信近邻集，Trusted Nearest Neighbor Set，TNNS） 令 $TeNNS(u_a)$ 和 $TSTeNN(u_a)$ 分别为用户 u_a 的时效近邻集和信度集，$RT(u_a,u_i)$ 为用户 u_i 呈现给 u_a 的推荐信度，则 u_a 的可信近邻集可表示为 $TNNS(u_a)=\{u_i\mid\forall u_i\in TeNNS(u_a)\ \wedge\ RT(u_a,u_i)\in topK(TSTeNN(u_a))\}$，其中 $topK(\{\cdot\})$ 表示由 $\{\cdot\}$ 中前 K 个较大元素构成的集合。

3.2 方法框架

本文提出的基于时效近邻可信选取策略的协同过滤推荐方法其框架如图1所示，该方法包括时效近邻筛选、可信近邻选取和评分预测3个策略单元。各单元的主要功能描述如下：

1）时效近邻筛选单元使用时效权重函数计算目标用户邻居在共同评分项目上的评分时效权重，之后采用纳入了时效权重的基于皮尔逊相关系数的兴趣相似度计算模型，计算出每个邻居与目标用户的时效相似度，最后依据时效相似度动态筛选目标用户的时效近邻。

2）可信近邻选取单元利用项目级信任计算模型对时效近邻的推荐能力进行推荐信度初估，之后利用比例-积分-微分控制器（Proportion Integration Differentiation，PID）的变体模型对推荐信度初估值进行重估，最后依据推荐信度重估值动态选取目标用户的可信近邻。

3）评分预测单元依据每个可信近邻的推荐信度计算出各自在目标项目上的推荐权重，之后利用他们在目标项目上的评分预测出目标用户对该项目的评分，进而向目标用户进行项目推荐。

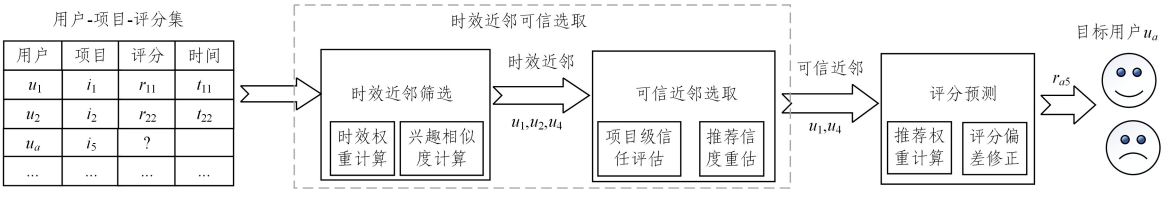


图1 方法框架

Fig. 1 Framework of the proposed method

3.3 时效近邻筛选

选取兴趣相似度高的用户作为目标用户近邻是提高推荐精度的关键，然而用户兴趣会随时间而改变，这要求兴趣相似性度量必须能够反映用户兴趣变化。基于皮尔逊相关系数（Pearson Correlation Coefficient，PCC）的相似性度量可处理评分等级膨胀[20]，非常适用于评分尺度存有差异的场合，为此本文拟选用PCC度量用户间的兴趣相似性。PCC的基本思想是利用共同评分项目上的用户评分协方差与标准偏差

之比来描述用户/项目之间的线性相关性。由于无法描述时间，因此其无法反映用户兴趣变迁。为度量兴趣时效相似度，对 PCC 模型改进如下：

$$TSD(u_a,u_i)=\frac{\sum_{i_k\in CRIS(u_a,u_i)}(r_{ak}*w(t_{ak})-\overline{r_a})*(r_{ik}*w(t_{ik})-\overline{r_i})}{\sqrt{\sum_{i_k\in CRIS(u_a,u_i)}(r_{ak}*w(t_{ak}))^2}\sqrt{\sum_{i_k\in CRIS(u_a,u_i)}(r_{ik}*w(t_{ik}))^2}} \tag{1}$$

其中，$TSD(u_a,u_i)$为目标用户 u_a与其邻居用户 $u_i\in NS(u_a)$之间的兴趣时效相似度；r_{ak}和 r_{ik}为 u_a和 u_i对项目 $i_k\in CRIS(u_a,u_i)$的评分；$\overline{r_a}$表示 u_a在 $RIS(u_a)$所有项目上的评分均值，$\overline{r_i}$表示 u_i在 $RIS(u_i)$所有项目上的评分均值；t_{ak}和 t_{ik}为评分 r_{ak}和 r_{ik}的生成时间(标准化后的时间)，$w(t_{ak})$和 $w(t_{ik})$为 r_{ak}和 r_{ik}的时效权重。

一般来讲，距离当前时间越近的用户评分越能代表该用户的当前兴趣，其在相似性计算时也要被赋予越高的权重。S 形函数 Logistic 单调递增，值域为(0,1)，常被用于描述时间衰变。本文将评分的时效权重 $w(t)$计算为：

$$w(t)=\text{Logistic}(t)=(1+e^{-t})^{-1};t\in[-1,1] \tag{2}$$

式(2)在实际使用时需采用 Min-Max 等数据标准化模型将项目评分时间映射到[-1,1]。

定理 1 对任意与目标用户 u_a存在共同评分项目的用户 u_i而言，他们在同一项目上的评分时间越近，则兴趣时效相似度就越大。

证明：令 u_a与 u_i存在 n 个共同评分项目$\{i_1,i_2,\cdots,i_n\}$，r_{aj}和 t_{aj}分别为 u_a对 i_j的评分和时间，r_{ij}和 t_{ij}分别为 u_i对 i_j的评分和时间，$\overline{r_a}$和$\overline{r_i}$分别为 u_a和 u_i的评分均值。u_a与 u_i的时效相似度为：

$$TSD(u_a,u_i)=\frac{\sum_{j=1}^{n}(r_{aj}*w(t_{aj})-\overline{r_a})(r_{ij}*w(t_{ij})-\overline{r_i})}{\sqrt{\sum_{j=1}^{n}(r_{aj}*w(t_{aj}))^2}\sqrt{\sum_{j=1}^{n}(r_{ij}*w(t_{ij}))^2}}$$

令 $r_{a1}=r_{a2}=\cdots=r_{an}$，$r_{i1}=r_{i2}=\cdots=r_{in}$，$t_{a1}=t_{a2}=\cdots=t_{an}$，$t_{i1}=t_{i2}=\cdots=t_{in}$，有：

$$\begin{aligned}TSD(u_a,u_i)&=\frac{(w(t_{a1})-1)(w(t_{i1})-1)}{w(t_{a1})*w(t_{i1})}\\&=(1-w(t_{a1})^{-1})(1-w(t_{i1})^{-1})\\&=(1-(1+e^{-t_{a1}}))(1-(1+e^{-t_{i1}}))\\&=\begin{cases}e^{-(2*t_{a1}+\delta)}, & t_{i1}=t_{a1}+\delta\\ e^{-(2*t_{i1}+\delta)}, & t_{a1}=t_{i1}+\delta\end{cases}\end{aligned}$$

