基于用户声誉的鲁棒协同推荐算法

张燕平 [1,2] 张 顺 [1,2] 钱付兰 [1,2] 张以文 [1,2]

摘 要 随着推荐系统在电子商务界的快速发展以及取得的巨大经济收益, 有目的性的托攻击是目前协同过滤系统面临的重大安全威胁, 研究一种可抵御攻击的鲁棒推荐技术已成为目前推荐系统领域的重要课题. 本文利用历史记录得到用户声誉, 建立声誉推荐系统, 并结合协同过滤推荐领域内的隐语义模型, 提出基于用户声誉的隐语义模型鲁棒协同算法. 本文提出的算法从人为攻击和自然噪声两个方面对系统的鲁棒性进行了改善. 在真实的数据集 Movielens 1M 上的实验表明, 与现有的鲁棒性推荐算法相比, 这种算法具有形式简单、可解释性强、稳定的特点, 且在精度得到一定提升的情况下大大增强了系统抵御攻击的能力.

关键词 推荐系统, 协同过滤, 声誉, 托攻击

引用格式 张燕平, 张顺, 钱付兰, 张以文. 基于用户声誉的鲁棒协同推荐算法. 自动化学报, 2015, **41**(5): 1004−1012

DOI 10.16383/j.aas.2015.c140073

Robust Collaborative Recommendation Algorithm Based on User′s Reputation

ZHANG Yan-Ping[1,2] ZHANG Shun[1,2] QIAN Fu-Lan[1,2] ZHANG Yi-Wen[1,2]

Abstract With the rapid development of recommender systems in e-commerce industry, such systems bring huge economic profits. As a consequence, shilling attacks pose a significant threat to the security of collaborative filtering recommender systems. Developing a kind of robust recommendation technology which can resist attacks has become an important issue in the field of the recommender system at present. In this paper, a reputation recommender system is built by user reputations which are obtained from the user historical records. Utilizing the latent factor model in the field of collaborative filtering recommendation, a novel robust collaborative recommendation algorithm based on user reputations is proposed. The algorithm improves the system′s robustness from two aspects of shilling attack and natural noise. Empirical results on Movielens 1M dataset demonstrate that compared with the existing robust recommendation, this algorithm is very effective. Characterized by simplicity, interpretability and stability, the algorithm has strong ability to resist the system attack along with the accuracy getting a certain improvement.

Key words Recommender system, collaborative filtering, reputation, shilling attack

Citation Zhang Yan-Ping, Zhang Shun, Qian Fu-Lan, Zhang Yi-Wen. Robust collaborative recommendation algorithm based on user′s reputation. *Acta Automatica Sinica*, 2015, **41**(5): 1004−1012

电子商务的飞速发展将人类带入网络经济时代, 面对大量的商品信息, 用户往往难以发现最需要或最合适的商品. 用户希望电子商务系统具有一种类似采购助手的功能来帮助其选购商品, 它能自动地把用户最感兴趣的商品推荐出来. 推荐系统正是针对以上问题和需求产生的, 联系用户和信息, 一方面帮助用户发现自己感兴趣的商品; 另一方面还能让商品展现在对它感兴趣的人群中, 从而实现供应商与用户的双赢. 自从 20 世纪 90 年代推荐系统被作为一个独立的概念提出来[1−2], 研究者们一直致力于研究出一种既快速又准确的推荐算法, 而往往忽视了推荐系统鲁棒性的研究. 推荐系统中准确性衡量预测评分与真实评分之间的吻合程度, 鲁棒性衡量有外部干扰时预测的变动程度, 对推荐系统来说也非常重要. 近年来推荐系统的鲁棒性的重要性日益凸显. 攻击者 (如 “水军” 等) 有目的地向推荐系统中注入虚假评分, 从而操纵推荐结果, 对竞争对手进行打击或牟取不当利益等的行为对推荐系统的鲁棒性造成了威胁, 建立有效防止攻击的推荐系统在实际中有重要的意义. 因此, 本文利用用户历史记录得到用户声誉, 并结合协同过滤推荐领域内的隐语

收稿日期 2014-01-28 录用日期 2014-12-03
Manuscript received January 28, 2014; accepted December 3, 2014
国家自然科学基金 (61175046), 安徽大学青年科学基金 (KJQN1116), 安徽省自然科学基金项目 (1408085MF132), 教育部人文社科青年基金 (14YJC860020) 资助
Supported by National Natural Science Foundation of China (61175046), Youth Science Fund of Anhui University (KJQN1116), Natural Science Found of Anhui Province (1408085MF132), and Humanities and Social Science Youth Fund of Ministry of Education (14YJC860020)
本文责任编委 赵铁军
Recommended by Associate Editor ZHAO Tie-Jun
1. 安徽大学计算机科学与技术学院 合肥 230601 2. 安徽大学智能计算与信号处理教育部重点实验室 合肥 230601
1. School of Computer Science and Technology, Anhui University, Hefei 230601 2. Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei 230601

义模型, 提出基于用户声誉的鲁棒协同过滤算法.

本文主要章节组织如下. 第 1 节介绍推荐系统中存在的托攻击以及相关工作; 第 2 节回顾协同过滤推荐算法中的隐语义模型, 并结合用户声誉推荐系统提出基于声誉的鲁棒协同推荐算法; 第 3 节为实验结果与分析; 最后, 第 4 节为全文的总结和未来工作的展望.

