科学通报 2014 年 第 59 卷 第 7 期: 551 ~ 560
评 述 www.scichina.com csb.scichina.com
《中国科学》杂志社
SCIENCE CHINA PRESS

推荐系统托攻击模型与检测技术

伍之昂[①], 王有权[②], 曹杰[①②*]

① 南京财经大学江苏省电子商务重点实验室, 南京 210003;
② 南京理工大学计算机科学与工程学院, 南京 210094
* 联系人, E-mail: Jie.Cao@njue.edu.cn

2013-01-30 投稿, 2013-04-23 接受
国家自然科学基金(61103229, 71072172, 61003074)、国家科技支撑计划(2013BAH16F00)、江苏省属高校自然科学研究重大项目(12KJA520001)、江苏高校优势学科建设工程和江苏省自然科学基金(BK2012863)资助

摘要 针对协同过滤根据近邻偏好产生推荐的特点, 恶意用户注入伪造用户模型成为正常用户近邻, 推进或打压目标项目的推荐排名, 从而达到改变推荐系统结果, 这种攻击方法称为"托攻击". 本文综述了托攻击模型与检测技术的研究现状和面临的主要问题, 试图为这一新兴的研究领域勾勒出较为全面清晰的概貌. 从推荐系统机理入手, 介绍托攻击产生动机、概念、目的、评分向量构成和模型分类, 然后提出衡量托攻击对推荐系统危害性的两类指标; 接着讨论区分正常用户和托攻击用户的特征指标; 然后以机器学习角度分类为主线, 综述 3 类托攻击检测算法, 分析 3 类算法的利与弊, 并介绍用于评估托攻击检测算法的数据集、指标和实验方法; 最后指出进一步的研究方向.

关键词
推荐系统
协同过滤
托攻击模型
托攻击检测算法

推荐系统是一种为用户提供建议的智能化软件工具, 目前已被应用于电子商务、电影和视频网站、音乐网络电台、社交网络、以及个性化阅读、邮件、广告等诸多领域[1]. 已有的推荐系统大多基于用户-项目矩阵进行推荐, 矩阵值 r_{mi} 表示第 U_m 个用户对第 I_i 个项目的评分值, 推荐系统的任务就是根据已知的用户-项目矩阵的部分值预测该矩阵的缺失值, 推荐系统选择 N 个预测值最高的项目作为用户的推荐列表.

协同过滤(Collaborative Filtering, CF)是最流行的推荐算法[2,3], 目前, 很多著名的推荐系统都是基于协同过滤的, 如亚马逊网络书店、GroupLens, TiVo, Netflix, YouTube 和 Facebook 等[4,5], CF 的基本思想是: 找到与目标用户 U_t 相似的 k 个用户, 根据 k 个相似用户对项目 I_i 的评分预测 U_t 对项目 I_i 的评分. 这一思想符合人们日常生活中的行为习惯, 即利用好朋友的喜好来推断某个陌生用户的喜好. 计算用户相似度仍然依据用户评分向量, 可以选择利用余弦相似度、皮尔逊相关系数、Jacaard 系数等[6]. 这种朴素的预测方法给了恶意用户可乘之机, 恶意用户如果能伪造出与目标用户 U_t 相似的评分向量, 就能影响 U_t 的预测评分, 恶意用户实施的这种攻击称为托攻击(Shilling Attack).

托攻击者通过伪造用户模型, 并使得伪造用户成为尽量多的正常用户的近邻, 由于协同过滤是基于近邻的兴趣来推荐的, 所以, 托攻击者就能干预系统的推荐结果, 增加或减少目标对象的推荐频率[7]. 电子商务的迅速发展, 使得网店店主和供货商利用托攻击攫取经济利益成为可能. 例如, 2001 年, 索尼影业承认利用伪造电影评论的方法向用户推荐许多新发行的电影; 2002 年, 亚马逊公司接到投诉后, 发现一起恶意用户利用托攻击使得网站在推荐一本基督教名著时还会推荐一本性方面的书籍[8].

由于推荐系统研究具有重要的理论意义和应用价值, 而托攻击检测是保障推荐系统安全性和健壮性的关键技术之一, SIGKDD, SIGIR, ICDM, AAAI,

引用格式: 伍之昂, 王有权, 曹杰. 推荐系统托攻击模型与检测技术. 科学通报, 2014, 59: 551–560
Wu Z A, Wang Y Q, Cao J. A survey on shilling attack models and detection techniques for recommender systems (in Chinese). Chin Sci Bull (Chin Ver), 2014, 59: 551–560, doi: 10.1360/972012-1712

WWW, RecSys, INFORMS *Journal on Computing* 等多个重要国际学术会议和国际期刊多次报道这方面的研究工作. 国际上很多著名大学和机构对该主题展开了深入研究, 如美国明尼苏达大学、德保尔大学和达特茅斯学院、爱尔兰都柏林大学、德国汉诺威大学、Google 公司等. 由于推荐系统是继搜索引擎之后兴起的研究方向, 目前, 国内外综述主要针对推荐系统本身[2,9,10], 只有少量涉及托攻击检测的综述文献[1,8,11]. 文献[1]的第 25 章聚焦于刻画托攻击对推荐系统造成的危害性, 未全面详细介绍托攻击检测方法. 文献[8]成文较早, 不能覆盖托攻击检测方法最新进展. 文献[11]虽然覆盖面极广, 但是对托攻击检测重要算法未能详细剖析, 且所涉及文献截止到 2011 年. 为此, 本文结合作者在托攻击检测领域的研究, 总结自 2004 年托攻击概念被提出到 2012 年为止, 托攻击模型与检测的主流研究和最新进展, 从机器学习角度将托攻击检测方法分为 3 类, 透彻剖析其中的重要成果, 并提出该领域的研究趋势, 期望能够让国内同行快速跟踪该领域的最新进展, 启迪未来的研究工作.