故当 u_a与 u_i针对相同项目的评分时差 δ 越小，$TSD(u_a,u_i)$就会越大，证毕。

时效近邻选取算法如算法 1 所示。

算法 1 时效近邻选取算法 STeCN

输入：目标用户 u_a，评分矩阵 **R**

输出：u_a时效近邻集 TeNNS(u_a)

1. TeNNS(u_a)←Ø；$G_{TSD}(u_a)$←0；/*初始化*/
2. 计算 NS(u_a)；
3. for each $u_i\in$NS(u_a)
4. 计算 TSD(u_a,u_i)；/*计算 u_a 时效相似度*/
5. $G_{TSD}(u_a)$←$G_{TSD}(u_a)$+TSD(u_a,u_i)/|NS(u_a)|；/*更新阈值*/
6. end for
7. for each $u_i\in$NS(u_a)
8. if TSD(u_a,u_i)≥$G_{TSD}(u_a)$ /*筛选时效近邻*/
9. TeNNS(u_a)←TeNNS(u_a)∪{u_i}；
10. end if
11. end for
12. return TeNNS(u_a)；

算法 1 中第 2 步是获取目标用户 u_a的邻居集 $NS(u_a)$；第 3—6 步是计算 u_a与邻居 u_i的时效相似度和 u_a的时效相似度阈值 $G_{TSD}(u_a)$；第 7—11 步是依据 $G_{TSD}(u_a)$为目标用户 u_a筛选时效近邻集 $TeNNS(u_a)$。本文中取 $G_{TSD}(u_a)$为 u_a与邻居集中所有用户时效相似度的均值，即：

$$G_{TSD}(u_a)=\sum_{u_i\in NS(u_a)}TSD(u_a,u_i)/|NS(u_a)| \tag{3}$$

表 1 给出了目标用户 Mike 及其邻居 u_1—u_7在项目 i_1—i_5上的评分及评分(标准化)时间，分别采用传统 PCC 和带时效权重的 PCC 对 Mike 与 7 个邻居的相似度进行了计算。

表 1 用户之间的相似度计算

Table 1 Calculation of similarity between users

用户		项目 i_1		项目 i_2		项目 i_3		项目 i_4		项目 i_5		与 Mike 的相似度	
		评分	时间	评分	时间	评分	时间	评分	时间	评分	时间	传统 PCC	时效 PCC
Mike		5	0.10	1	0.15	-	-	3	0.15	3	0.10	-	-
邻居用户	u_1	4	0.10	1	0.20	2	0.15	2	0.30	2	0.22	**0.9682**	**0.4038**
	u_2	4	0.30	1	0.40	—	—	3	0.50	3	0.35	0.8689	0.3191
	u_3	1	0.50	3	0.60	2	0.55	1	0.65	1	0.70	−0.7906	−0.0514
	u_4	5	1.00	—	—	4	0.90	—	—	2	0.95	0.5937	−0.0321
	u_5	5	0.40	1	0.45	2	0.44	3	0.50	3	0.35	**0.9535**	**0.5544**
	u_6	5	1.00	1	0.90	2	0.95	3	0.97	3	0.99	**0.9535**	**0.3715**
	u_7	4	0.20	2	0.15	2	0.17	2	0.19	2	0.18	0.7906	**0.3086**

1)前者获得的结果依次为{0.9682,0.8689,−0.7906,0.5937,0.9535,0.9535,0.7906}(剔除相似度为负数的 u_3后，Mike 相似度阈值为 0.8548)，筛选出的 Mike 近邻集为$\{u_1,u_2,u_5,u_6\}$；

2)后者获得的结果依次为{0.4038,0.3191,−0.0514,−0.0321,0.5544,0.3715,0.3086}(Mike 时效相似度阈值 0.3209)，筛选出的 Mike 时效近邻集为$\{u_1,u_5,u_6\}$。

以 u_2为例，其和 u_1虽然在评分值上与 Mike 相似，但由于其与 Mike 的评分时间间隔大于 u_1与 Mike 的时间间隔，因此其被时效 PCC 方法剔除在 Mike 近邻之外。

3.4 可信近邻选取

兴趣时效相似度的采用改善了目标用户近邻的筛选质量，然而却无法保证所筛选到的近邻就一定可信。恶意用户 u_i可以在目标用户 u_a已评分项目上通过迎合 u_a的时效兴趣获取到时效近邻的身份，之后通过在 u_a未评分项目上实施策略评分，达到干扰或误导推荐结果的目的。为防范上述现象，此处提出基于推荐信度的可信近邻选取算法，拟利用其从目标用户的时效近邻集中进一步筛选出可信近邻(集)。

基于推荐信度的可信近邻选取过程为，首先利用改进的项目级信任计算模型对目标实体时效近邻的推荐能力进行推荐信度初估，接着利用改进的基于 PID 鲁棒控制技术的信度重估模型[21]对初估信度实施重估，最后依据重估信度从目标实体时效近邻中筛选出可信近邻。

1）推荐信度初估

令 u_a 为目标用户，u_i 为 u_a 的时效近邻，即 $u_i \in TeNNS(u_a)$，i_j 为 u_a 与 u_i 的共同评分项目，即 $i_j \in CRIS(u_a,u_i)$，r_{aj} 和 r_{ij} 分别为 u_a 和 u_i 对 i_j 的评分，$\overline{r_a}$ 和 $\overline{r_i}$ 为 u_a 和 u_i 的评分均值，$TSD(u_a,u_i)$ 为 u_a 和 u_i 的时效相似度，$CRIS(u_a,u_i)$ 中元素按 u_i 评分时间先后有序，且 i_j 是该集合中第 j 个项目。基于文献[18]，此处给出如下形式的推荐信度初估模型。

（1）u_i 预测 u_a 对 i_j 的评分 $\hat{r}_{aj}$，即

$$\hat{r}_{aj}=\overline{r_a}+\frac{(r_{ij}-\overline{r_i})\times TSD(u_a,u_i)}{|TSD(u_a,u_i)|} \tag{4}$$

（2）计算 u_a 对 u_i 预测的满意度 $SAT^j(u_a,u_i)$，即

$$SAT^j(u_a,u_i)=\begin{cases}1, & |\hat{r}_{aj}-r_{aj}|\leqslant\varepsilon\\ 0, & \text{else}\end{cases} \tag{5}$$

其中，常数 ε 为 u_a 定义的推荐偏差值，实验中取 ε 为经验值 1.2[18]。

（3）计算 u_i 的推荐信度原始值 $RRT^j(u_a,u_i)$，即

$$RRT^j(u_a,u_i)=\sum_{k=1}^{j}SAT^k(u_a,u_i)/j \tag{6}$$

其中，$1\leqslant j\leqslant|CRIS(u_a,u_i)|$。

2）推荐信度重估

u_i 评分的策略波动、无意识错误，以及不一致性，均会导致 $RRT^j(u_a,u_i)$ 与 u_a 对 u_i 推荐能力的真实满意度之间存在偏差。为修正该偏差，此处基于我们前期工作所提的信度重估模型[21]，对 RRT^j 重估如下：

$$\begin{aligned}RT^j(u_a,u_i)=&\alpha*RRT^j(u_a,u_i)+\beta*HRT_1^{j-1}(u_a,u_i)+\\&\gamma*DRT^j(u_a,u_i)+\delta*DRT^j(u_a,u_i)*\\&|SDRT^j(u_a,u_i)|\end{aligned} \tag{7}$$