1 相关工作

1.1 托攻击

推荐系统外部干扰既有自然噪声的也有人为噪声的. 自然的干扰, 如推荐系统中用户在给项目打分时, 有些用户比较严谨, 有些用户则不严谨, 不严谨用户的评分就会成为自然的评分噪声. 人为的干扰主要是指有意识的恶意行为 — 托攻击 (Shilling attacks)[3]. 恶意用户通过向推荐系统中注入虚假的用户概要文件 (伪造目标项目和装填项目的评分) 来尽量模拟其他真实用户的兴趣爱好, 对推荐结构产生影响. 托攻击按照对目标项的攻击可以分为两类[4], 提高目标项的评价, 称为推攻击 (Push attack), 如抬高自己商品评分的攻击; 降低目标项的评价, 称为核攻击 (Nuke attack), 如恶意降低竞争对手商品评分的攻击. 根据装填项目填充的评分的不同, 一般将托攻击分为五类[5−6], 分别为抽样攻击、均值攻击、随机攻击、流行攻击以及分段攻击. 一般主要采用随机攻击、均值攻击和流行攻击三种攻击方式去评估算法的效果. 随机攻击[7] 的装填项目评分以所有评分的均值为中心, 在一个很小的范围内随机选取, 目标项打最高分 (推攻击) 或者最低分 (核攻击); 均值攻击[7] 的装填项目评分以每一个项目的评分均值为中心, 在一个很小的范围内随机产生, 目标项打最高分或者最低分. 而流行攻击[8] 则是将系统中受欢迎的项目作为装填项目打最高分, 目标项打最高分或者最低分. 其中均值攻击需要了解每一个项目的评分平均值, 知识成本较高, 实际情况中实现难度较大, 随机攻击知识成本较低, 较易实现. 但实验表明[9], 均值攻击比随机攻击的效率更高, 且均值攻击采用项目评分的平均值来填充, 在检测过程中不易被发现. 因此本文为了获取更好的攻击效果, 采用的攻击方式为均值攻击.

1.2 鲁棒推荐技术的相关研究

现有的推荐系统主要运用两种托攻击防御策略[10]: 1) 在推荐之前检测并删除攻击; 2) 依靠推荐算法内在的健壮性平滑托攻击的不良影响. 最早的鲁棒协同过滤算法[11] 利用概貌效用 (Profile utility) 改进了基于 Pearson 相关性 K-最近邻选取策略, 阻止攻击者进入目标用户的最近邻. 然而这种算法的鲁棒性并不理想. 文献 [12] 中提出鲁棒矩阵分解 (Robust matrix factorization, RMF) 算法, 利用 M-估计子 (Mestimator)[13] 对离群点的容忍性, 一定程度抑制了评分噪声的影响, 并没有实质地提升推荐系统的鲁棒性, 但相比于之前的鲁棒协同算法在推荐精度上有所提升. 上述两种算法都属于策略 1). 文献 [9] 提出基于主成分分析 (Principal component analysis, PCA) 方法的变量 – 选择奇异值分解 (Singular value decomposition using variable selection, VarSelect SVD) 算法探测并标识可疑用户, 限制其对推荐模型构建过程的干扰, 属于策略 2), 其鲁棒性显著优于 RMF 算法, 但随着攻击用户的填充项目的比率增加, 其识别可疑用户的正确率明显下降, 鲁棒性得不到很好的保证.

现有的算法只考虑人为噪声 (攻击) 所带来的影响, 并未考虑自然噪声. 推荐系统中不仅存在恶意攻击问题对鲁棒性的影响, 还存在一些非恶意的用户行为导致推荐结果不准确, 如用户在给项目打分时, 有些用户比较严谨, 有些用户则不严谨或者说他们的评分为垃圾评分. 为了提高推荐系统的性能, 在给用户做推荐时, 不仅要避免攻击对推荐结果的影响, 推荐系统还必须要消除或者减小那些不严谨的评分对推荐结果的影响.

故本文提出了结合用户声誉 (Reputation) 的鲁棒协同推荐算法, 给每个用户赋予相应的声誉系数, 对于声誉较低的用户通过声誉系数抑制其在推荐过程中的权重, 相当于依靠推荐算法内的健壮性平滑托攻击的不良影响, 并能修正系统中那些声誉较低的真实用户所带来的自然因素对精度的影响. 在攻击前, 我们识别出声誉系数过高的攻击用户并在推荐中删除. 该模型综合考虑了人为因素和自然因素, 并结合均值攻击来检测模型的防御效果, 并在 MovieLens 1M 数据集上进行验证. 实验结果表明, 与现有针对鲁棒性的模型相比, 这种模型不仅一定程度提升了推荐精度且大大提升了推荐系统的鲁棒性.

2 隐语义分解模型

自第一个协同过滤系统 Grundy 投入应用以来[14], 协同过滤算法作为目前使用最多、最成功的算法, 可以分为两类[15]: 基于记忆 (Memory-based) 的和基于模型的 (Model-based) 的算法. 基于记忆的算法根据系统中所有被打过分的产品信息进行预测. 通过找出用户之间的相似性或项目之间的相似性来进行推荐. 而基于模型的算法收集打分数据进行学习并建立用户行为模型, 进而利用模型对用户给产品的打分进行预测. 基于模型的算法其中一大类就是采用矩阵分解方法的隐语义模型构建的, 通

过降维计算用户矩阵和项目矩阵的内积来预测评分, 相比于基于记忆的算法, 这种算法的稳定性和精度都有所提升[16]. 故本文采用的基础算法为隐语义模型.

2.1 符号说明

这里, 我们介绍本文将会使用到的符号, 采用英文字母和希腊字母分别表示某个用户 i 和某个项目 α, 用户对项目的评分设为 $r_{i\alpha}$, 推荐系统的预测评分为 $\hat{r}_{i\alpha}$, 使用 $\boldsymbol{u}$、$\boldsymbol{v}$ 分别表示用户和项目的语义向量. 用户的声誉系数用 Cu_i 表示.

2.2 隐语义模型

隐语义模型 (Latent factor model)[17] 就是联系用户兴趣与物品发掘之间的隐含关系来构建模型的方法. 典型的隐语义模型从奇异值分解 (Singular value decomposition, SVD) 的角度将评分矩阵 R 分解为用户 U 和项目 V 的两个低维语义矩阵相乘的形式, 通过直接最小化训练集中的均方根误差 (Root mean square error, RMSE) 来学习矩阵, 最终得到用户对项目的评分的预测值, 损失函数如式 (1)[17].

$$C = \sum_{(i,\alpha)\in Train} r_{i\alpha} - \left(\mu + \boldsymbol{b}_i + \boldsymbol{b}_\alpha + \sum_{f=1}^{F} \boldsymbol{u}_{if}\boldsymbol{v}_{\alpha f}\right)^2 + \lambda(\|\boldsymbol{u}_i\|^2 + \|\boldsymbol{v}_\alpha\|^2 + \|\boldsymbol{b}_i\|^2 + \|\boldsymbol{b}_\alpha\|^2) \tag{1}$$