1 托攻击模型及其分类

从攻击者角度来看, 最好的托攻击对推荐系统造成的危害最大, 而实施托攻击的成本却降至最低[12], 成本包括伪造用户模型的数量和长度, 伪造用户模型所需知识, 如项目平均分、流行项目集合等. 图 1 描述了托攻击规模、威力和可检测性之间的关系[1], 高效托攻击以较小的规模对推荐系统造成极大的危害, 同时规模越大使得托攻击者越容易暴露, 而托攻击检测的研究目标正是试图扩大图 1 中的检测框. 从托攻击目的来看, 托攻击可以分为 3 类: 推攻击、核攻击和恶意扰乱攻击[12]. 推攻击试图提高目标项目的推荐排名, 反之, 核攻击试图降低目标项目的推荐排名. 而恶意扰乱攻击试图使推荐系统失灵.

为清晰描述托攻击模型, 托攻击评分向量通常被分为目标项目、装填项目、选择项目、及未评分项目, 如图 2 所示. 目标项目是攻击者试图提高或降低推荐频率的项目, 攻击者利用装填项目伪装成正常用户, 利用选择项目成为尽量多正常用户的近邻, 同时, 装填项目和选择项目增强了攻击威力.

Williams 等人[12,13]将托攻击构造模型分为 6 类, 如表 1 所示, 推攻击都对目标项目评最高分, 核攻击都对目标项目评最低分, 随机攻击对装填项目评分取随机值, 平均攻击对装填项目评分取该项目的平均值, 平均攻击的构造代价比随机攻击高. 分段攻击将目标项目的近邻项目作为选择项目, 以加强对同类型用户的影响程度. 流行攻击的基本思想是齐普夫定律, 即少数项目可以吸引大多数人的注意, 攻击者将流行项目作为选择项目, 项目的流行程度通常使用其被评分的次数来衡量. 抽样攻击也称为拷贝模型攻击, 需要其他用户的评分记录作为先验知识. Love/Hate 攻击[12]随机选择填充项目, 无需任何先验知识, 其核攻击版本是对基于用户的协同过滤推荐系统最有效的攻击手段.

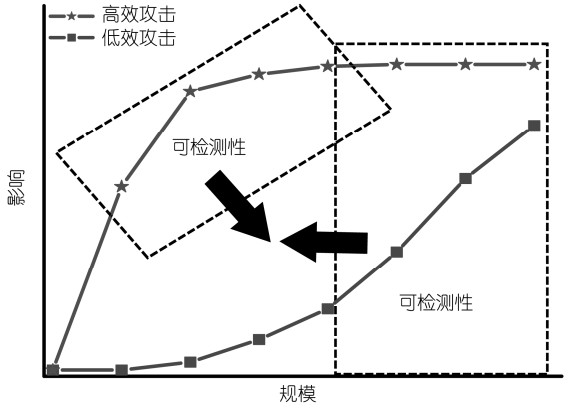

图 1 托攻击规模、威力和可检测性关系图[1]

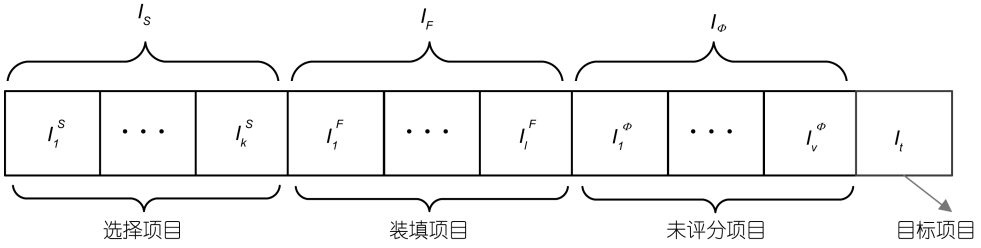

图 2 托攻击评分向量

表 1　托攻击模型

模型名称	推攻击	核攻击
随机攻击	$I_S=\varnothing$; $I_F=r_{ran}$; $I_t=r_{max}$	$I_S=\varnothing$; $I_F=r_{ran}$; $I_t=r_{min}$
平均攻击	$I_S=\varnothing$; $I_F=r_{avg}$; $I_t=r_{max}$	$I_S=\varnothing$; $I_F=r_{avg}$; $I_t=r_{min}$
分段攻击	$I_S=r_{max}$; $I_F=r_{min}$; $I_t=r_{max}$	$I_S=r_{min}$; $I_F=r_{max}$; $I_t=r_{min}$
流行攻击(随机装填)	$I_S=r_{max}$; $I_F=r_{ran}$; $I_t=r_{max}$	$I_S=r_{min}$; $I_F=r_{ran}$; $I_t=r_{min}$
流行攻击(平均装填)	$I_S=r_{max}$; $I_F=r_{avg}$; $I_t=r_{max}$	$I_S=r_{min}$; $I_F=r_{avg}$; $I_t=r_{min}$
抽样攻击	$I_S=\varnothing$; $I_F=r_{copy}$; $I_t=r_{max}$	$I_S=\varnothing$; $I_F=r_{copy}$; $I_t=r_{min}$
Love/Hate 攻击	$I_S=\varnothing$; $I_F=r_{min}$; $I_t=r_{max}$	$I_S=\varnothing$; $I_F=r_{max}$; $I_t=r_{min}$

为逃避检测, 恶意用户可能采用混淆技术, Williams 等人[12]提出两种混淆技术: 噪音注入和目标偏移. Hurley 等人[14]提出在最流行项目中选择装填项目构造平均攻击的混淆版本 AoP(Average-over-Popular)攻击, 这种技巧(不妨命名为“流行装填”)显然也适用于其他类型攻击, 我们将托攻击混淆技术归结为如下 3 种:

(1) 噪音注入: 在装填项目或选择项目的评分上加上一个随机数, William 等人[12]建议该随机数可以由常数因子 α 与高斯分布的随机数相乘得到.

(2) 目标偏移: 对目标项目评最高分或最低分容易引起检测器的注意, 目标偏移就是将目标项目评分改成次高分或次低分.

(3) 流行装填: 在 Top-x%的最流行项目集合内等概率选择装填项目, 项目流行程度可以通过项目被评分的数量来衡量.