其中，RRT^j，HRT_1^{j-1}，DRT^j，$SDRT^j$ 和 RT^j 分别表示推荐信度原始值、历史值、变化率、变化趋势以及重估值，α，β，γ 和 δ 为各部分的权重，且 $0<\alpha,\gamma,\beta,\delta<1$，同时设置上要求 β/α 正比于 $j-1$，γ 和 δ 遵循式(8)和式(9)。

$$\gamma=\begin{cases}\gamma_1, & DRT^j\geqslant 0\\ \gamma_2, & DRT^j<0\end{cases},\gamma_1\leqslant\gamma_2 \tag{8}$$

$$\delta=\begin{cases}\delta_1, & DRT^j*SDRT^j<0\\ \delta_2, & DRT^j*SDRT^j\geqslant 0\end{cases},\delta_1\leqslant\delta_2 \tag{9}$$

HRT_1^{j-1}，DRT^j 和 $SDRT^j$ 的计算如式(10)－式(12)所示，其中 $\rho(0<\rho\leqslant 1)$ 和 $\theta(0<\theta\leqslant 1)$ 分别为历史推荐信度及其变化率的关注因子。

$$HRT_1^{j-1}=\begin{cases}0, & j=1\\ \dfrac{\sum_{k=1}^{j-1}RT^k*\rho^{j-(k+1)}}{\sum_{k=1}^{j-1}\rho^{j-(i+1)}}, & j>1\end{cases} \tag{10}$$

$$DRT^j=\begin{cases}0, & j=1\\ RRT^j-HRT_1^{j-1}, & j>1\end{cases} \tag{11}$$

$$SDRT^j=\begin{cases}0, & j=1\\ DRT^j-\dfrac{\sum_{k=1}^{j-1}DRT^k*\theta^{j-(k+1)}}{\sum_{k=1}^{j-1}\theta^{j-(k+1)}}, & j>2\end{cases} \tag{12}$$

文献[21]中的信度重估模型的理论原型是 PID 模型，原始的 PID 模型利用输入的原始值（P 分量）、历史值（I 分量）和变化率（D 分量）对原输入进行值修正。而文献[21]对 PID 进行了扩展，引入二阶导数表征变化趋势，给出了基于信度原始值（P 分量）、历史值（I 分量）、变化率（D 分量）和变化趋势（SD 分量）的信度重估模型，提高了信度评估结果与真实值的逼近度。式(7)中，通过加权 D 分量（DRT^j）实现了对推荐能力突发波动的度量，通过加权平均 I 分量（HRT_1^{j-1}）、D 分量（DRT^j）和 SD 分量（$SDRT^j$）实现了对推荐能力改善与恶化的区分，通过加权平均 P 分量（RRT^j）与 I 分量（HRT_1^{j-1}）实现了对无意识错误推荐的容忍和一致性推荐能力的反映，通过加权平均 D 分量（DRT^j）与 SD 分量（$SDRT^j$）实现了对推荐能力变化趋势的跟踪，提高了推荐信度与真实推荐能力满意度的逼近度。

推荐信度评估如算法 2 所示。

算法 2 推荐信度评估 ERR

输入：目标用户 u_a，时效近邻 u_i，评分矩阵 **R**

输出：推荐信度 $RT(u_a,u_i)$

1. $RT(u_a,u_i)\leftarrow 0$；
2. if $CRIS(u_a,u_i)\neq\varnothing$
3. SortByTime($CRIS(u_a,u_i)$)；/*按时间排序*/
4. for each order $i_j\in CRIS(u_a,u_i)$ do
5. 依据式(4)计算 $\hat{r}_{aj}$；
6. 依据式(5)计算 $SAT^j(u_a,u_i)$；
7. 依据式(6)计算 $RRT^j(u_a,u_i)$；
8. 依据式(7)计算 $RT^j(u_a,u_i)$；
9. end for
10. $RT(u_a,u_i)\leftarrow RT^j(u_a,u_i)$；
11. end if
12. return $RT(u_a,u_i)$.

继 3.3 节中示例，若取 $\varepsilon=0.5$，$\alpha=0.2$，$\beta=0.8$，$\gamma_1=\delta_1=0.05$，$\gamma_2=\delta_2=0.2$，$\rho=\theta=1$，如表 2 所列，依据算法 2 可以求出 u_1，u_5 和 u_6 的推荐信度分别为 0.73，1.00 和 1.00。

表 2 推荐信度评估

Table 2 Evaluation on recommendation trust degree

用户		项目 i_1		项目 i_2		项目 i_3		项目 i_4		项目 i_5	
		推荐信度	评分时间	推荐信度	评分时间	推荐信度	评分时间	推荐信度	评分时间	推荐信度	评分时间
Mike		—	0.10	—	0.15	—	—	—	0.15	—	0.10
时效近邻	u_1	$1.00^{(j=1)}$	0.10	$0.60^{(j=2)}$	0.20	—	0.15	$\mathbf{0.73}^{(j=4)}$	0.30	$0.78^{(j=3)}$	0.22
	u_5	$1.00^{(j=2)}$	0.40	$1.00^{(j=3)}$	0.45	—	0.44	$\mathbf{1.00}^{(j=4)}$	0.50	$1.00^{(j=1)}$	0.35
	u_6	$1.00^{(j=2)}$	0.40	$1.00^{(j=3)}$	0.45	—	0.44	$\mathbf{1.00}^{(j=4)}$	0.50	$1.00^{(j=1)}$	0.35

3）可信近邻确定

从 u_a 时效近邻中选取推荐信度最高的 $topK$ 个用户作为 u_a 的可信近邻集 $TNNS(u_a)$。

$$TNNS(u_a)=\{u_i\,|\,RT(u_a,u_i)\in topK(\{RT(u_a,u_i)\,|\,u_i\in TeNNS(u_a)\})\} \tag{13}$$

算法 3 给出了可信近邻动态选取过程。

算法 3 可信近邻筛选 STNN

输入：目标用户 u_a，时效近邻集 TeNNS(u_a)

输出：可信近邻集 TNNS(u_a)

1. TNNS(u_a)←∅；
2. for each $u_i \in$ TeNNS(u_a) do
3. 调用算法 2 计算 RT(u_a, u_i)；
4. end for
5. 依据式(13)计算可信近邻 TNNS(u_a)；
6. return TNNS(u_a).