式中, 引入全局平均数 μ 是为了抵消不同评分系统的差异性, 使得到的预测评分是针对特定系统的. 偏置项 $\boldsymbol{b}_i$ 表示用户的评分与物品没有关系的那种因素, 如有的用户喜欢打高分, 有的用户喜欢打低分, 不同用户的偏好不同. 物品偏置项 $\boldsymbol{b}_\alpha$ 表示物品接受的评分和用户没有什么关系的那种因素, 如物品本身质量的原因, 质量高的物品评分相对比质量低的物品评分高. 增加的两项偏置项进一步增加了预测的准确性, 更适合实际的推荐系统, 使推荐更加的合理化. $\boldsymbol{b}_i$ 和 $\boldsymbol{b}_\alpha$ 同 $\boldsymbol{u}$、$\boldsymbol{v}$ 一样初始化后通过学习得到. 并为防止损失函数可能会导致学习的过拟合, 加入了防止过拟合项 $\lambda(\|\boldsymbol{u}_i\|^2 + \|\boldsymbol{v}_\alpha\|^2 + \|\boldsymbol{b}_i\|^2 + \|\boldsymbol{b}_\alpha\|^2)$. 这种带偏置的矩阵分解 (Bias SVD) 采用随机梯度下降法[17] 最小化式 (1) 的损失函数来得到 $\boldsymbol{u}$、$\boldsymbol{v}$、$\boldsymbol{b}_i$ 和 $\boldsymbol{b}_\alpha$. 根据 $\hat{r}_{i\alpha} = \mu + \boldsymbol{b}_i + \boldsymbol{b}_\alpha + \sum_{f=1}^{F} \boldsymbol{u}_{if}\boldsymbol{v}_{\alpha f}$ 计算出预测的评分. 上述模型对原始的评分矩阵降维后, 大大降低了运算的复杂度, 增强了系统的扩展性[18−19].

2.3 声誉推荐系统

声誉推荐系统通过收集和聚合一系列实体历史行为 (如: 供应商的提供历史或用户的评分历史) 的反馈信息来计算声誉值[20]. 声誉推荐系统主要分为内容驱使 (Content-driven) 和用户驱使 (User-driven) 两种系统[21]. 本文涉及到的声誉为用户驱使的系统中的声誉, 用户声誉的评估方法有多种, 文献 [22] 中使用 PageRank 理论计算用户的排名, 排名高的用户声誉高, 但这种方法是在基于信任列表的基础上计算的, 故不适合无信任列表的推荐系统或数据集. 文献 [23] 中提出一种 QTR (Quality-trust-reputation) 算法计算用户的声誉值, 但计算过程中参数过多且时间复杂度较高. 本文用户的声誉值采用 Zhou 等提出的声誉相关系数[24] 大小来衡量. 该声誉值通过皮尔森相关系数计算, 如式 (2).

$$Corr_i = \frac{1}{ku_i}\sum_{\alpha\in O_i}\left(\frac{r_{i\alpha}-\overline{r}_i}{\sigma_{r_i}}\right)\cdot\left(\frac{Qo_\alpha - \overline{Q}o_i}{\sigma_{Qo_i}}\right) \tag{2}$$

ku_i 表示用户的评分个数 (用户的度), O_i 表示用户 i 评过分的项目集合, $\overline{r}_i$, $\overline{Q}o_i$ 分别表示用户 i 评分的平均数和用户打过分项目的声誉加权平均分的平均数, σ_{r_i}, σ_{Qo_i} 分别表示各自的标准差. 其中, 项目的声誉加权平均分如式 (3).

$$Qo_\alpha = \frac{\sum\limits_{i\in U\alpha} Cu_i r_{i\alpha}}{\sum\limits_{i\in U\alpha} Cu_i} \tag{3}$$

如果一个用户的项目评分与用来衡量项目的声誉加权平均分相关性越高, 那么这个用户的声誉应该越高. 在文献 [24] 中, 当皮尔森相关系数小于 0 时将用户声誉设置为 0, 为了使每个用户在推荐过程中都发挥一定的作用, 以防删除过多声誉较低的用户给推荐精度方面带来不利的影响, 本文采用式 (4) 的方法使用户声誉的值域区间为 $[0, 1]$.

$$Cu_i = \frac{Corr_i + 1}{2} \tag{4}$$

结合式 (2) ∼ (4) 迭代计算出最终的声誉值.

图 1 为均值攻击下用户声誉的分布情况, 定义填充率 Filler size = ‖装填项目‖/‖总项目数‖, 攻击强度 Attack size = ‖攻击用户‖/‖真实用户‖. 从图 1 中可以看出, 真实用户的声誉值与攻击用户的有明显区别, 攻击用户的声誉较高. 故可以用该声誉系数来区别真实用户与攻击用户. 对于声誉过高的攻击用户, 我们在攻击前将其检测出并删除, 即将其声誉系数置 0.

为了更好地反映正常用户的整体声誉值范围, 图 2 为正常用户偏离整体声誉均值的概率分布图. 可以看出, 绝大部分正常用户都靠近在声誉均值的附近, 只有极少部分的用户的声誉值较大程度地偏

离均值, 如大于声誉均值 0.19 的用户比例只占整体用户数的 0.4 %. 攻击者为了避免很容易被发现, 往往采取均值填充的方法, 即让其对填充项目的打分接近于项目的均值, 使其伪装成声誉值过高的用户, 偏离正常用户的声誉值范围, 如图 1.

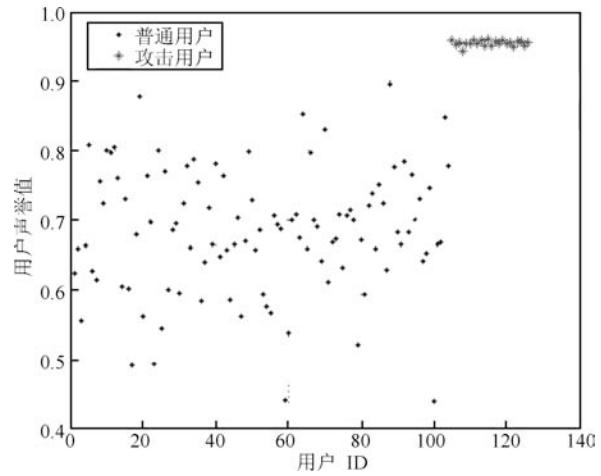


图 1　均值攻击下的用户声誉分布

Fig. 1　Reputation distribution under average attacks (Attack size = 10 %, Filler size = 5 %)