由表 1 中单种模型构成的托攻击往往易于检测, 混合使用6种攻击模型和3种混淆技术可以构造出更为复杂的托攻击, 而且, 如果多个恶意用户同时攻击推荐系统, 他们自由选择攻击模型和混淆技术, 混合型托攻击也就应运而生了.

2　托攻击危害性衡量指标

托攻击危害性的衡量指标用于定量刻画描述一组托攻击对某种推荐系统的影响程度, 同时也反映出需要定义托攻击危害性的衡量指标, 这些指标可以用于比较不同托攻击模型的危害性, 也可以比较部署不同推荐算法的推荐系统面对托攻击时的脆弱性. 危害性与托攻击类型、装填项目数量、托攻击者数量、以及推荐系统核心算法有关. 衡量推荐系统性能可以从预测准确度和排序准确度入手[15], 预测准确度表示推荐系统预测评分与实际分值的接近程度, 排序准确度表示推荐系统所产生的 Top-N 推荐列表中有多少个项目是用户真正喜欢的. 我们将已有的托攻击危害性指标分为以下 2 方面.

(1) 预测准确度: 平均预测偏移刻画托攻击者对正常用户预测评分造成的影响, 设预测的评分数量为$|T|$, p'_{mi} 是存在托攻击时的预测值, p_{mi} 是正常情况下的预测值, 平均预测偏移计算如式(1)所示:

$$\overline{\Delta}=\sum_{r'_{mi}\in T}\frac{p'_{mi}-p_{mi}}{|T|}. \tag{1}$$

平均预测偏移越大, 说明推荐系统面对托攻击时越脆弱, 即托攻击对该推荐系统越有效.

(2) 排序准确度: 预测准确度通常并非推荐系统关注的重点, Top-N 推荐列表显得尤为重要. 因此, 托攻击者更为关注目标项目是否进入了正常用户的 Top-N 推荐列表. 击中率就从排序角度来衡量托攻击的危害性, 平均击中率的计算如式(2)所示:

$$\overline{HitRatio}=\frac{\sum_{U_m\in U}hits_{U_m}}{N\cdot|U|}, \tag{2}$$

其中, $hits_{Um}$ 表示目标项目在用户 U_m 的 Top-N 推荐列表中的个数.

Lam 和 Riedl[7]注意到多个预测分值相同的项目将占用 Top-N 列表中的一个排名, 提出预期 Top-N 占用数(Expected Top-N Occupancy, ExpTopN)指标, 计算公式如(3)所示:

$$ExpTopN=\sum_{I_t\in Targ}\frac{N_{I_t}}{RN_{I_t}}, \tag{3}$$

其中, $Targ$ 是目标项目集合, N_{I_t}是与目标项目 I_t有同样排名的项目个数, RN_{I_t}是 N_{I_t}个项目占用掉的排名个数. 大部分研究工作都是以平均预测偏移来衡量托攻击的危害性, Mehta 和 Nejdl[16]以最大化平均预测偏移为目标函数得出平均攻击在所有攻击模型中最具威力这一结论. 但是, $HitRatio$ 和 $ExpTopN$ 更加契合推荐系统本质和实施托攻击的目标.

3　托攻击的检测特征指标

攻击者与正常用户在评分方式上存在差异, 比如, 正常用户根据自己的喜好对商品差异化评分, 而托攻击总是对目标商品评最高分, 即便该商品未得到多数人的喜爱, 特征指标正是用于捕捉托攻击者与正常用户在评分方式上的差异.

表 2 列出文献[12,13,17,18]中提出的 11 个特征指标, 它们从不同角度刻画用户评分向量的差异性, 如评分向量的变化程度、用户与其近邻的相似度、评分项目与其平均分之间的关系等. 其中, 有 6 个指标用于刻画用户模型评过分项目与其平均值之间的关系, RDMA[17]是最早被定义的一阶矩指标, 它以 $1/|I_j|$规格化, 这使得$|I_j|$很小的项目对 RDMA 贡献过高, 不符合实际情况, 将规格化因子改成 $1/|I_j|^2$ 得到 WDMA. 进一步, Burke 等人[18]在 RDMA 的基础上提出了 WDA 和 FMD, 分别表示用户评分与其均值的偏差在列向量长度和行向量长度上的平均值. Williams 和 Mobasher[12]定义了 MeanVar 和 FMV 刻画二阶矩关系, MeanVar 和 FMV 区别在于: 前者去掉一个最高分项目, 后者将所有最高分项目都去除. 针对分段攻击, FMTD[12]指标用于刻画评最高分项目集合 $U_m{}^T$ 与剩余项目集合 $U_m{}^F$ 平均分之间的偏差, 而 TMF 则用于衡量用户模型对目标项目的关注度.

由于用户评分向量容易计算出各个特征指标值, 很多检测器将托攻击检测建模为分类问题, 综合多个特征指标作为属性训练分类器, 如图 3 将要介绍的基于 C4.5 决策树的检测器采用 5 个特征指标作为分裂属性.

4 托攻击检测算法

自从 2004 年托攻击概念被提出以来, 国内外学者提出了很多检测算法来加强推荐系统的健壮性和安全性. 托攻击检测本质上是一个分类问题, 从对先

表 2 托攻击的检测特征指标 a)