继上例，若 $K=2$，依据算法 3 可以遴选到 Mike 的可信近邻为 u_5 和 u_6。第 4 节实验中取 $topK$ 为 90%。

3.5 评分预测及推荐

针对目标项目 i_j，依据 $TNNS(u_a)$ 中用户对 i_j 的评分，预测出 u_a 在 i_j 上的评分 $\hat{r}_{aj}$。

令 $TRU(u_a)=\{u_i \mid u_i \in TNNS(u_a) \wedge r_{ij} \neq \emptyset\}$，则

$$\hat{r}_{aj}=\begin{cases}\overline{r_a}, & \text{if } TRU(u_a)=\emptyset \\ \overline{r_a}+\dfrac{\sum\limits_{u_i \in TRU(u_a)}(r_{ij}-\overline{r_i})\times RT(u_a,u_i)}{\sum\limits_{u_i \in TRU(u_a)} RT(u_a,u_i)}, & \text{else}\end{cases} \tag{14}$$

基于时效近邻可信选取的协同过滤推荐流程描述如下：

1)依据评分矩阵 R，选取出目标用户 u_a 的邻居集 $NS(u_a)$；

2)依据兴趣时效相似度 $TSD(u_a, *)$，利用算法 1 从邻居集 $NS(u_a)$ 中提取出 u_a 的时效近邻集 $TeNNS(u_a)$；

3)依据算法 2 计算到的推荐信度 $RT(u_a, *)$，利用算法 3 从 $TeNNS(u_a)$ 中进一步筛选出 u_a 的可信近邻集 $TNNS(u_a)$；

4)依据式(14)预测出 u_a 对项目 i_j 的评分 $\hat{r}_{aj}$，进而完成项目推荐。

基于时效近邻可信选择策略的协同过滤推荐算法如算法 4 所示。

算法 4 协同过滤推荐算法 CF-TeCNTS

输入：目标用户 u_a，目标项目 i_j，评分矩阵 R

输出：u_a 对 i_j 的预测评分 $\hat{r}_{aj}$

1. 调用算法 1 计算 TeNNS(u_a)；
2. 调用算法 3 计算 TNNS(u_a)；
3. 依据式(14)预测 u_a 对 i_j 的评分 $\hat{r}_{aj}$；
4. returnr $\hat{r}_{aj}$.

继 3.4 节中示例，利用算法 4 可以求出 Mike 对 i_3 的预测评分为 2。

4 实验及分析

4.1 实验数据与平台

实验采用了 MovieLens 的 movielens-100K 数据集(约 1000名用户在近 1700 部电影上约 10 万次评分，稀疏率约为 96.695%)和 ml-latest-small 数据集(约 610 名用户在近 9742 部电影上 100837 次评分，稀疏率约为 98.303%)，以及亚马逊 video game 的 ratings only 数据集(约 50000 名用户在近 830000 个游戏上近 2600000 次评分，稀疏率约为 99.99%)。

实验机器配置如下：CPU(E5-1230V3，2.40 GHz)、内存(47GB)、硬盘(13TB)、GPU(RTX3090 24GB)，具体实现采用了 python 3.9.7 和 math 库。实验中推荐信度重估的超参数设置为 $\alpha=0.2$，$\beta=0.8$，$\gamma_1=\delta_1=0.05$，$\gamma_2=\delta_2=0.2$，$\rho=\theta=1$。

4.2 性能评估指标

本文采用的评估指标主要有 2 个。

1)平均绝对误差(Mean Absolute Error，MAE)

MAE 用于衡量预测评分与真实评分之间的误差。计算公式为：

$$MAE=\sum_{j=1}^{n}\frac{|\hat{r}_{aj}-r_{aj}|}{n} \tag{15}$$

其中，$\hat{r}_{aj}$ 和 r_{aj} 分别为目标用户 u_a 在项目 i_j($1 \leqslant j \leqslant n$)上的预测评分与真实评分。MAE 值越小，算法推荐精度就越高。

2)平均预测增量(Average Prediction Shift，APS)

APS 用于衡量项目受攻击前后系统预测的增量。计算公式为：

$$APS=\sum_{j=1}^{n}\frac{(\hat{r}'_{aj}-\hat{r}_{aj})}{n} \tag{16}$$

其中，$\hat{r}_{aj}$ 和 $\hat{r}'_{aj}$ 分别为项目受攻击前后目标用户 u_a 在项目 i_j($1 \leqslant j \leqslant n$)上的预测评分。APS 值越小，算法抗攻击能力就越强。

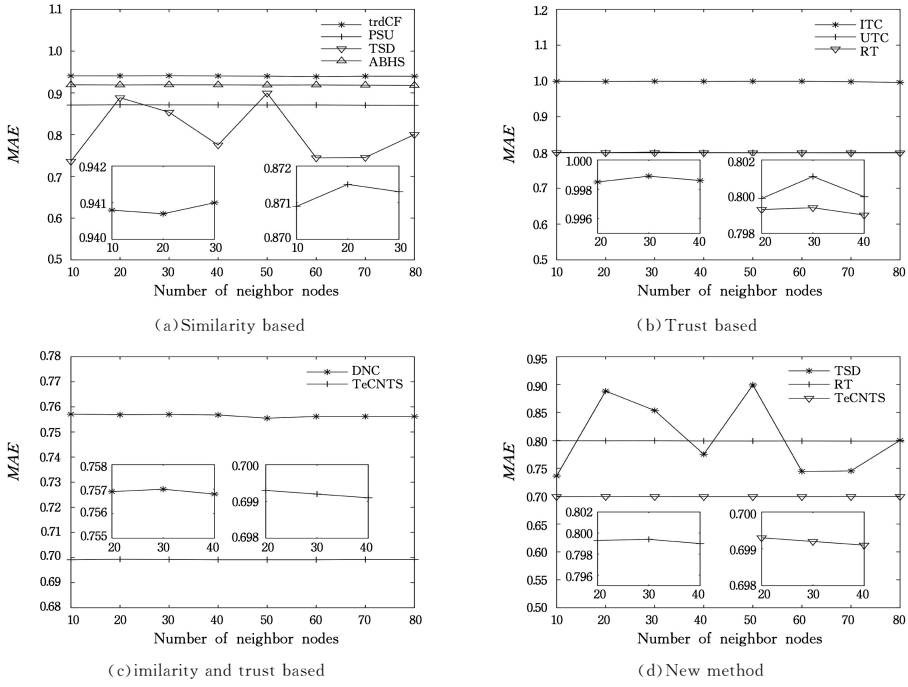
此外，对攻击用户检测能力的评估拟采用 3 种常用的评估分类器度量标准：查准率、查全率和调和平均值。具体地，设 TP 是被正确识别为攻击用户的样本数，FN 是被误判为非攻击用户的样本数，FP 是被误判为攻击用户的样本数，则攻击用户查全率 $R=TP/(TP+FN)$，攻击用户查全率 $P=TP/(TP+FP)$，调和平均值 $F_1=2*P*R/(P+R)$。

4.3 推荐精度评估

采用 MAE 评估指标，将本文所提的基于时效近邻选取策略的 CF 推荐(CF-TSD)、基于可信近邻确定策略的 CF 推荐(CF-RT)、基于时效近邻可信选取策略的 CF 推荐(CF-TeCNTS)，与文献[18]给出的基于兴趣相似用户集的 CF 推荐(CF-PSU)、基于用户信任计算模型的 CF 推荐(CF-UTC)和基于双重邻居选取策略 CF 推荐(CF-DNC)[18]，以及与基于社会信息和动态时间窗口的 CF 推荐(CF-ABHS)[13]、经典的 O'Donovan 项目级信任策略的 CF 推荐(CF-ITC)[22]和传统的 CF 推荐(CF)这 9 个策略(合计 3 类)的推荐精度做了比较。通过选取拥有 8 种不同数量邻居的用户作为目标用户，图 2 和图 3 分别给出了在 MovieLens-100K 数据集和亚马逊 video game 数据集上各 CF 策略的推荐 MAE 情况，可以看出 CF-TSD，CF-RT 和 CF-TeCNTS 策略在同类策略中具有更低的 MAE。