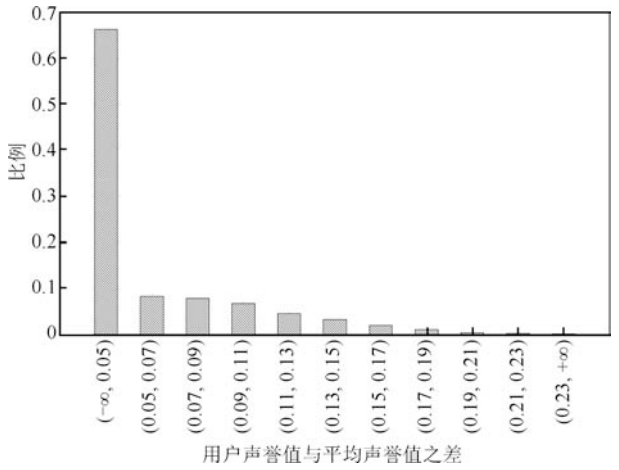


图 2　均值攻击下正常用户声誉概率分布

Fig. 2　Normal users′ reputation probability distribution under average attacks (Attack size =10 %, Filler size = 5 %)

因此我们可以设置阶段函数如式 (5). 其中 $\overline{C}u_i$ 为所有用户的声誉平均数, 在推荐过程中, 检测出声誉值过高的用户并删除. 并在第 3 节的实验过程中, 采用这样检测攻击用户的方式可以很准确地识别出攻击用户 (不同攻击比例下, 平均识别率为 99.55 %), 同时将正常用户误检测为攻击的概率也较低 (平均误判率为 1.7 %). 因此, 我们可以采用这种方式检测推荐系统中存在的攻击用户.

$$Cu_i = \begin{cases} Cu_i, & \text{若 } Cu_i - \overline{C}u_i < \beta \\ 0, & \text{若 } Cu_i - \overline{C}u_i > \beta \end{cases} \tag{5}$$

2.4　基于声誉的隐语义模型

将声誉系数引入到式 (1) 的 Bias SVD 模型中, 得到基于声誉的隐语义模型 (Reputation-bias SVD):

$$C = \sum_{(i,\alpha)\in Train} Cu_i \left(r_{i\alpha} - (\mu + \boldsymbol{b}_i + \boldsymbol{b}_\alpha + \sum_{f=1}^{F} \boldsymbol{u}_{if}\boldsymbol{v}_{\alpha f}) \right)^2 + \lambda(\|\boldsymbol{u}_i\|^2 + \|\boldsymbol{v}_\alpha\|^2 + \|\boldsymbol{b}_i\|^2 + \|\boldsymbol{b}_\alpha\|^2) \tag{6}$$

在最优化损失函数的过程中, 当声誉值越高, 即迫使式 (6) 中预测值项与该用户的评分越接近, 声誉值高的用户对推荐结果影响越大. 反之则可以削弱那些声誉值低的用户对推荐结果的影响. 对于实际系统中打分不严谨的自然噪声用户, 其打分与项目的声誉加分平均分相关性较低, 即不严谨用户声誉较低, 削弱自然噪声对推荐结果的影响, 因而达到提升推荐质量的目的. 采用式 (5) 的方法检测出声誉过高的攻击用户, 并在推荐过程中删除, 达到提升推荐鲁棒性的效果.

使用随机梯度下降法优化上述模型. 结合用户声誉的隐语义模型的具体算法过程如算法 1, 利用算法 1 求出最终的 $\boldsymbol{u}$、$\boldsymbol{v}$, 再根据公式 $\hat{r}_{i\alpha} = \mu + \boldsymbol{b}_i + \boldsymbol{b}_\alpha + \sum_{f=1}^{F} \boldsymbol{u}_{if}\boldsymbol{v}_{\alpha f}$ 即可求出相应的预测评分.

算法 1. 基于用户声誉的 Bias SVD

输入. 用户 – 项目评分矩阵 R, 语义维数 F, 随机梯度下降法的学习速率 g (迭代步长).

输出. 用户语义向量 $\boldsymbol{u}$, 项目语义向量 $\boldsymbol{v}$.

步骤 1. 采用随机数填充的方法初始化 $\boldsymbol{u}$、$\boldsymbol{v}$, 全零填充用户、项目偏置项 $\boldsymbol{b}_i$ 和 $\boldsymbol{b}_\alpha$;

步骤 2. 计算系统评分的全局平均数 μ 和每个用户在系统中的声誉系数 Cu_i;

步骤 3. 计算 $\frac{\partial C}{\partial \boldsymbol{b}_i}$、$\frac{\partial C}{\partial \boldsymbol{b}_\alpha}$、$\frac{\partial C}{\partial \boldsymbol{u}}$ 和 $\frac{\partial C}{\partial \boldsymbol{v}}$;

步骤 4. $\boldsymbol{b}_i \leftarrow \boldsymbol{b}_i - g\frac{\partial C}{\partial \boldsymbol{b}_i}$, $\boldsymbol{b}_\alpha \leftarrow \boldsymbol{b}_\alpha - g\frac{\partial C}{\partial \boldsymbol{b}_\alpha}$;

步骤 5. $\boldsymbol{u} \leftarrow \boldsymbol{u} - g\frac{\partial C}{\partial \boldsymbol{u}}$, $\boldsymbol{v} \leftarrow \boldsymbol{v} - g\frac{\partial C}{\partial \boldsymbol{v}}$;

步骤 6. 当 $\boldsymbol{u}$、$\boldsymbol{v}$ 不收敛, 转至步骤 3.

3　实验分析

为了衡量将声誉添加到推荐系统中的作用, 我们将从自然噪声和托攻击两方面分别分析用户声誉系数在推荐过程中所起的作用, 即其对推荐的精度和鲁棒性的影响. 下面先介绍算法的衡量指标.

3.1 衡量指标

衡量推荐算法的指标有很多[19], 本文使用常用的绝对平均误差 (Mean absolute error, MAE) 和预测偏差 (Prediction shift, PS)[25] 两种指标分别衡量评分预测推荐的准确性与鲁棒性.