含义	计算公式
用户评分向量的熵	$Entropy=-\sum_{r=1}^{r_{\max}}\frac{n_r}{\sum_{i=1}^{r_{\max}}n_i}\log_2\frac{n_r}{\sum_{i=1}^{r_{\max}}n_i}$
用户模型与其 k 近邻的平均相似度	$DegSim=\frac{\sum_{n=1}^{k}Sim(U_m,U_n)}{k}$
用户评分向量的长度变化	$LengthVar=\frac{\lvert U_m\rvert-\lvert\overline{U}\rvert}{\sum_{n=1}^{\lvert U\rvert}(\lvert U_n\rvert-\lvert\overline{U}\rvert)^2}$
用户模型评过分项目与其平均值之间的一阶矩关系	$RDMA=\frac{1}{\lvert U_m\rvert}\sum_{j=1}^{\lvert U_m\rvert}\frac{\lvert r_{mj}-\overline{r_j}\rvert}{\lvert I_j\rvert}$ $WDMA=\frac{1}{\lvert U_m\rvert}\sum_{j=1}^{\lvert U_m\rvert}\frac{\lvert r_{mj}-\overline{r_j}\rvert}{\lvert I_j\rvert^2}$ $WDA=\sum_{j=1}^{\lvert U_m\rvert}\frac{\lvert r_{mj}-\overline{r_j}\rvert}{\lvert I_j\rvert}$
用户模型评过分项目与其平均值之间的二阶矩关系	$FMD=\frac{1}{\lvert U_m\rvert}\sum_{j=1}^{\lvert U_m\rvert}\lvert r_{mj}-\overline{r_j}\rvert$ $MeanVar=\frac{1}{\lvert U_m\rvert-1}\sum_{I_j\in(U_m-I_t)}(r_{mj}-\overline{r_j})^2$ $FMV=\frac{1}{\lvert U_m^F\rvert}\sum_{I_j\in U_m^F}(r_{mj}-\overline{r_j})^2$
评最高分项目集合与剩余项目集合的平均分偏差	$FMTD=\left\lvert(\frac{\sum_{i\in U_m{}^T}r_{mi}}{\lvert U_m^T\rvert})-(\frac{\sum_{j\in U_m{}^F}r_{mj}}{\lvert U_m^F\rvert})\right\rvert$
用户模型对目标项目的关注度	$TMF=\max_{I_j\in U_m^T}\frac{\lvert I_j^{\max}\rvert}{\sum_{U_n\in I_j^{\max}}\lvert U_n^T\rvert}$

a) r 表示用户的评分值. n_r表示评分值 r 在 U_m中出现的次数. $|U_m|$表示 U_m的长度. $Sim(U_m, U_n)$表示两个用户之间的相似度, 可以选用皮尔逊相关系数、余弦相似度或 Jaccard 相关系数等. $\overline{r_j}$ 表示项目 I_j的平均评分. $|I_j|$表示对项目 I_j评分的用户数量. $|\overline{U}|$表示所有用户的平均评分长度. $U_m{}^T$表示用户 U_m评最高分的项目集合. $U_m{}^F$表示用户 U_m的其他评分项目集合. $I_j{}^{\max}$表示对项目 I_j评最高分的用户集合

验知识的使用程度, 检测算法可分为基于监督学习、无监督学习和半监督学习3类. 从算法使用用户模型信息来看, 检测算法可分为两类: 第一类算法依据第3节所介绍的特征指标来检测; 第二类算法直接依据用户评分记录来检测. 我们以机器学习角度分类为主线, 融合所依据的用户模型信息类型来介绍现有的托攻击检测算法.

4.1 监督学习

以已知类别用户作为参照来训练检测器是人们应对托攻击检测时的直观想法, 其本质是基于监督学习构造分类器, 该类检测算法大多将第3节定义的特征指标作为分类器属性. Chirita等人[17]提出的第一个托攻击检测算法通过观察根据标记用户的各项特征指标分布规律, 首次提出结合DegSim和RDMA的托攻击检测经验算法使用平均相似度和RDMA两个指标来检测平均攻击和随机攻击. 随后, 美国DePaul大学Williams等人[12,18,19]系统定义了检测指标, 在基于决策树检测托攻击方面做了大量工作, Williams的技术报告总结了他们所做的工作, 图3描述了用于检测平均、随机、分段、流行和Love/Hate5种托攻击的C4.5决策树. 显然, 特征指标的选择是影响监督学习检测器性能的重要因素, 人为地预设一些特征指标难以适应随时变化的攻击者, 为此, 文献[20]提出了一种基于标记用户模型的特征指标选择算法, 根据训练集自动选择区分能力好的指标. 特征选择方法无疑能作为一种补充手段提升监督学习检测器性能, 其本身一直是数据挖掘中的研究议题, 适用于托攻击检测应用的特征选择方法却鲜有论述.

4.2 无监督学习

由于监督学习的检测器过度地依赖于特征指标和训练集, 对于跟训练集特征相似的用户模型检测效果良好, 而对于新的托攻击或经过混淆后托攻击则力所不逮. 因此, 研究者转向使用无监督学习构造检测器, 即试图将托攻击者"聚"到一起加以识别.

Mehta 等人[16,21]发现托攻击者之间的皮尔逊相似度极高(>0.9), 因此, 相似度最高的一些用户很可能就是托攻击者. 据此, Mehta 等人提出第一个基于无监督学习的检测器PCASelectUsers, 无需任何先验知识, 而且不依赖于特征指标, 该算法首先将用户-项目评分矩阵转化为z-score, 然后将 D 的转置矩阵与 D 相乘得到协方差矩阵, 再利用主元分析获得3~5个Eigen向量以计算距离, 返回 r 个距离最小的用户作为托攻击者, PCASelectUsers算法流程如图4所示. PCASelectUsers极具巧思, 在没有任何先验知识的指导下取得了不错的效果, 但是, 它难以对付 AoP 攻击[14], 更为致命的是, 人们难以知道实际的推荐系统中隐藏了多少个托攻击者, 所以, 很难设定参数 r, 这极大限制了 PCASelectUsers 的实际应用.

Bryan 等人[22]提出 UnRAP 通过 3 个步骤检测托

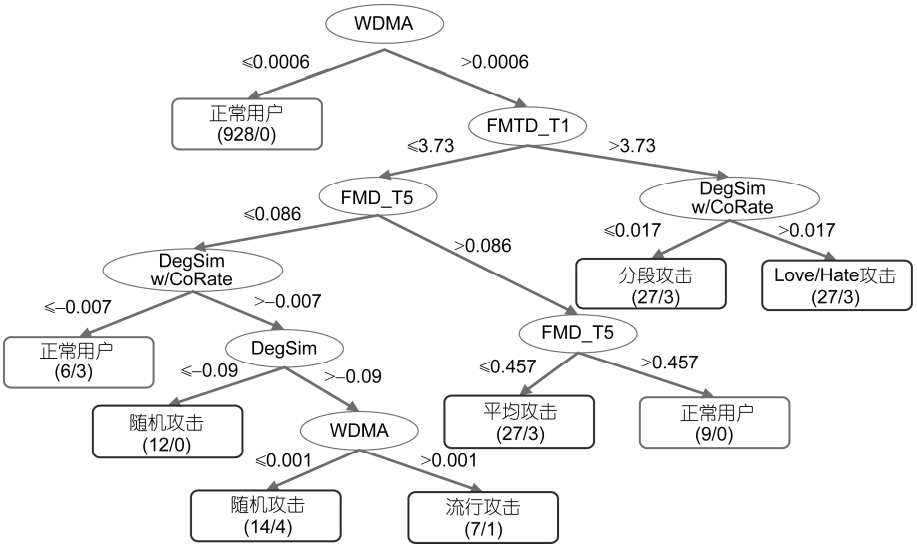

图 3 基于 C4.5 决策树的托攻击检测算法[12]