以图 2 为例(图 3 类似)，可以看出：1)从图 2(a)可以看出，在基于相似度筛选近邻的策略中，CF-ABHS，CF-PSU，CF-TSD 的 MAE 值均低于 CF(均值分别降低了 2.2655%，7.3616%和 14.3482%)，表明选择近邻时滤掉低相似度的用户可以改善推荐精度；CF-TSD 的 MAE 均值低于 CF-PSU(降低了7.5418%)，表明在相似度评估时考虑用户兴趣变化能够提升推荐精度；CF-TSD 的 MAE 均值低于 CF-ABHS(降低了12.3628%)，表明在相似度计算中将对兴趣变化的关注由短期变更为长期可提升推荐精度。2)从图 2(b)可以看出，在基于信任度筛选近邻的策略中，CF-UTC 和 CF-RT 获得的 MAE 低于 CF-ITC(均值降低了 19.8553%和 19.9555%)，表明将用户信度从大众对其推荐能力的全局评价转变为目标

用户个人对其的局部评价能够提升近邻筛选的针对性，从而提高推荐精度。再者，CF-RT 在 *MAE* 及其波动程度上均低于 CF-UTC（分别降低了 0.1250%和 68.7500%），表明依据重估后的推荐信度来筛选近邻，可以提高近邻筛选的质量，改善推荐精度和精度稳定性。3）从图 2（c）可以看出，在基于相似度和信任度筛选近邻的策略中，CF-TeCNTS 获得的 *MAE* 及其波动程度均低于 CF-DNC（分别降低了 7.5778%和 80.0000%），表明依据本文提出的时效近邻可信选取策略筛选近邻，可以获得更稳定、更高的推荐精度。4）从图 2（d）可以看出，CF-TeCNTS 所表现出来的对用户兴趣变迁的敏感反应来源于 CF-TSD，推荐精度稳定性来源于 CF-RT。

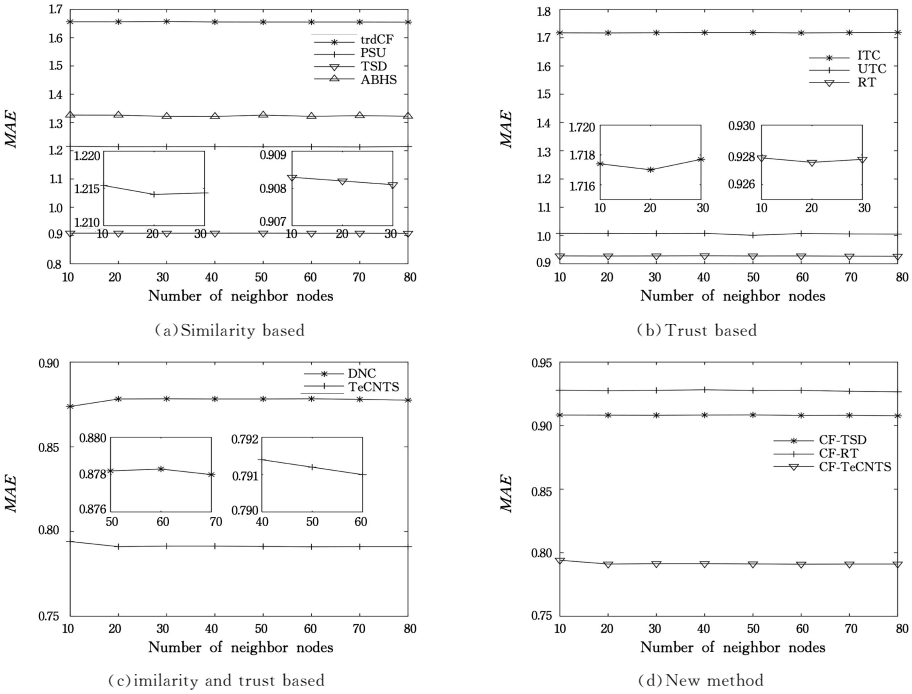


（a）Similarity based　（b）Trust based

（c）imilarity and trust based　（d）New method

图 2　使用 MovieLens-100K 数据集时不同策略的 MAE 对比

Fig. 2　MAE comparison of different strategies on MovieLens-100K dataset



（a）Similarity based　（b）Trust based

（c）imilarity and trust based　（d）New method

图 3　使用亚马逊 video game 数据集时不同策略的 MAE 对比

Fig. 3　MAE comparison of different strategies on Amazon video game dataset

4.4　抗攻击力评估

为了评估本文所提策略的抗攻击力，分别向 ml-latest-small 数据集和亚马逊 video game 数据集中注入随机攻击数据。选取邻居用户数为 40，攻击规模（虚假用户占比）分别为 1%，3%，5%，10%，25%，填充规模（每个虚假用户的评分数占比）分别为 1%，2%，5%，10%。

表 3 和表 4 分别给出了 9 种策略在 2 个数据集上的推荐偏差 MAE。

表 3 使用 ml-latest-small 数据集时随机攻击下不同策略的 MAE 对比

Table 3 MAE comparison of different strategies under random attack on ml-latest-small dataset

攻击规模/%	填充规模/%	*MAE*								
		基于相似度				基于信任度			基于相似度和信任度	
		CF	CF-ABHS	CF-PSU	**CF-TSD**	CF-ITC	CF-UTC	**CF-RT**	CF-DNC	**CF-TeCNTS**
1	1	1.9083	0.8343	0.8545	0.8326	2.4268	0.8085	0.8084	0.8052	0.7408
	2	1.9108	0.8225	0.8461	0.8216	2.6718	0.8093	0.8092	0.8049	0.7446
	5	2.0061	0.8262	0.8392	0.8341	2.7901	0.8083	0.8082	0.8052	0.7481
	10	1.9984	0.8297	0.8412	0.8288	2.9019	0.8081	0.8079	0.8047	0.7449
3	1	2.0252	0.8480	0.8633	0.8291	2.8176	0.8199	0.8198	0.8159	0.7479
	2	2.1312	0.8555	0.8574	0.8362	2.8137	0.8208	0.8207	0.8175	0.7535
	5	2.0425	0.8639	0.8648	0.8482	2.6848	0.8228	0.8222	0.8156	0.7621
	10	2.0956	0.8743	0.9167	0.8611	2.8818	0.8213	0.8212	0.8165	0.7677
5	1	2.2685	0.8740	0.9128	0.8483	2.9505	0.8325	0.8324	0.8278	0.7648
	2	2.3936	0.8782	0.8709	0.8644	2.6184	0.8325	0.8325	0.8293	0.7713
	5	2.3338	0.8873	1.0212	0.8571	2.9943	0.8348	0.8347	0.8311	0.7749
	10	2.3195	0.8953	0.8801	0.8762	2.7871	0.8319	0.8318	0.8287	0.7781
10	1	2.4073	0.9305	0.9111	0.8757	3.2456	0.8623	0.8622	0.8591	0.7801
	2	2.4445	0.9527	0.9146	0.8968	3.3077	0.8627	0.8626	0.8596	0.7995
	5	2.6219	0.9536	0.9202	0.9291	3.1911	0.8625	0.8624	0.8577	0.8244
	10	2.4498	0.9615	0.9266	0.9393	3.1876	0.8647	0.8646	0.8633	0.8284
25	1	3.4749	1.0388	0.9828	0.9221	3.7277	0.9301	0.9301	0.9307	0.8157
	2	3.8061	1.0600	0.9903	0.9788	3.5059	0.9288	0.9288	0.9274	0.8633
	5	3.1920	1.0651	1.0166	0.9861	3.6956	0.9303	0.9302	0.9274	0.8749
	10	3.7292	1.0613	1.0396	1.0389	3.7475	0.9318	0.9317	0.9319	0.8881

