**利用k极大团检测意见垃圾用户群**

Detecting Opinion Spammer Groups via k-Clique

**摘要**：电子商务的发展促使在线用户评论数量急剧增长，产品评论成为人们上网购物的重要参考信息。随着在线评论对商品销量及商家名誉的影响力越来越大，某些组织或个人在各种利益的驱动下开始利用网络信息监管的缺失，制造虚假评论混淆视听，误导用户。目前国内外学者已经提出了很多方法用于检测意见垃圾和意见垃圾用户，但是关于意见垃圾用户群的检测问题一直没有得到广泛的研究。本文利用k极大团来检测意见垃圾用户群。所谓的k极大团就是由k个节点构成的连通图，这样的k极大团关系非常密切，这样高度密集的区域很有可能是具有强烈可疑性的垃圾用户群。本文利用评论者二分图的拓扑结构，提出了一种改进的派系过滤算法用以检测评论关系图中的k极大团，以此检测出意见垃圾用户群。给定一个评论者二分图，本方法识别其垃圾分数超过给定的垃圾阈值的所有极大团。对于大型不可疑的极大团，利用最小切割算法分为两个子图，进一步利用本方法进行递归处理。实验研究表明，我们提出的方法优于几个最先进的基线。

**关键字**：评论垃圾，垃圾用户群检测，社区发现，派系过滤算法

# 1.引言

Web2.0的出现使人们由单纯的网络信息接受者转向信息贡献和创造者。电子商务的发展促使在线用户评论数量急剧增长，产品评论成为人们上网购物的重要参考信息。购买商品或消费前，用户往往会查看相关评论信息，如果评价积极，消费者的购买意向可能就会很大。因而随着网络应用的不断深入，在线“网络口碑”对商品销量及商家名誉的影响力越来越大。据哈佛大学研究报道，美国最大的评论网站Yelp的产品评分每增长一星，将为该产品创造5-9%的收入[1]。

令人震惊的是，Yelp网站中14-20%的评论都是虚假评论[2]。某些组织或个人在各种利益的驱动下开始利用网络信息监管的缺失，弄虚作假，制造虚假评论混淆视听，误导用户。不良商家会雇佣一批人撰写虚假好评以此来获得消费者的好感，或撰写虚假差评以此来诋毁对手的产品或服务。撰写虚假评论的人被称为***意见垃圾用户***（又称评论垃圾用户），意见垃圾用户撰写的虚假评论被称为***意见垃圾***（又称垃圾评论），受到虚假好评或虚假差评的产品被称为***目标产品***。垃圾评论问题由来已久，现今的垃圾评论活动已经衍生为团伙作案。针对某一目标产品，一群评论者一起发布大量的好评以此来提升产品的声誉，或者一起发布大量的差评以此来诋毁某产品，这样的一群协同工作的评论者被称为***意见垃圾用户群***（又称评论垃圾用户群）。如果某产品在发布初期就遭受到意见垃圾用户群的攻击，那么攻击者将完全掌控消费者的情绪，这对产品的影响将是毁灭性的。清除网络垃圾，净化网络环境，为人们提供一个真实可信的信息获取平台的需求日益迫切。因此，关于意见垃圾用户群检测问题的研究，已成为国内外研究的热点，具有重要的应用价值。

自从N. Jindal和Liu Bing在文献[3]中第一次提出意见垃圾检测问题以来，垃圾意见/意见垃圾用户检测问题受到了广泛的关注，研究方向主要分3类：垃圾意见检测[4,5]、意见垃圾用户检测[3,6-9]和意见垃圾用户群检测[10-15]。目前国内外学者已经提出了很多方法用于检测意见垃圾和意见垃圾用户，但是关于意见垃圾用户群的检测问题一直没有得到广泛的研究。意见垃圾检测技术正在不断发展，因不同领域而异。因此，在检测意见垃圾活动中没有一个适用于所有的解决方案。虽然研究人员提出了各种垃圾检测技术，但是并没有得到整体的胜利。