攻击: (1) 计算所有用户的 H_v 分值; (2) 确定目标项目; (3) 确定托攻击者. Hurley 等人[14]提出了基于奈曼-皮尔逊准则的托攻击检测器, 分别提供监督学习和无监督学习两个版本. Lee 和 Zhu[23]提出的检测器首先利用聚类的方法将相似用户聚成同一簇, 然后提出 Group RDMA 来判断某一簇的用户是否为托. 文献[24,25]利用奇异值分解的无监督检测方法. 本质上, 上述这些基于无监督学习的检测器潜在假设了托攻击者具有极大的相似性, 检测器的准确性也依赖于这一规律是否应验.

4.3 半监督学习

珍贵的标记用户信息若弃之不用殊为可惜, 大量无标记用户表现出的分布规律却也不容忽视, 研制基于半监督学习的托攻击检测器便呼之欲出了. 在亚马逊、淘宝等实际的电子商务网站中, 存在大量无法确定身份的用户(即无标记数据), 而只有少量用户的身份可以确定(即标记数据), 例如, 淘宝网上好评率极高或极低的用户、皇冠用户等的身份容易确定, 大量好评率适中用户的身份难以确定. 同时, 无标记数据往往容易获取, 获取标记数据可能耗费大量的人力物力, 例如, 逐一辨明淘宝网海量用户的身份是非常困难的, 所以半监督学习[26]的托攻击检测方法适应实际需求.

Wu 等人[27~29]提出一种基于半监督学习的托攻击检测方法, 首先使用朴素贝叶斯分类器在标记数据上训练初始分类器, 然后在无标记数据上改进分类器. 以往的检测器都聚焦于由单种模型构造出的托攻击, 事实上, 不同的恶意用户可能使用不同模型和混淆技术实施托攻击, 推荐系统面临着混合型托攻击的威胁. 为此, 我们提出一种针对混合型托攻击的基于半监督学习检测器, 称为 HySAD, 图 5 描述了 HySAD 的总体框架, HySAD 是基于特征指标的, 集成了特征自动选择功能, 核心部分的学习过程基于半监督朴素贝叶斯(SNB_λ)方法展开, SNB_λ利用极大似然估计参数值, 使用类似 EM 算法迭代求解[29].

基于半监督学习的检测器综合了标记数据的准确性, 又合理使用了无标记数据的分布规律, 比以往监督学习和无监督学习的检测器的性能更为优越.

机器学习角度对现有托攻击检测算法的分类说明了算法使用先验知识的方式, 从先验知识内容来看, 现有检测算法或将第 3 节所介绍的特征指标作为输入数据, 如 Chirita 等人[17]最早提出的算法、基于 C4.5 决策树的算法[12]、及半监督的 HySAD[29]等; 抑或直接利用评分向量作为输入数据, 如: 基于奈曼-皮尔逊准则的检测器[14]、PCASelectUsers[16]及 UnRAP[22]; 只有 Lee 和 Zhu[23]提出的基于聚类和 GRDMA 的检测算法综合利用了特征指标和评分向量.

5 托攻击检测实验数据集与评价指标

为评价新提出托攻击检测算法的有效性, 需要通过在一些开放数据集上实验获得一系列评价指标. 仿真实验是将模拟产生各种类型托攻击注入开放数据集中, 以评价算法的检测效果. 而真实案例分析试

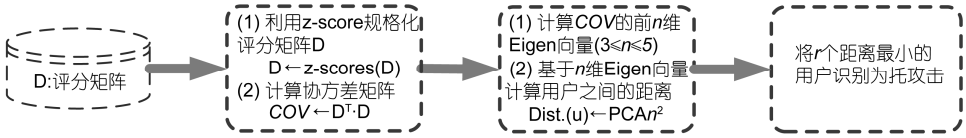

图 4 PCASelectUsers 算法总体流程

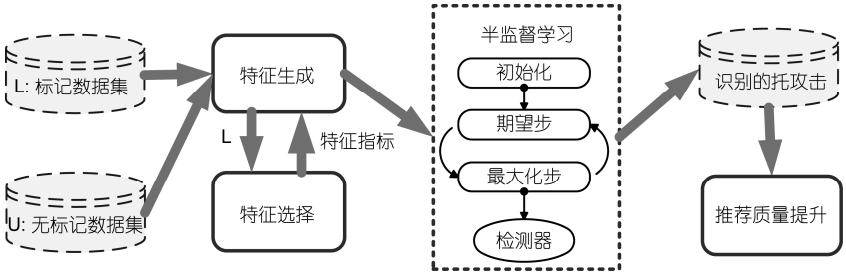

图 5 HySAD 总体框架[29]

图检测实际网站中存在的托攻击, 并为检测结果寻求语义支撑. 本节从仿真实验和真实案例分析两个角度介绍托攻击检测实验中的常用数据集、评价指标、实验方法等内容, 为其他研究者进一步实施实验评价方面提供基础知识.