表 4 使用亚马逊 video game 数据集时随机攻击下不同策略的 MAE 对比

Table 4 MAE comparison of different strategies under random attack on Amazon video game dataset

攻击规模/%	填充规模/%	*MAE*								
		基于相似度				基于信任度			基于相似度和信任度	
		CF	CF-ABHS	CF-PSU	**CF-TSD**	CF-ITC	CF-UTC	**CF-RT**	CF-DNC	**CF-TeCNTS**
1	1	2.9654	1.1694	1.4301	0.9915	2.7008	0.8551	0.8474	0.8139	0.8041
	2	3.3816	1.1692	1.4665	0.9992	3.0878	0.8561	0.8545	0.8153	0.7936
	5	3.4144	1.1663	1.4752	1.0125	3.4678	0.8591	0.8556	0.8185	0.7966
	10	3.4889	1.1857	1.5284	1.0078	3.8739	0.8562	0.8587	0.8175	0.7921
3	1	4.1495	1.1359	1.5470	1.0048	3.1719	0.8636	0.8559	0.8232	0.7976
	2	3.8912	1.1948	1.6546	1.0181	3.4684	0.8624	0.8631	0.8279	0.7982
	5	4.0532	1.2025	1.6225	1.0435	4.3056	0.8702	0.8620	0.8319	0.8089
	10	3.9852	1.1870	1.6247	1.0466	4.6948	0.8686	0.8698	0.8309	0.7920
5	1	3.7104	1.1787	1.6504	0.9998	3.8043	0.8798	0.8683	0.8374	0.8031
	2	4.1288	1.2191	1.6905	1.0424	4.0034	0.8777	0.8794	0.8380	0.8047
	5	4.2164	1.1295	1.7163	1.0742	4.9394	0.8748	0.8773	0.8416	0.8102
	10	4.4311	1.2177	1.7465	1.0769	5.0178	0.8783	0.8745	0.8403	0.8077
10	1	4.4436	1.1899	1.7727	0.9961	4.7099	0.8987	0.8780	0.8605	0.8286
	2	5.2004	1.2682	1.7962	1.0910	5.3610	0.9021	0.8985	0.8672	0.8374
	5	5.0295	1.3337	1.8589	1.1557	5.1726	0.9059	0.9018	0.8664	0.8438
	10	4.7843	1.3764	1.7410	1.1539	6.4365	0.9057	0.9056	0.8638	0.8336
25	1	5.5770	1.3924	1.8613	1.0294	4.7405	0.9635	0.9054	0.9261	0.8345
	2	5.4120	1.4131	1.9491	1.1811	4.7662	0.9650	0.9634	0.9304	0.8369
	5	5.4855	1.3865	1.9700	1.3291	5.1447	0.9587	0.9648	0.9237	0.8619
	10	6.8221	1.3980	2.0154	1.3225	6.0737	0.9603	0.9585	0.9626	0.9236

以表 3 为例，可以看出：1）在相同填充规模（攻击规模）下，随攻击规模（填充规模）的增大，9 种策略的推荐 *MAE* 基本上均呈现上升趋势，这表明系统中攻击越多，推荐质量也会越差；且相较于攻击规模对推荐 MAE 的影响力而言，填充规模的影响力更大。2）在基于相似度筛选近邻的策略中，CF，CF-ABHS，CF-PSU 和 CF-TSD 策略的推荐 MAE 逐渐降低（后者均值上依次比前者降低了 63.048 8%，0.233 2% 和 3.095 2%），表明本文对长期时效兴趣相似度低的用户进行过滤的近邻选择方法可以降低随机攻击下的推荐偏差。3）在基于信任度筛选近邻的策略中，CF-ITC，CF-UTC 和 CF-RT 策略的推荐 *MAE* 逐渐降低（后者在均值上依次比前者降低 72.067 9% 和 0.013 5%），表明本文提出的利用可一致性表征用户推荐满意度的重估信度来滤掉低信度用户的近邻选择方法可以降低随机攻击下的推荐偏差。4）在基于相似度和信任度筛选近邻的策略中，CF-TeCNTS 的推荐 *MAE* 在均值上比 CF-DNC 降低了 6.995 5%，这表明本文融合 CF-TSD 和 CF-RT 两种策略所提的新方法在随机攻击面前具有更低的推荐偏差。

与上述类似，表 5 和表 6 分别列出了 9 种策略在 2 个数据集上的平均预测增量 APS。以表 5 为例，可以看出：1）在相同填充规模（攻击规模）下，随攻击规模（填充规模）的增大，9 种策略的 *APS* 基本上呈现上升趋势，这表明系统中攻击越多，预测偏差也会越大；而且，相较于填充规模对预测偏差的影响程度而言，攻击规模的影响力表现得更大。2）在基于相似度筛选近邻的策略中，CF，CF-PSU，CF-ABHS 和 CF-TSD 4 种策略的 APS 逐渐降低（后者在均值上依次比前者降低

72.8585%，29.883%和47.9572%）；在基于信任度筛选近邻的策略中，CF-ITC，CF-UTC和CF-RT 3个策略的APS逐渐降低（后者在均值上依次比前者降低92.4772%和1.7578%）；在基于相似度和信任度筛选近邻的策略中，CF-TeCNTS的*APS*在均值上比CF-DNC降低了11.4286%。以上表明，本文通过在用户相似度中引入对长期兴趣变化的关注，在信度评估中引入对用户推荐能力波动的关注，可使所提策略在随机攻击面前拥有更低的预测偏差。

表5　使用ml-latest-small数据集时随机攻击下不同策略的预测偏差APS对比

Table 5　APS comparison of different strategies under random attack on ml-latest-small dateset