本课题在前人的研究基础上，采用基于网络结构的方法，通过发现评论者二分图中的极大团，以此来检测意见垃圾用户群。社区发现领域的派系过滤算法可以检测出这种k极大团。已经有研究表明评论用户之间存在隐式社区[16]。所以，我们有理由相信，通过社区检测技术可以找到评论者二分图中个体之间关系紧密的子图，这样的子图可能是具有强烈可疑性的垃圾用户群。本文的贡献如下：

(1)我们利用k极大团来反映垃圾用户群之间相互协作的紧密关系。

(2)我们将评论数据建模为一个评论者二分图，评论网络中的节点代表现实生活中的评论者，评论者因为共同评论一个产品而建立一条关系边，边权重由评论时间间隔和评级分数偏差来决定，提出了b种改进的的派系过滤算法用以检测评论关系图中的k极大团，以此检测出意见垃圾用户群。给定一个评论者二分图，本方法识别其垃圾分数超过给定的垃圾阈值的所有极大团。对于大型不可疑的极大团，利用最小切割算法分为两个子图，进一步利用本方法进行递归处理。

(3) 我们利用现实世界的数据集进行了实验，以评估我们提出的方法的性能。实验结果表明，我们提出的方法可以找到高质量的垃圾用户群，并且优于一些最先进的方法。

本文余下部分的组织如下。第二部分讨论了相关工作。第三部分对本文涉及的概念进行精确了的定义。第四部分介绍了构建评论数据的二分图模型的方法和所使用的垃圾指标，以及提出了垃圾用户群检测算法。第五部分公布了我们所采用的方法的实验结果。最后，第6部分，我们对本文进行了总结，并且介绍了未来工作。

# 2.相关工作

目前国内外学者已经提出了很多方法用于垃圾意见检测/意见垃圾用户检测，但大多数方法主要侧重于开发纯基于内容的分类器[4,5,8,11,17]。这些方法背后的基本思想是通过分析评论内容来检测垃圾意见/意见垃圾用户。然而，这种纯基于内容的分类器目前已经进入了一个瓶颈期，原因如下：(1)垃圾用户可以轻松地操纵评论内容，避免检测[11,17]。例如，如果重复的文本评论被认为是垃圾意见，垃圾用户可能会简单地对内容进行更改。(2)纯基于内容的分类器经常被设计为特定的应用领域，不能轻易应用于不同的领域[8]。(3)因为大多数基于内容的分类器通常需要真实的标签，但是带标注的真实数据集通常很难获得。以前的一些研究通常采用人工标注数据，然而，这种方法的高成本使得大规模数据集不可能这样做[18]。

最近，意见垃圾用户群的检测问题开始受到学者的关注。与垃圾意见检测和意见垃圾用户检测相比，针对意见垃圾用户群的检测具有天然的优势，因为群体作案是意见垃圾用户的常用手段，并且群体检测可以提高检测效率，所以研究意见垃圾用户群的检测问题必然成为未来的一个趋势。目前，关于意见垃圾用户群的检测研究还比较少，检测方法主要分为基于频繁项集挖掘(Frequent itemsets mining, FIM)的方法和基于网络结构的方法。

## A、基于FIM的方法

文献[10]首先引入FIM技术来生成候选意见垃圾用户群。他的方法主要分为三步。第一步，通过FIM技术找到候选垃圾用户群。他们将评论者作为项目(Item)，产品作为事务(Transactions)，例如，通过将最小支持数量设置为3，可以检测到至少两个评论者针对至少3个产品进行评论的情况。第二步，他们提出了8个垃圾指标以衡量垃圾用户群的可疑性。第三步采用SVM Rank算法产生候选垃圾用户群的垃圾程度(Spamicity)的排名。文献[11]中也采用了与文献[10]类似的三步法，首先通过FIM技术找到候选垃圾用户群；然后提出了8个群组垃圾行为指标和4个个人垃圾行为指标；最后提出GSRank算法，使用迭代计算框架对候选人进行排名，这个迭代计算框架可以反映候选群组、目标产品和个体评论者之间的关系。文献[19]采用基于KNN的方法和基于图的分类方法来预测FIM生成的候选垃圾用户群里的每个评论者是否是垃圾用户。文献[20]提出了一种统计模型，利用EM算法来计算FIM生成的候选垃圾用户群的每个群组成员之间的联系。