5.1 仿真实验

绝大部分托攻击模型与检测领域的研究[7,12,14,16~20]都将美国明尼苏达大学 GroupLens 研究小组发布的 MovieLens (http://movielens.umn.edu/)作为实验数据集, MovieLens 包含 100 K, 1 和 10 M 3 个数量级的评分个数, 在托攻击模型与检测领域应用最广泛的是 ML-100 K 数据集, 它包含 5 个训练集和测试集分组, 每一个分组是 943 个用户在 1682 部电影上 100000 条评分的不同比例拆分. 少量研究采用了 EachMovie 数据集[30], 原因在于 MovieLens 部分数据来自于 EachMovie. Netflix (http://www.netflixprize.com/)是另一个著名电影评分数据集, Netflix 数据集规模极大, 很难在全部数据集上做分析, 文献[29,31]在 Netflix 部分数据集上验证了检测算法性能.

评价托攻击检测方法性能需要同时观察准确率和召回率, 而 F-measure 指标综合了准确率和召回率两个指标, 能刻画检测算法的整体性能. 与经典分类问题一样, 我们首先得到真正(*TP*)、假负(*FN*)、假正(*FP*)和真负(*TN*)4 个分量, 然后由式(4)可计算出准确率(*P*)、召回率(*R*)和 F-measure(*F*).

$$P=\frac{TP}{TP+FP},\ R=\frac{TP}{TP+FN},\ F=\frac{2PR}{P+R}. \tag{4}$$

图 6 描述了托攻击检测仿真实验流程, 托攻击生成器根据表 1 所示的托攻击模型定义, 产生托攻击评分向量, 并注入开放数据集, 形成合成数据集, 其中开放数据集的原有用户被标记正常用户、新注入的用户被标记为托攻击者. 图中托攻击检测器可部署各类检测器, 如 C4.5 决策树、PCASelectUsers, HySAD 等, 基于特征指标的检测器根据用户评分向量和表 2 计算公式得到各种指标值, 检测器的输出是合成数据集内所有用户的身份标识(正常用户或托攻击者). 合成数据集标记已知, 因此, 计算准确率、召回率和 F-measure 等指标就可以定量评估检测算法.

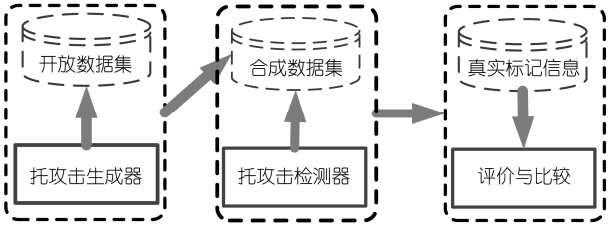

图 6 托攻击检测仿真实验流程

5.2 真实案例分析

目前尚未有精确标记正常用户和托攻击的真实数据集公布, 因而才需要仿真产生托攻击用户模型, 显然, 合成数据集的真实标记信息正确的前提条件是开放数据中用户都为正常用户, 而事实上, 这个前提条件未必正确, 即我们无法确定开放数据集原有用户是否包含托攻击、其比例有多大. 据此, 真实案例分析试图识别数据集中的托攻击者, 利用语义来支撑识别结果, 而语义信息的获取视不同应用和数据而异, 并无固定方法.

文献[29]在 Amazon.cn 评论数据集上做了真实案例分析. Amazon.cn 数据集包含 49289 个用户在 504170 件商品上的 2347178 个评论记录, 每条记录包含评论 ID、用户 ID、商品 ID、评论题目和内容、5 分制评分值、创建和更新时间、及是否购买标记. 首先, 以评分值为输入数据, 使用半监督托攻击检测算法 HySAD 识别出概率最高的托攻击用户集合; 其次, 通过评论内容、创建和更新时间、购买行为、Amazon.cn 排名等几个方面来分析这些值得怀疑用户的行为, 从而为检测算法结果找到语义支撑.

真实案例分析面向实际系统, 相比于仿真实验, 语义支撑在验证托攻击检测方法上更具说服力. 但是, 语义支撑依据往往通过手工获取, 耗时耗力, 难以排查实际系统中的所有用户. 因此, 结合仿真实验和真实案例分析两种方法, 从基准测试数据和语义信息两方面来评价新提出的托攻击检测算法是一个非常合理可信的手段.

6 未来的研究方向

托攻击模型与检测具有丰富的应用前景, 尽管在该领域已经投入了大量的研究工作, 并取得了诸多令人鼓舞的研究成果, 表现为攻击模型、检测器、健壮性算法 3 方面, 图 7 总结了现有工作及其随时间的演变. 然而, 我们认为该领域内尚有以下几个方向值得进一步探索.

(1) 群组托攻击检测: 一组托攻击者协同工作以

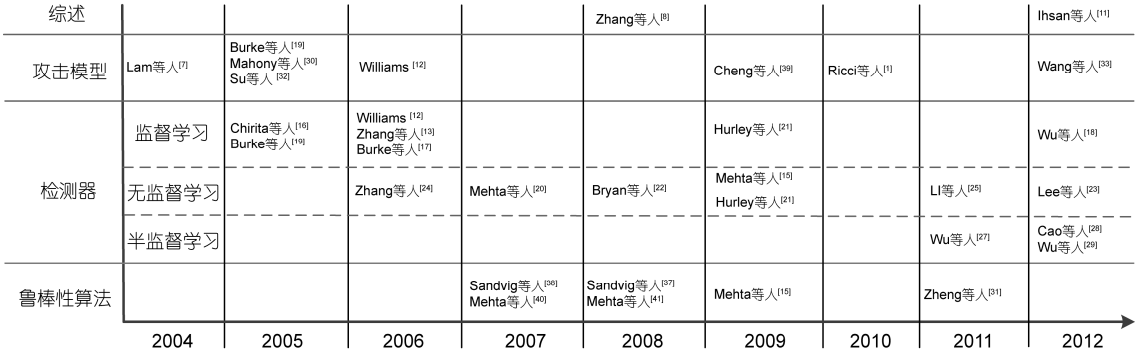

图 7 托攻击模型与检测领域现有研究分类及其时间线

增加或降低目标项目的推荐排名, 群组托攻击内的单个攻击模型看起来更接近于正常用户, 已有的检测器难以逐个发现群组托攻击, 尽管很多学者[32,33]对群组托攻击方式进行了勾勒, 但是, 目前已有的工作仅包括文献[32]对群组托攻击检测的简单讨论, 及文献[33]探索了群组托攻击模型构造方法, 却尚未有系统性的群组托攻击构造方式和检测手段研究成果出现.