攻击规模/%	填充规模/%	*APS*								
		基于相似度				基于信任度			基于相似度和信任度	
		CF	CF-ABHS	CF-PSU	**CF-TSD**	CF-ITC	CF-UTC	**CF-RT**	CF-DNC	**CF-TeCNTS**
1	1	0.1034	0.0468	0.1239	0.0110	0.0605	0.0078	0.0076	0.0074	0.0023
	2	0.1062	0.0350	0.1157	0.0050	0.3050	0.0087	0.0084	0.0072	0.0057
	5	0.2013	0.0387	0.1102	0.0127	0.4237	0.0077	0.0074	0.0074	0.0089
	10	0.1945	0.0422	0.1122	0.0091	0.5351	0.0076	0.0071	0.0071	0.0058
3	1	0.2209	0.0605	0.1328	0.0097	0.4509	0.0190	0.0190	0.0177	0.0083
	2	0.3264	0.0680	0.1268	0.0153	0.4470	0.0205	0.0199	0.0195	0.0141
	5	0.2380	0.0764	0.1340	0.0278	0.3180	0.0225	0.0214	0.0176	0.0224
	10	0.2911	0.0868	0.1860	0.0432	0.5150	0.0212	0.0204	0.0185	0.0277
5	1	0.4635	0.0865	0.1815	0.0313	0.5839	0.0323	0.0316	0.0347	0.0238
	2	0.5891	0.0907	0.1396	0.0464	0.2518	0.0323	0.0317	0.0369	0.0305
	5	0.5292	0.0998	0.2897	0.0392	0.6278	0.0350	0.0339	0.0388	0.0338
	10	0.5150	0.1078	0.1487	0.0583	0.4207	0.0320	0.0310	0.0366	0.0374
10	1	0.6024	0.1430	0.1797	0.0580	0.8784	0.0622	0.0614	0.0618	0.0394
	2	0.6395	0.1652	0.1859	0.0791	0.9410	0.0629	0.0618	0.0621	0.0586
	5	0.8172	0.1661	0.1887	0.1117	0.8242	0.0628	0.0616	0.0606	0.0630
	10	0.6452	0.1740	0.1954	0.1218	0.8219	0.0649	0.0636	0.0664	0.0702
25	1	1.6702	0.2513	0.2515	0.1047	1.3610	0.1306	0.1293	0.1381	0.0747
	2	2.0013	0.2725	0.2588	0.1621	1.1396	0.1295	0.1280	0.1355	0.1220
	5	1.3873	0.2776	0.2854	0.1702	1.3259	0.1312	0.1294	0.1355	0.1333
	10	1.9244	0.2738	0.3084	0.2171	1.3809	0.1330	0.1309	0.1397	0.1469

表6　使用亚马逊video game数据集时随机攻击下不同策略的预测偏差APS对比

Table 6　APS comparison of different strategies under random attack on Amazon video game dataset

攻击规模/%	填充规模/%	*APS*								
		基于相似度				基于信任度			基于相似度和信任度	
		CF	CF-ABHS	CF-PSU	**CF-TSD**	CF-ITC	CF-UTC	**CF-RT**	CF-DNC	**CF-TeCNTS**
1	1	−0.0820	−0.0002	0.0634	0.0117	0.1081	0.0070	0.0071	0.0056	0.0197
	2	0.3341	−0.0032	0.0998	0.0194	0.4952	0.0081	0.0082	0.0070	0.0092
	5	0.3669	0.0163	0.1085	0.0326	0.8752	0.0111	0.0113	0.0103	0.0123
	10	0.4414	−0.0336	0.1616	0.0279	1.2813	0.0082	0.0085	0.0092	0.0078
3	1	1.1021	0.0254	0.1803	0.0249	0.5793	0.0156	0.0157	0.0149	0.0133
	2	0.8438	0.0331	0.2879	0.0382	0.8758	0.0144	0.0146	0.0196	0.0139
	5	1.0057	0.0175	0.2558	0.0636	1.7130	0.0222	0.0224	0.0236	0.0245
	10	0.9377	0.0092	0.2580	0.0667	2.1021	0.0206	0.0209	0.0226	0.0076
5	1	0.6630	0.0497	0.2837	0.0199	1.2117	0.0318	0.0320	0.0292	0.0187
	2	1.0813	−0.0400	0.3238	0.0625	1.4108	0.0296	0.0299	0.0297	0.0204
	5	1.1689	0.0483	0.3496	0.0943	2.3468	0.0268	0.0271	0.0333	0.0259
	10	1.3837	0.0205	0.3798	0.0970	2.4252	0.0303	0.0306	0.0320	0.0234
10	1	1.3961	0.0988	0.4060	0.0163	2.1173	0.0507	0.0511	0.0522	0.0442
	2	2.1530	0.1642	0.4295	0.1111	2.7684	0.0541	0.0544	0.0589	0.0530
	5	1.9821	0.2070	0.4922	0.1758	2.5800	0.0579	0.0582	0.0581	0.0594
	10	1.7369	0.2230	0.3743	0.1740	3.8438	0.0576	0.0580	0.0555	0.0492
25	1	2.5296	0.2436	0.4946	0.0495	2.1479	0.1155	0.1160	0.1179	0.0501
	2	2.3646	0.2170	0.5824	0.2012	2.1735	0.1170	0.1174	0.1221	0.0525
	5	2.4380	0.2432	0.6033	0.3492	2.5521	0.1107	0.1111	0.1154	0.0775
	10	3.7747	0.2086	0.6487	0.3426	3.4811	0.1123	0.0763	0.1543	0.1393

4.5　攻击用户识别能力评估

采用ml-latest-small数据集，设定攻击规模与填充规模分别为5%与10%（标记为①）、10%与10%（标记为②）、25%与5%（标记为③），以及25%与10%（标记为④），设定邻居用户数为40，表7和表8比较了4种随机攻击场景下9种策略的目标用户数、攻击用户识别数、被过滤用户数、攻击用户查全率、查准率和调和平均值。可以看出，在填充规模一定时，随着攻击规模的变大（由①到②到④），攻击用户的识别数增多；但当攻击规模一定时，随着填充规模的变大（由③到④），攻击用户识别数却在降低，这表明攻击规模的变小或填充规模的变大，均会加大此类协同过滤推荐策略识别攻击用户的难度。

1）CF和CF-ABHS不具备攻击用户识别能力，而CF-PSU和CF-TSD均识别到了数量不等的攻击用户，这表明在基于兴趣相似度的近邻筛选中，依据相似度阈值进行用户筛选可以过滤掉攻击用户。相比CF-PSU，CF-TSD的

攻击用户识别数分别提升了 10.6383%(由①到②到④)和28.9474%(由③到④),攻击用户查全率相应提升了1.6911%和28.9357%。

2)CF-ITC 不具备攻击用户识别能力,但 CF-RT 与 CF-UTC 均检测到了一定数量的攻击用户,这表明基于信任度进行近邻筛选能够有效过滤掉攻击用户。与 CF-UTC 相比,CF-RT 在攻击用户识别数上分别提升了84.9315%(由①到②到④)和 3.6765%(由③到④),且对应的攻击用户查全率在填充规模恒定、攻击规模较小(5%和 10%)时显著提升。

3)与 CF-DNC 相比,CF-TeCNTS 在攻击检测数上分别提升了 3.2680%(由①到②到④)和 29.1667%(由③到④),攻击用户查全率分别提升了 2.2880%和29.1649%。以上实验表明本文所提策略具有更强的攻击用户识别能力。

表 7 随机攻击下不同策略的攻击用户检测能力对比(1)

Table 7 Comparison of attacker identification ability of different strategies under random attack(1)