准确性指标 MAE 用于衡量真实评分与预测评分间的平均偏差. 设 T 为测试集中评分记录的数目, $r_{i\alpha}$ 和 $\hat{r}_{i\alpha}$ 分别为真实评分和预测评分, 则 MAE 定义为

$$MAE = \frac{1}{T}\sum_{i,\alpha\in T}|r_{i\alpha} - \hat{r}_{i\alpha}| \tag{7}$$

MAE 越小, 说明算法的推荐精度越好.

鲁棒性指标 PS 用于衡量托攻击前后目标项的评分预测偏差. 令 $\hat{r}_{i\alpha}$ 和 $\hat{r}'_{i\alpha}$ 分别为托攻击前后的预测评分, 则 PS 定义为

$$PS = \frac{1}{T}\sum_{i,\alpha\in T}|\hat{r}_{i\alpha} - \hat{r}'_{i\alpha}| \tag{8}$$

同样, PS 越小, 说明算法的鲁棒性越好.

3.2 数据集

实验数据集为 MovieLens 1M (www.grouplens.org), 包含了 6 040 个用户对 3 952 部电影的 1 000 209 个评分, 每个用户至少对 20 部电影评过分, 评分范围为 1～5, 代表喜好程度从低到高. 将数据集平均分成五份, 采用五折交叉验证的方法进行实验, 训练集与测试集的大小比例为 4∶1, 并保证不存在冷启动问题[26], 即所有的测试集中的用户和项目在训练集中都有评分. 图 3 为各模型的语义维数的选取, 在下列实验中 Reputation-bias SVD 模型以及其他参照模型的语义维数都设置为 10.

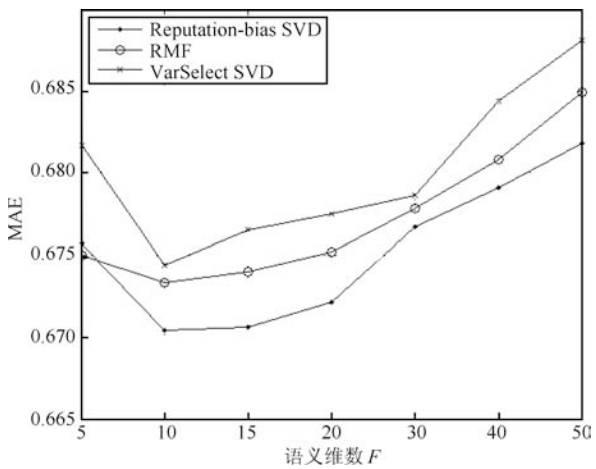


图 3 语义维数的选取

Fig. 3 Dimension selection of factor

3.3 β 的选取

作为区分真实用户与攻击用户的阈值, 为了更好地说明其作用, 我们定义识别率 (True rate) T 和误判率 (False rate) F 来评估检测攻击用户的结果, 如式 (9) 与式 (10).

$$T = \frac{TA}{TA + FA} \tag{9}$$

$$F = \frac{FN}{FN + TN} \tag{10}$$

其中, TA 表示正确检测出的攻击用户数, FA 为未被检测出的攻击用户数, FN 为错误检测出的正常用户数, TN 为未被检测出的正常用户数. 如图 4, 为 β 对检测攻击用户精度 (T 和 F) 和推荐误差 (MAE 与 PS) 的影响, 当 β 的值在 0.13 和 0.21 之间, 可以保证很高的识别率 (接近 100 %), 但是其误判率也相对较高. 当 β 大于 0.21 时, 误判率渐渐下降, 但是识别率也以较快的速率下降. 所以, 我们结合 β 对推荐误差的影响来选择 β 的值. 如图 4, 一方面, β = 0.19 时, 其预测偏差最小, 另一方面, β 的取值对推荐的精度指标 MAE 影响不大, 故选择 β = 0.19.

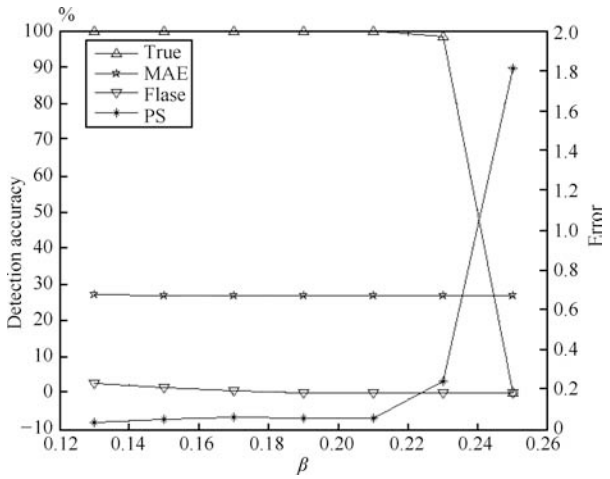


图 4 阈值 β 对推荐结果的影响

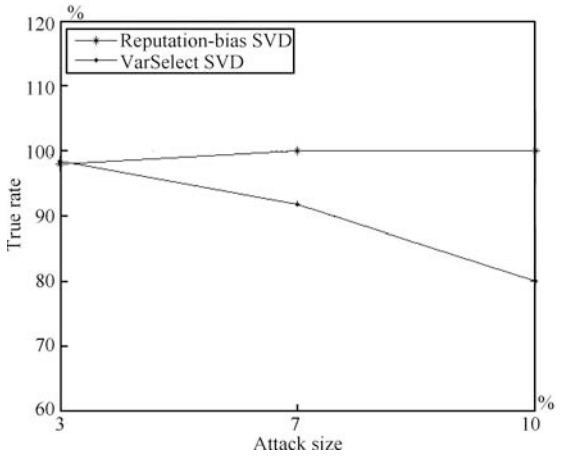
Fig. 4 Effect of threshold β on the recommendation

3.4 托攻击的鲁棒性分析

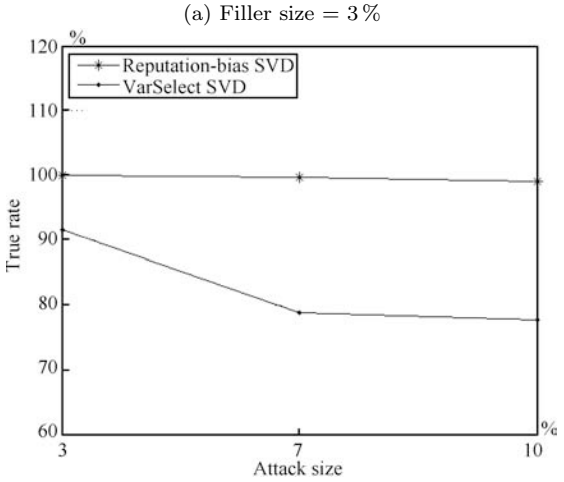
为了评估算法的鲁棒性, 本实验采用填充率为 3 %, 5 %, 7 %, 10 % 以及不同的攻击强度 (3 %, 7 %, 10 %) 来对比不同算法的性能.