(2) 基于商品评论的托攻击检测: 已有的研究都是假设托攻击通过伪造用户对项目的评分来进行的, 在实际电子商务系统中, 用户对商品的评论将直接影响其他用户的购买行为, 因此, 恶意用户有基于商品评论展开托攻击的可能. 尽管恶意评论检测、甚至群组恶意评论检测得到了研究者的重视[34~36], 尚未有研究检测隐藏在商品评论中的托攻击.

(3) 非协同过滤式推荐系统的攻击与防范: 目前的托攻击都是针对采用协同过滤的推荐系统展开的, 协同过滤诚然是应用最广泛、且最简洁有效的推荐算法, 但是还存在很多其他推荐算法, 如基于 Kmeans 聚类、概率潜在语义分析、频繁模式挖掘等方法的推荐算法[37,38]. 都柏林大学的 Cheng 和 Hurley[39]提出了针对非协同过滤式推荐系统的托攻击模型, 具有极高的击中率.

(4) 抗托攻击的推荐算法设计: 攻守相辅相成, 检测已存在的托攻击固然是保障推荐系统安全性的一个方法, 设计难以为托攻击所乘的高健壮性推荐算法成为研究者的自然想法. 很多学者[16,40,41]在提出托攻击检测方法的同时, 也提出抗托攻击的推荐算法. 但是, 设计具有较高的预测准确度的抗托攻击推荐算法极具挑战性, 还需要付出进一步的努力.

7 结束语

随着推荐系统重要性的日益显现, 推荐系统的健壮性和安全性问题得到越来越多的关注, 推荐系统托攻击检测问题也在近几年受到工业界和学术界的重视, 结合数据挖掘和机器学习托攻击检测方法的研究羽翼渐丰. 本文主要从以下 5 个方面对现有工作做了回顾: (1) 介绍托攻击概念、目的、评分向量构成和模型分类; (2) 介绍衡量托攻击对推荐系统危害性的两类指标; (3) 总结区分正常用户和托攻击用户的特征指标; (4) 以机器学习角度分类为主线, 介绍 3 类托攻击检测算法; (5) 总结用于评估托攻击检测算法的数据集、指标和实验方法. 最后, 本文提炼出托攻击模型与检测领域中值得进一步探索的方向, 以期启发同行们的研究工作.

参考文献

1 Ricci F, Shapira B. Recommender Systems Handbook. Berlin: Springer, 2011
2 许海玲, 吴潇, 李晓东, 等. 互联网推荐系统比较研究. 软件学报, 2009, 20: 350–362
3 Bell R M, Koren Y. Improved neighborhood-based collaborative filtering. In: Berkhin P, Caruana R, Wu X D, eds. Proceedings of the 13th International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2007. 7–14
4 Linden G, Smith B, York J. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Comput, 2003, 7: 76–80
5 Fang L, Kim H, LeFevre K, et al. A privacy recommendation wizard for users of social networking sites. In: Chen Y, Danezis G, Shmatikov V, eds. Proceedings of the 17th International Conference on Computer and Communications Security. New York: ACM, 2010. 630–632

6 Herlocker J L, Konstan J A, Terveen L G, et al. Evaluating collaborative filtering recommender systems. ACM Trans Inform Syst, 2004, 22: 5–53

7 Lam S K, Riedl J. Shilling recommender systems for fun and profit. In: Feldman S I, Uretsky M, Najork M, et al, eds. Proceedings of the 13th International Conference on World Wide Web. New York: ACM, 2004. 393–402

8 张富国, 徐升华. 推荐系统安全问题及技术研究综述. 计算机应用研究, 2008, 25: 656–659

9 王立才, 孟祥武, 张玉洁. 上下文感知推荐系统. 软件学报, 2012, 23: 1–20

10 Cacheda F, Carneiro V, Fernández D, et al. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. ACM Trans Web, 2011, 5: 2

11 Gunes I, Kaleli C, Bilge A, et al. Shilling attacks against recommender systems: A comprehensive survey. Artif Intell Rev, 2012, doi: 10.1007/s10462-012-9364-9

12 Williams C, Mobasher B. Profile injection attack detection for securing collaborative recommender systems. Technical Report, Computer Science, DePaul University. 2006

13 Zhang S, Chakrabarti A, Ford J, et al. Attack detection in time series for recommender systems. In: Eliassi-Rad T H, Ungar L, Craven M, et al, eds. Proceedings of the 12th International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2006. 809–814

14 Hurley N J, Cheng Z P, Zhang M. Statistical attack detection. In: Bergman L D, Tuzhilin A, Burke R, et al, eds. Proceedings of the 3rd International Conference on Recommender Systems. New York: ACM, 2009. 149–156

15 Liu Q, Chen E H, Xiong H, et al. Enhancing collaborative filtering by user interest expansion via personalized ranking systems. IEEE Trans Syst Man Cy B, 2012, 42: 218–233

16 Mehta B, Nejdl W. Unsupervised strategies for shilling detection and robust collaborative filtering. User Model User-Adap, 2009, 19: 65–97

17 Chirita P, Nejdl W, Zamfir C. Preventing shilling attacks in online recommender systems. In: Bonifati A, Lee D, eds. Proceedings of the 7th International Workshop on Web Information and Data Management. New York: ACM, 2005. 67–74

18 Burke R, Mobasher B, Williams C, et al. Classification features for attack detection in collaborative recommendation systems. In: Bonifati A, Fundulaki I, eds. Proceedings of the 12th International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2006. 542–547

19 Burke R, Williams C, Bhaumik R. Segment-based injection attacks against collaborative filtering recommender systems. In: Han J W, Wah B, eds. Proceedings of the 5th International Conference on Data Mining. New York: IEEE, 2005. 577–580

20 伍之昂, 庄毅, 王有权, 等. 基于特征选择的推荐系统托攻击检测算法. 电子学报, 2012, 40: 1687–1693

21 Mehta B, Hofmann T, Fankhauser P. Lies and propaganda: detecting spam users in collaborative filtering. In: Chin D N, Zhou M X, Lau T A, et al, eds. Proceedings of the 12th International Conference on Intelligent User Interfaces. New York: ACM, 2007. 14–21