使用FIM来生成候选垃圾用户群有许多缺点[12]：(1)FIM中使用的最小支持数不能小于3，即每个候选垃圾用户群只检测到了至少2个评论者针对至少3个产品进行评论的情况。然而，在现实世界中，旨在推广一种或两种产品的群组非常普遍。(2)现实生活中垃圾用户群通常必须在预先限定的时间内完成任务，而基于FIM的方法不会考虑到生成候选垃圾用户群的时间段。在FIM生成的候选垃圾用户群中，很多在不同的时间段评论相同产品的正常用户也可能碰巧被包含进来，所以这导致候选用户群的质量很低。(3)在每个FIM生成的候选垃圾用户群中，每位用户必须评论该群组每个成员所评论的所有产品，这一点也很容易使垃圾用户轻易绕过检测。基于网络结构的方法[12,13]可以很好的解决上述问题。

## B、基于网络的方法

文献[21]提出了一个新的两步法用于发现垃圾用户群及其目标产品。首先，介绍了NFS（网络足迹分数，量化产品成为垃圾活动目标的可能性），并利用NFS发现所有可疑的产品，即垃圾活动的目标产品；然后，从可疑度排名靠前的产品的诱导子图中聚类得到垃圾用户群。文献[14]提出了一种三阶段的结合聚类技术的方法，通过为每个用户分配怀疑分数来识别个人垃圾用户和垃圾用户群。该方法基于评论网络，一个评论网络由3个节点组成，分别是评论者，评论和产品。阶段一计算了评论图中每个节点的怀疑得分。阶段二是向前和向后迭代地更新在阶段一中获得的怀疑分数。向前更新是指从评论者节点更新产品的怀疑分数。反向更新是指从产品节点更新评论者的怀疑分数。阶段三专注于辨别垃圾用户群和计算群组垃圾分数，在这个阶段执行KMeans聚类。该方法在具有成千上万评论的产品上表现良好，在只有几十个或几百个评论的产品上表现不佳。

文献[16]证明了评论用户之间存在隐式社区。文献[22]提出了一种基于用户互动加上情绪分析的无监督混合方法。通过对从亚马逊收集的数据集进行广泛的实验，他们发现垃圾用户故意建立强大的积极社区，使自己的观点具有影响力。

文献[12]采用评论者的二分图投影描述垃圾用户群。首先，他们将评论者之间的相似性（权重）定义为两个相邻评论者共同评论的产品数量。然后，将大型的图迭代分割为k连通图，以此找到松散的垃圾用户群(Loose Spammer Group)，即群成员不需要评论群组内所有的目标产品，这样就可以检测到至少两个评论者针对至少1个产品进行评论的情况。最后，利用自定义的垃圾指标计算该用户群的垃圾程度并进行排名。该论文有一个缺陷，即它在定义评论者之间的相似性时没有考虑评论的时间间隔，而评论的时间间隔对于垃圾用户群检测是非常重要的，因为垃圾用户往往在一个特定的时间段集体活动。文献[13]弥补了这个缺陷。首先，采用双连通图描述垃圾用户群，评论者之间的相似性（权重）定义为评论时间间隔和评分分数偏差。然后，利用最小切割算法将大型双连通图拆分。最后，利用自定义的垃圾指标计算用户群的垃圾程度并进行排名。

## C、其他方法

文献[23]指出，心理语言特征可以在群体检测中发挥更重要的作用。文献[15]提出了一种使用社交网络分析指标作为特征来检测垃圾用户的新方法。产品评论的作者与评论的回复之间的关系被用于垃圾用户群检测。文献[24]使用网络参数和地理统计信息（如MAC地址、IP地址、位置、日期时间和浏览器ID等隐私数据）来检测垃圾评论。该方法的缺点是没有办法进行实验验证，因为评论的隐私数据无法得到。文献[25]使用作者话题模型来为每个评论者建立兴趣资料，基于兴趣向量的consin相似度（或KL距离）来计算不同评论者之间的相似性，通过评论者之间的距离发现候选垃圾用户群。我们提出的方法把评论数据建模为评论者二分图，利用k极大团来检测意见垃圾用户群。