图 5 为算法 Reputation-bias SVD 与 VarSelect SVD 均值攻击下攻击用户检测精度比较曲线. 从图 5 (a) 和 5 (b) 中可以看出, 当填充率从 3 % 提高到 5 % 时, Reputation-bias SVD 算法的平均检测识别率为 99.55 %, 但 VarSelect SVD 算法随着攻击强度 (3 %～10 %) 或者填充率 (3 %～5 %)

的增加, 其对均值攻击进行检测的准确率逐渐降低, 这是由于 VarSelect SVD 算法是通过主成分分析 (PCA) 标示出攻击用户, 其认为攻击用户对系统贡献的信息较少, PCA 得分低的用户即为攻击用户. 但是随着攻击强度或者填充率的增大, 攻击用户的 PCA 得分相应会增大, 因此导致检测识别率降低. 而本文提出的 Reputation-bias SVD 算法是根据用户声誉的分布情况设计阈值来过滤攻击用户, 其中攻击用户声誉过高, 且随着攻击强度或填充效率的增加, 攻击用户的声誉仍然会偏离正常的范围, 所以不会出现检测识别率大幅下降的情况.



(a) Filler size = 3 %



(b) Filler size = 5 %

图 5　预测偏移量对比

Fig. 5　Comparison of the PS

图 6 为不同算法采用不同的攻击强度和填充率均值攻击下对预测偏移量 PS 的影响. 从图 6 中可知, RMF 算法的鲁棒性能最差, VarSelect SVD 算法在鲁棒性能上相比于 RMF 算法有较大提升, 但随着填充率或攻击强度的增加, 其检测攻击用户的准确率随之下降, 故导致其鲁棒性能不能得到很好的保证, 本文提出的基于声誉的鲁棒协同算法 Reputation-bias SVD 只有当填充率和攻击强度同为 3 % 时的鲁棒性不如 VarSelect SVD 算法, 其余情况都优于其他算法, 且随着攻击强度和填充率的增大, 鲁棒性能并未出现明显降低, 相比其他算法提升程度很大.

3.5　对自然噪声的影响

为了评估用户声誉在推荐过程中对自然噪声的影响, 分别采取训练集比例为 70 %, 80 %, 90 % 且在没有攻击的情况下对比 Bias SVD 算法添加声誉前和添加声誉后的推荐精度. 不加声誉的情况, 即令式 (8) 中的声誉为 1, 且对加声誉的 3 组实验我们设定的声誉阈值 β 都为 0.19. 如表 1, 对比添加声誉前后的算法推荐精度 MAE, 可以看出相比于普通的 Bias SVD 算法, 含有声誉的 Reputation-bias SVD 算法在 3 组不同训练集情况下其 MAE 均有提升. 添加用户声誉后过滤掉系统中部分用户, 系统的推荐精度得到提升, 并且训练集比例越高, 推荐精度越高, 说明过滤掉的部分用户对于系统来说为自然噪声, 因此本文提出的 Reputation-bias SVD 算法有能力过滤掉系统的自然噪声达到提升推荐精度的目的.

表 1　用户声誉在无攻击情形下对推荐精度 MAE 的影响

Table 1　Effect of user′ reputation on the MAE with no attack

Train (%)	Bias SVD with no attack	Reputation-bias SVD with no attack
70	0.6799	0.6785
80	0.6715	0.6702
90	**0.6690**	**0.6673**

为进一步评估本模型在过滤自然噪声提升推荐精度的性能, 我们采用向系统中注入垃圾评分的方式评估添加声誉后对模型的精度提升性能. 实际系统的自然噪声, 即垃圾评分, 按照用户的评分习惯主要分为两种. 一种是用户对评分没有主观意识, 偏好打物品的平均分, 很明显这种自然噪声, 在推荐的过程中对推荐精度无影响. 另一种噪声是用户不严谨地打分, 通常为随意打分, 这种自然噪声会对推荐精度产生影响. 因此, 我们注入若干个用户, 随机对 10 % 的项目进行填充, 填充评分值为 1 ∼ 5 的随机数. 表 2 为注入噪声后, 添加声誉前后两种模型的推荐精度对比. 可以看出, 随着垃圾用户数的增加, 基本的 Bias SVD 模型推荐精度很不稳定, 越来越低, 而 Reputation-bias SVD 模型的推荐精度基本不变, 且随着垃圾用户数的增加, 相比于 Bias SVD 模型的推荐精度提升的越多, 当垃圾用户数为 900 时, 精度提升了 1.1 %. 因此, Reputation-bias SVD 模型有过滤自然噪声的能力, 且系统中的噪声越多, 其提升的推荐精度越明显.