策略名称	目标用户数				攻击用户识别数				被过滤用户数			
	①	②	③	④	①	②	③	④	①	②	③	④
CF	640	671	741	741	0	0	0	0	1	1	22	22
CF-PSU	624	636	694	710	12	32	64	50	17	36	69	53
CF-ABHS	315	346	437	437	0	0	0	0	326	326	326	326
CF-TSD	281	306	345	357	14	22	79	68	360	366	418	406
CF-ITC	630	667	755	752	0	0	0	0	11	5	8	11
CF-UTC	547	578	603	603	0	7	68	68	94	94	160	160
CF-RT	576	603	602	603	27	40	73	68	65	69	161	160
CF-DNC	574	603	603	604	29	54	74	70	67	69	160	159
CF-TeCNTS	253	275	311	322	29	56	113	73	388	397	452	441

表 8 随机攻击下不同策略的攻击用户检测能力对比(2)

Table 8 Comparison of attacker identification ability of different strategies under random attack(2)

策略名称	查准率				查全率				F_1调和平均值			
	①	②	③	④	①	②	③	④	①	②	③	④
CF	0	0	0	0	0	0	0	0	—	—	—	—
CF-PSU	0.7059	0.8889	0.9275	0.9434	0.3871	0.5161	0.4183	0.3268	0.5000	0.6531	0.5766	0.4854
CF-ABHS	0	0	0	0	0	0	0	0	—	—	—	—
CF-TSD	0.0389	0.0601	0.1890	0.1675	0.4516	0.3548	0.5163	0.4444	0.0716	0.1028	0.2767	0.2433
CF-ITC	0	0	0	0	0	0	0	0	—	—	—	—
CF-UTC	0	0.0532	0.4250	0.4250	0	0.1129	0.4444	0.4444	—	0.0723	0.4345	0.4345
CF-RT	0.4154	0.5797	0.4534	0.4250	0.8710	0.6452	0.4771	0.4444	0.5625	0.6107	0.4650	0.4345
CF-DNC	0.4328	0.7826	0.4625	0.4403	0.9355	0.8710	0.4837	0.4575	0.5918	0.8244	0.4728	0.4487
CF-TeCNTS	0.0747	0.1411	0.2500	0.1655	0.9355	0.9032	0.7386	0.4771	0.1384	0.2440	0.3736	0.2458

结束语 本文提出了一种基于时效近邻可信选取策略的协同过滤推荐方法,该方法在目标用户近邻筛选过程中充分考虑了用户兴趣异常变化和推荐能力波动这两个因素,通过采用时效近邻筛选、可信近邻选取和目标用户评分预测 3 个推荐策略,实现了向目标用户的精准推荐,有效缓解了传统协同过滤推荐在数据稀疏情况下存在的推荐精度低和抗攻击能力差的问题。由于用户评分行为的改变诱因涉及到价值观、个体需要和情绪等诸多因素,进一步提高所提策略的推荐精度和抗攻击力,还需要从用户项目评分数据中深度挖掘与分析造成用户兴趣变迁和推荐能力变化的深层次规律,这将是论文下一步的研究重点。

参 考 文 献

[1] PATEL K, PATEL H B. A state-of-the-art survey on recommendation system and prospective extensions[J]. Computers and Electronics in Agriculture, 2020, 178: 105779.

[2] UMAIR J, KAMRAN S, HAMEED I A, et al. A Review of content-based and context-based recommendation systems[J]. International Journal of Emerging Technologies in Learning (iJET), 2021, 16(3): 274-306.

[3] WANG J, LAN Y X, WU C Y. Survey of recommendation based on collaborative filtering[C]// 3rd International Conference on Electrical, Mechanical and Computer Engineering. Bristol: Journal of Physics: Conference Series, 2019: 012078.

[4] WU Z, LI C S, CAO J, et al. On scalability of association-rule-based recommendation: a unified distributed-computing framework[J]. ACM Transactions on the Web(TWEB), 2020, 14(3): 1-21.

[5] ZIHAYAT M, AYANSO A, ZHAO X, et al. A utility-based news recommendation system[J], Decision Support Systems, 2019, 117: 14-27.

[6] TARUS J K, NIU Z D, MUSTAFA G. Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning[J]. Artificial Intelligence Review, 2018, 50: 21-48.

[7] CAI X JN, HU M, ZHAO P, et al. A hybrid recommendation system with many-objective evolutionary algorithm[J]. Expert Systems with Applications, 2020, 159: 113648.

[8] VINAGRE J, JORGE A M, GAMA J. An overview on the exploitation of time in collaborative filtering[J]. WIREs Data Mining and Knowledge Discovery, 2015, 5(5): 195-215.

[9] ZHAO J Y, ZHUANG F Z, AO X, et al. Survey of collaborative filtering recommender systems[J]. Journal of Cyber Security, 2021, 6(5): 17-34.

[10] LIU X J. An improved clustering-based collaborative filtering recommendation algorithm[J]. Cluster Computing, 2017, 20(2): 1281-1288.

[11] DONG C L, KE X S. Study on collaborative filtering algorithm based on user interest change and comment[J]. Computer Science, 2018, 45(3): 213-217, 246.

[12] XU G X,TANG Z J,MA C,et al. A collaborative filtering recommendation algorithm based on userconfidence and time context[J]. Journal of Electrical and Computer Engineering,2019,2019:1-12.

[13] LI D,WANG C,LI L,et al. Collaborative filtering algorithm with social information and dynamic time windows[J]. Applied Intelligence,2022,52:5261-5272.

[14] CHEN T,ZHU Q,ZHOU M X,et al. Trust-based recommendation algorithm in social network[J]. Journal of Software,2017,28(3):721-731.

[15] WANG W,CHEN J Y,WANG J Z,et al. Trust-enhanced collaborative filtering for personalized pointof interests recommendation[J]. IEEE Transactions on Industrial Informatics,2020,16(9):6124-6132.

[16] WANG F,ZHU H B,SRIVASTAVA G,et al. Robust collaborative filtering recommendation with user-item-trust records [J]. IEEE Transactions on Computational Social Systems,2022,9(4):986-996.

[17] CAI X J,TAN W A. Improved collaborative filtering algorithm combining similarity and trust[J]. Computer Science,2022,49(S1):238-241.

[18] JIA D Y,ZHANG F Z. A collaborative filtering recommendation algorithm based on double neighbor choosing strategy[J]. Journal of Computer Research and Development,2013,50(5):1076-1084.

[19] REZAIMEHR F,DADKHAH C. A survey of attack detection approaches in collaborative filtering recommender systems[J]. Artificial Intelligence Review,2020,54:2011-2066.

[20] SARANYA K G,SADASIVAM G S,CHANDRALEKHA M. Performance comparison of different similarity measures for collaborative filtering technique[J]. Indian Journal of Science and Technology,2016,9(29):1-8.

[21] HAN Z G,FENG X,CHEN G. SDN based e-mail repudiation source restraining method[J]. Journal on Communications,2016,37(9):55-67.

[22] O'DONOVAN J,SMYTH B. Trust in recommender systems [C]// Proceedings of the 10th International Conference on Intelligent User Interfaces(IUI'05). New York:ACM,2005:167-174.

HAN Zhigeng, born in 1976, Ph.D, associate professor, postgraduate supervisor, is a member of China Computer Federation. His main research interests include recommendation system security and intelligent audit.

FAN Yuanzhe, born in 1998, postgraduate. His main research interests include recommendation system security and intelligent audit.