# 3.问题定义

在我们定义垃圾用户群的数据模型之前，我们对垃圾用户群活动做出以下假设。(1)评论者必须在相对较短的时间内完成垃圾活动，才能对目标产品产生最大的影响；(2)垃圾用户进行高分好评以推广某目标产品，或进行低分差评以诋毁某目标产品；(3)同一个活动中的每个评论者不一定会评论所有的目标产品，也就是说，评论者可以通过评论目标产品的一个子集来平衡评论任务。

在本节中，我们提出了连通的垃圾用户群概念及其数据模型。为了便于陈述，我们在表1中列出了相关的符号及其含义。

表1 符号表

|  |  |
| --- | --- |
|  | 一个评论者 |
|  | 一个评论 |
|  | 一个产品 |
|  | 一个垃圾用户群 |
|  | 评论者集 |
|  | 评论集 |
|  | 目标产品集 |
|  | 的垃圾评论者集 |
|  | 的目标产品集 |
|  | 的评论集 |
|  | 的垃圾指标 |
|  | 的垃圾用户群网络 |
|  | 的评分函数 |
|  | 产品的评论者集 |
|  | 被评论的产品集 |
|  | 评论者对产品的评论时间 |
|  | 评论者对产品的5星评分 |

***定义1******垃圾用户群****：*垃圾用户群被建模为一个六元集的形式，是用户指定的时间范围，用来选择中合适的评论，只有发生在时间范围内的评论会被考虑；是垃圾用户群中的评论者集；是垃圾用户群中的目标产品集，其指的是在时间范围内被中至少两个评论者共同评论的产品；是在共同评论时间范围内，由中的评论者对中的产品撰写的评论集，；是垃圾指标，用于测量不同维度的的垃圾程度；是的评分函数，它测量的垃圾程度。

在本方法中，起着非常重要的作用，因为在垃圾用户群检测中，检测的时间范围的设定非常重要，时间范围设置过大，很多真实的共同评论会被包括进来，时间范围设置过小又会造成漏检。

**定义2** ***共同评论的相关性***：给定评论者，如果和共同评论产品，我们定义的相关性为：

 (1)

注意，如果评论者和共同评论产品超过时间窗口或其评分偏差大于2，则不会考虑共同评论。

**定义3**  ***垃圾指标***：给定一个垃圾用户群，其垃圾分数由表示，范围从0到1，是垃圾指标的函数，。群组垃圾分数反映了一个群组的垃圾程度。

根据应用背景，有很多方法来定义垃圾评分函数。通常，将的垃圾分数定义为所有垃圾指标的平均值是比较安全的。

**定义5**  ***垃圾用户群网络***：给定垃圾用户群，其垃圾用户群网络是一个加权图，其中是边集。给定评论者，边的权重由表示，被定义为：

 (2)

其中，

 (3)

其中和分别是由和评论的产品集。

如果两个评论者评论的产品中的大多数是共同的，更可能发生勾结，故在公式3中引入了Jaccard相似系数。公式2中的利用Sigmoid函数将进行了归一化，我们在图1中绘制了函数。



Fig.1 The Sigmoid function

**定义6**  ***连通的垃圾用户群***：给定垃圾用户群，如果的垃圾用户群网络是连通图，我们称是连通的垃圾用户群。

因为相互勾结，所以垃圾用户群的成员之间的关系要比普通用户之间的关系更加密切。连通图是一种任意两个结点都是连通的图，所以图内各点的关系非常密切，这里我们采用连通图来描述垃圾用户群能够体现出垃圾用户群成员间的异常密切的关系。

# 4.连通的垃圾用户群检测

本节中我们采用社区发现方法来检测垃圾用户群。社区的定义如下：同一社区内的节点与节点之间的连接很紧密，而社区与社区之间的连接比较稀疏。[26]这种紧密的结构和垃圾用户群之间的关系很相似，并且已经有研究表明评论用户之间存在隐式社区[16]，所以，我们有理由相信，通过社交网络社区检测技术可以找到评论网络中个体之间关系紧密的子图，这样的子图可能是具有强烈可疑性的垃圾用户群。