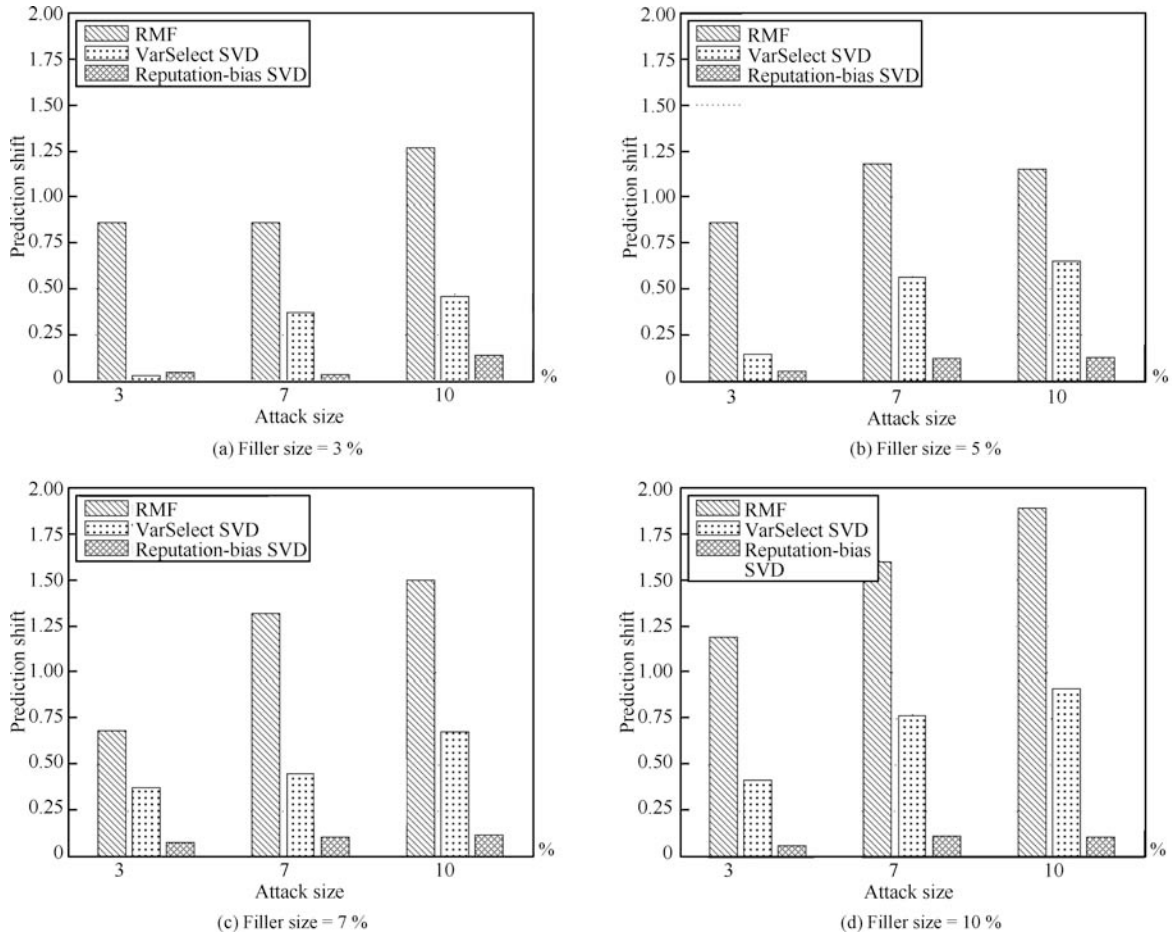


图 6 预测偏移量对比

Fig. 6 Comparison of the PS

表 2 注入噪声后用户声誉对推荐精度 MAE 的影响

Table 2 Effect of user′ reputation on the MAE with noise injection

垃圾用户数	Bias SVD	Reputation-bias SVD
300	**0.6736**	**0.6704**
600	0.6774	0.6720
900	0.6801	0.6724

3.6 推荐精度分析

表 3 为不同算法在不同攻击组合下推荐精度的比较. 从表中可以看出, 本文提出的基于用户声誉的鲁棒协同算法 Reputation-bias SVD 的推荐精度都优于其他两种算法, RMF 算法次之. 相比于 RMF 算法提升了 0.5 %, 相比于 VarSelect SVD 算法提升了 0.8 %. 这是由于采用了增加了系统、用户和项目的偏置项的 Bias SVD 算法, 实验结果说明, 这些偏置项有利用提升系统的推荐精度. 综合表 1 和表 3 来看, 其中推荐精度的提升还有部分是来自对自然噪声的过滤, 所以总的来说, 本文提出的 Reputation-bias SVD 算法在推荐精度上也获得了提升.

表 3 推荐精度 MAE 对比

Table 3 Comparison of the MAE

Attack Size (%)	Filler Size (%)	RMF	VarSelect SVD	Reputation-bias SVD
3	3	0.6749	0.6744	**0.6706**
	5	0.6757	0.6757	**0.6697**
	7	0.6747	0.6755	**0.6702**
	10	0.6748	0.6759	**0.6703**
7	3	0.6733	0.6760	**0.6698**
	5	0.6734	0.6751	**0.6702**
	7	0.6733	0.6757	**0.6697**
	10	0.6724	0.6757	**0.6705**
10	3	0.6735	0.6741	**0.6704**
	5	0.6729	0.6780	**0.6701**
	7	0.6726	0.6741	**0.6711**
	10	0.6722	0.6734	**0.6703**

4 结束语与展望

本文针对推荐系统的鲁棒性这一问题, 结合协同过滤算法中的隐语义模型并引入用户声誉系

数, 提出基于声誉的鲁棒协同过滤算法 Reputation-bias SVD. 算法通过用户声誉系数对系统中一些不严谨用户在推荐过程的作用进行了限制, 即过滤掉自然噪声, 以此来提升精度, 并通过设置阶段函数检测出声誉过高的攻击用户对系统的攻击. 通过在 Movielens 1M 数据集上的实验, 验证了本文的基于声誉的鲁棒协同过滤算法在自然噪声和托攻击方面均有效果. 通过与现有的鲁棒性推荐算法相比, 本文提出的算法具有形式简单、可解释性强、稳定的特点, 不仅对系统的鲁棒性能有很大程度的提升, 推荐精度也有提升.

随着推荐系统的发展, 推荐系统的鲁棒性变得尤为重要. 因此, 未来关于推荐系统鲁棒性的研究工作主要围绕下面两个方面进行展开: 1) 探索更合理的声誉系数, 建立一个适合更多推荐系统的声誉系统; 2) 尝试将社交信息利用到声誉系统内, 通过社交信息建立朋友之间的声誉系数, 发掘能提升推荐系统性能的社交信息.

References

1 Resnick P, Iakovou N, Sushak M, Bergstrom P, Riedl J. GroupLens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 Computer Supported Cooperative Work. Chapel Hill: ACM, 1994. 175−186

2 Hill W C, Stead L, Rosenstein M, Furnas G W. Recommending and evaluating choices in a virtual community of use. In: Proceedings of the 1995 SIGCHI Conference on Human Factors in Computing Systems. Denver: ACM, 1995. 194−201

3 Lam S K, Riedl J. Shilling recommender systems for fun and profit. In: Proceedings of the 13th International Conference on World Wide Web. New York, USA: ACM, 2004. 393−402

4 O′Mahony M P, Hurley N J, Kushmerick N, Silvestre G C M. Collaborative recommendation: a robustness analysis. *ACM Transactions on Internet Technology (TOIT)*, 2004, **4**(4): 344−377

5 Mobasher B, Burke R, Sandvig J J. Model-based collaborative filtering as a defense against profile injection attacks. In: Proceedings of the 21st National Conference on Artificial Intelligence and the 18th Innovative Applications of Artificial Intelligence Conference. Boston, Massachusetts, USA: AAAI, 2006.