22 Bryan K, O'Mahony M P, Cunningham P. Unsupervised retrieval of attack profiles in collaborative recommender systems. Technical Report, University College Dublin, 2008

23 Lee J, Zhu D. Shilling attack detection—A new approach for a trustworthy recommender system. Informs J Comput, 2012, 24: 117–131

24 Zhang S, Ouyang Y, Ford J, et al. Analysis of a low-dimensional linear model under recommendation attacks. In: Efthimiadis E N, Dumais S T, Hawking D, et al, eds. Proceedings of the 29th International Conference on Research and Development in Information Retrieval. New York: ACM, 2006. 517–524

25 李聪, 骆志刚, 石金龙. 一种探测推荐系统托攻击的无监督算法. 自动化学报, 2011, 37: 160–167

26 Zhou Z H, Li M. Tri-Training: Exploiting unlabeled data using three classifiers. IEEE Trans Knowl Data En, 2005, 17: 1529–1541

27 Wu Z A, Cao J, Mao B, et al. Semi-SAD: Applying semi-supervised learning to shilling attack detection. In: Mobasher B, Burke R, Jannach D, et al, eds. Proceedings of the 5th International Conference on Recommender Systems. New York: ACM, 2011. 289–292

28 Cao J, Wu Z A, Mao B, et al. Shilling attack detection utilizing semi-supervised learning method for collaborative recommender system. World Wide Web, 2012, doi: 10.1007/s11280-012-0164-6

29 Wu Z A, Wu J J, Cao J, et al. HySAD: A semi-supervised hybrid shilling attack detector for trustworthy product recommendation. In: Yang Q, Agarwal D, Pei J, et al, eds. Proceedings of the 18th International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2012. 985–993

30 O'Mahony M P, Hurley N J, Silvestre G C M. Recommender systems: Attack types and strategies. In: Anderson M, Oates T, eds. Proceedings of the 20th National Conference on Artificial Intelligence. USA: MIT Press, 2005. 334–339

31 Zheng S, Tao J, Baras J S. A robust collaborative filtering algorithm using ordered logistic regression. In: Hagimoto K, Ueda H, Jamallipour A, eds. Proceedings of the 17th International Conference on Communications. New York: IEEE, 2011. 1–6

32 Su X F, Zeng H J, Chen Z. Finding group shilling in recommendation system. In: Ellis A, Hagino T, eds. Proceedings of the 14th International Conference on World Wide Web. New York: ACM, 2005. 960–961

33 Wang Y Q, Wu Z A, Cao J, et al. Towards a tricky group shilling attack model against recommender systems. In: Zhou S G, Karypis G, Zhang S M, eds. Proceedings of the 8th International Conference on Advanced Data Mining and Applications. Berlin: Springer, 2012. 675–688
34 Mukherjee A, Liu B, Glance N S. Spotting fake reviewer groups in consumer reviews. In: Mille A, Gandon F L, Misselis J, et al, eds. Proceedings of the 21st International Conference on World Wide Web. New York: ACM, 2012. 191–200
35 Lim E, Nguyen V, Jindal N, et al. Detecting product review spammers using rating behaviors. In: Huang J, Koudas N, Jones G J F, et al, eds. Proceedings of the 19th International Conference on Information and Knowledge Management. New York: ACM, 2010. 939–948
36 Wang G, Xie S H, Liu B, et al. Review graph based online store review spammer detection. In: Cook D J, Pei J, Wang W, et al, eds. Proceedings of the 11th International Conference on Data Mining. New York: IEEE, 2011. 1242–1247
37 Sandvig J J, Mobasher B, Burke R. A survey of collaborative recommendation and the robustness of model-based algorithms. IEEE Data En Bull, 2008, 31: 3–13
38 Sandvig J J, Mobasher B, Burke R. Robustness of collaborative recommendation based on association rule mining. In: Konstan J A, Riedl J, Smyth B, eds. Proceedings of the 1st International Conference on Recommender Systems. New York: ACM, 2007. 105–112
39 Cheng Z P, Hurley N. Effective diverse and obfuscated attacks on model-based recommender systems. In: Bergman L D, Tuzhilin A, Burke R, et al, eds. Proceedings of the 3rd International Conference on Recommender Systems. New York: ACM, 2009. 141–148
40 Mehta B, Hofmann T, Nejdl W. Robust collaborative filtering. In: Konstan J A, Riedl J, Smyth B, eds. Proceedings of the 1st International Conference on Recommender Systems. New York: ACM, 2007, 49–56
41 Mehta B, Nejdl W. Attack resistant collaborative filtering. In: Myaeng S, Oard W D, Sebastiani F, et al, eds. Proceedings of the 31st International Conference on Research and Development in Information Retrieval. New York: ACM, 2008. 75–82

A survey on shilling attack models and detection techniques for recommender systems

WU ZhiAng[1], WANG YouQuan[2] & CAO Jie[1,2]

[1] *Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing 210003, China;*
[2] *College of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China*

Since collaborative filtering generates personalized recommendations according to the preference of nearest neighbors, malicious users can fake profiles to be nearest neighbors of normal users, in order to push or suppress the recommendation rank of the target item and thus change the output of recommender systems. Such attack is termed “shilling attack”. This paper reviews the states of art and the main problems of existing works related to shilling attack models and detection techniques, and attempts to sketch a comprehensive and explicit outline for this new and active research realm. In particular, the motivations, concepts, intent, ingredients and classifications of the shilling profiles are introduced, and two kinds of metrics for evaluating the harmness of shilling attacks are presented. A set of metrics for characterizing the normal user and the shilling attacker are discussed. Moreover, the instant shilling attack detection algorithms can fall into three categories from the machine learning aspect, and then data sets, evaluation measures, as well as experimental methods for evaluating these algorithms are addressed. Finally, a wealth of research directions that are worth for further exploration are marked out.

recommender systems, collaborative filtering, shilling attack model, shilling attack detection algorithm

doi: 10.1360/972012-1712