现存的重叠社区发现算法可以分为四大类：派系过滤算法、局部社区发现算法、模糊重叠社区发现算法和边社区发现算法。派系过滤算法[27]基于k极大团来进行重叠社区发现。所谓的k极大团就是由k个节点构成的连通图，这样的k极大团关系非常密切，因此该算法可以识别高度密集的区域，这样高度密集的区域很有可能是具有强烈可疑性的垃圾用户群。

基于上述设想，本文所采用的检测模型如图2所示。首先，将评论数据集建模为加权的评论关系图，作为输入。第一步，删去相似度阈值较小的连边，并且把加权图转换为无权图。第二步，找到网络中的所有团，构造一个团团重叠矩阵，矩阵是对称的，矩阵的第行第列表示第个团和第个团的公共节点数。第三步，给定参数，将社团团重叠矩阵中非对角线上元素小于，且对角线上元素小于的所有项置0，其他的元素为1。第四步，利用第三步得到的矩阵可以得到相应的社区，所有对角线为1的团为团，而非对角线为1的团、团是相邻的。第五步，计算各社区的垃圾分数。第六步，将垃圾分数过小，或者规模较大的社区分为两部分，这两部分利用本方法递归地计算。最后输出所有垃圾用户群的可疑度排名。



Fig.2 The implementation step of Improved Clique Percolation Method

在本节中，我们描述了构建评论者二分图的方法，本方法所使用的垃圾指标，以及垃圾用户群检测算法。

## 4.1.评论数据的二分图模型

在二分图中，图形的节点被分成两组，分别表示和，边将一组的一个成员连接到另一个组的一个成员。二分图广泛用于表示两种对象之间的关系，例如用户及其兴趣。

在本方法中，二分图的两组节点对应于评论者集和产品集，边对应于评论集。如果评论者评论了一个产品，那么该评论者和该产品之间就有一条连边。图3a显示了由4个评论者和3个产品组成的二分图。我们可以看到评论者和评论了所有目标产品，而评论者和只评论了产品和。



图3 评论数据的二分图

为了表示一组二分图中的对象之间的关系，单模投影(one-mode projection)通常被使用在二分图中。单模投影（也称为二分图）构造一个新的图其中（对于投影），对于任何，当且仅当和在原始二分图中至少具有一个公共相邻节点时，边。在加权二分投影图中，边权重表示两个对象之间的连接强度。图3b是图3a中的评论二分图的评论者投影，此图中边权重表示两个节点的公共节点的数量。

在意见垃圾用户群检测问题中，根据上述方法构建评论者节点投影，边权重的计算方法在第3部分的定义5中已经介绍。

## 4.2.垃圾指标

设计有效的垃圾指标以衡量垃圾用户群的可疑性是非常重要的。以前的研究人员提出了许多垃圾指标来评估个人或群体垃圾用户的垃圾程度[11,12,13,28]。这些指标包括语言指标，行为指标，联合指标等。Xu等[19]总结了许多最先进的垃圾指标，并分析了这些指标在一个受欢迎的中国电子商务网站上的有效性。根据他们的研究，共谋者通常采取与普通评论者相似的方式表达，或者在语言层面上表现正常，所以语言指标在辨别垃圾用户/正常用户时往往表现不佳。因此，本研究不考虑语言指标。

我们将本方法中涉及的垃圾指标分为两类，一类为个人垃圾指标，另一类为群组垃圾指标。一个垃圾用户群的垃圾程度分数最终由所有的垃圾指标共同决定。

为了减少小型垃圾用户群的偶然性，[12]设置了一个惩罚函数，在本方法中我们也使用了该惩罚函数，

 (4)

因为评论者的最小人数为2，产品的最小数为1，所以。

### 4.2.1.群组垃圾指标

**1.评论紧密度(Review tightness, RT)**：对于给定的垃圾用户群，由表示的评论紧密度，被定义为中的评论数与中的评论者集和产品集的笛卡尔积的基数之比。但是，具有较少成员和目标产品的群体更有可能是巧合。例如，如果两位用户评论了一个受欢迎的产品，那么它更有可能是意外，而不是因为相互勾结。相反，如果大量的人对几个产品写了很多评论，那就强烈地表明这是一个垃圾活动，所以，我们利用惩罚函数来减少小规模群组的影响。