6 Gunes I, Kaleli C, Bilge A, Polat H. Shilling attacks against recommender systems: a comprehensive survey. *Artificial Intelligence Review*, 2014, **42**(4): 767−799

7 Mobasher B, Burke R, Williams C, Bhaumik R. Analysis and detection of segment-focused attacks against collaborative recommendation. In: Proceedings of the 7th International Workshop on Knowledge Discovery on the Web. Chicago, IL: Springer Berlin Heidelberg, 2006. 96−118

8 Burke R D, Mobasher B, Williams C, Bhaumik R. Classification features for attack detection in collaborative recommender systems. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Philadelphia, PA, USA: ACM, 2006. 542−547

9 Mehta B, Nejdl W. Attack resistant collaborative filtering. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM, 2008. 75−82

10 Li Cong, Luo Zhi-Gang. A metadata-enhanced variational Bayesian matrix factorization model for robust collaborative recommendation. *Acta Automatica Sinica*, 2011, **37**(9): 1067−1076
(李聪, 骆志刚. 用于鲁棒协同推荐的元信息增强变分贝叶斯矩阵分解模型. 自动化学报, 2011, **37**(9): 1067−1076)

11 O′Mahony M P, Hurley N J, Silvestre G C M. Efficient and secure collaborative filtering through intelligent neighbor selection. In: Proceedings of the 16th European Conference on Artificial Intelligence. Valencia, Spain: IOS Press, 2004. 383−387

12 Mehta B, Hofmann T, Nejdl W. Robust collaborative filtering. In: Proceedings of the 2007 ACM Conference on Recommender Systems. New York, USA: ACM, 2007. 49−56

13 Huber P J. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 1964, **35**(1): 73−101

14 Rich E. User modeling via stereotypes. *Cognitive Science*, 1979, **3**(4): 329−354

15 Liu Jian-Guo, Zhou Tao, Wang Bing-Hong. The research progress of personalized recommendation system. *Progress in Natural Science*, 2009, **19**(1): 1−15
(刘建国, 周涛, 汪秉宏. 个性化推荐系统的研究进展. 自然科学进展, 2009, **19**(1): 1−15)

16 Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Las Vegas, Nevada, USA: ACM, 2008: 426−434

17 Koren Y, Robert B, Chris V. Matrix factorization techniques for recommender systems. *Computer*, 2009, **42**(8): 30−37

18 Vozalis M G, Margaritis K G. Applying SVD on item-based filtering. In: Proceedings of the 5th International Conference on Intelligent Systems Design and Applications. Greece: IEEE, 2005. 464−469

19 Xiang Liang. *Recommendation System Practice*. Beijing: Posts and Telecom Press, 2012.
(项亮. 推荐系统实践. 北京: 人民邮电出版社, 2012.)

20 Li R H, Yu J X, Huang X, Cheng H. Robust reputation-based ranking on bipartite rating networks. In: Proceedings of the 2012 SDM International Conference on Data Mining. Hong Kong, China: SIAM, 2012. 612−623

21 De Alfaro L, Kulshreshtha A, Pye I, Adler T. Reputation systems for open collaboration. *Communications of the ACM*, 2011, **54**(8): 81−87

22 Tang J L, Hu X, Gao H J, Liu H. Exploiting local and global social context for recommendation. In: Proceedings of the 12nd International Joint Conference on Artificial Intelligence. Bellevue, Washington, USA: AAAI Press, 2013. 2712−2718

23 Liao H, Cimini G, Medo M. Measuring quality, reputation and trust in online communities. In: Proceedings of the 20th International Symposium on Foundations of Intelligent Systems. Macau, China: Springer Berlin Heidelberg, 2012. 405−414

24 Zhou Y B, Lei T, Zhou T. A robust ranking algorithm to spamming. *EPL (Europhysics Letters)*, 2011, **94**(4): 48002

25 Mobasher B, Burke R, Bhaumik R, Williams C. Toward trustworthy recommender systems: an analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology*, 2007, **7**(4): 1−40

26 Lv L Y, Medo M, Yeung C H, Zhang Y C, Zhang Z K, Zhou T. Recommender systems. *Physics Reports*, 2012, **519**(1): 1−49

张燕平 安徽大学计算机科学与技术学院教授. 主要研究方向为商空间与智能计算. E-mail: zhangyp2@gmail.com
(**ZHANG Yan-Ping** Professor at the School of Computer Science and Technology, Anhui University. Her research interest covers quotient space and intelligent computing.)

张 顺 安徽大学计算机科学与技术学院硕士研究生. 2013 年获得安徽大学学士学位. 主要研究方向为社交网络与个性化推荐. E-mail: zhangs.ahu@gmail.com
(**ZHANG Shun** Master student at the School of Computer Science and Technology, Anhui University. He received his bachelor degree from Anhui University in 2013. His research interest covers social networks and personalized recommendation.)

钱付兰 安徽大学计算机科学与技术学院博士研究生. 2005 年获得安徽大学硕士学位. 主要研究方向为社交网络与个性化推荐. 本文通信作者.
E-mail: qianfulan@hotmail.com
(**QIAN Fu-Lan** Ph. D. candidate at the School of Computer Science and Technology, Anhui University. She received her master degree from Anhui University in 2005. Her research interest covers social networks and personalized recommendation. Corresponding author of this paper.)

张以文 安徽大学计算机科学与技术学院副教授. 分别于 2006 年和 2013 年获得合肥工业大学硕士学位和博士学位. 主要研究方向为服务计算, 云计算和电子商务.
E-mail: zywahu@qq.com
(**ZHANG Yi-Wen** Associate professor at the School of Computer Science and Technology, Anhui University. He received his master and Ph. D. degrees from Hefei University of Technology in 2006 and 2013, respectively. His research interest covers service computing, cloud computing, and e-commerce.)