 (5)

**2.相邻紧密度(Neighbor Tightness, NT)**：群组的相邻紧密度指标考虑到成对的评论者间的紧密度。我们使用两位评论员评论的产品集的Jaccard相似度来衡量两位评论员的串通行为，然后取所有评论者对的平均值。

 (6)

**3.产品紧密度(Product Tightness, PT)**： 对于紧密的垃圾用户群，许多小组成员专注于一定数量的产品，如果这些评论人员不评论任何其他产品，他们最有可能是垃圾用户群。给定一个垃圾用户群，的产品紧密度被定义为中所有成员共同评论的产品数量与中所有成员评论的产品总数的比例：

 (7)

**4.评级偏差(Group Rating Deviation, RD)**：因为其相同的目的，垃圾用户群内各成员对于某一特定产品的评分应该是相同或者相似的。令为中的评论者对产品的评分的方差，我们取所有目标产品的平均方差，并使用逻辑函数将其限制在范围[0.5,1]。方差越大，垃圾程度越小。我们将整个公式乘上一个常数值2，于是这个指标的范围扩展为了（0,1）。最后，我们还使用惩罚函数来减少小规模群组的影响：

 (8)

**5.群组规模(Group Size, GS)**：群组勾结也由其规模展现。对于大群体来说，成员偶然在一起评论的可能性很小。此外，组越大，损害越大。一个小规模的群组（由2-3个评论者组成）更有可能是意外，而不是故意为之。考虑到这一点我们设计的指标倾向于捕获更大规模的垃圾用户群。我们使用逻辑函数来定义群组规模指示符。

 (9)

由于一个群组中评论者最小数目为2人，GS的范围约为（0.27,1）。

在上述所有群组垃圾指标中，评论紧密度、相邻紧密度、产品紧密度和群组规模属于结构性指标，评级偏差属于行为性指标。

### 4.2.2.个人垃圾指标

**1.时间突发性(Burstiness, BST)**：垃圾用户经常在短时间登录网站进行垃圾活动。给定一个评论者，我们定义评论者活动的时间范围为：

 (10)

其中和是评论者的最新评论日期和第一次评论日期，是用户指定的时间阈值，例如30天。

**2.最大评论数量(Maximum Number of Reviews, MNR)**：在一天之内发表很多评论也表明一个不正常的行为。此指标计算一个评论者一天内的最多评论数，并根据所有数据的最大值对其进行归一化。

 (11)

**3.平均评级偏差(Average Rating Deviation, avgRD)**：正常评论者对于某一个产品的评分很大程度上是与同一产品的其他评论者类似的。由于垃圾用户试图提升或诋毁产品，因此他们的评分可能与其他评论者有很大偏差。因此评级偏差是垃圾用户可能表现出来的行为。代表被评论者评论的产品集，代表评论者对产品的评级，代表所有评论者对产品的平均评分，我们将评论者的评分偏差定义如下：

 (12)

评分最高为5分，那么最大的评分偏差为4，利用最大的评分偏差来进行归一化。

以上3个个人垃圾指标都是行为性指标。

## 4.3.垃圾用户群检测算法

本算法的核心就是通过发现k极大团来检测垃圾用户群。所谓的k极大团就是由k个节点构成的连通图，这样的k极大团关系非常密切，很有可能是具有强烈可疑性的垃圾用户群。

算法1(*Group Spammer detection via Improved Clique Percolation Method*)将评论者二分图作为输入，首先初始化评论者二分图中的边权重为0；然后计算每个边权重的值，计算方法参考第3部分定义5；接着删除边权重小于相似度阈值*s*的边；最后调用算法2(*ICPM(G)*)找到连通的垃圾用户群。最终输出垃圾用户群的可疑度排名。

|  |
| --- |
| **Algorithm 1** *Group Spammer detection via Improved Clique Percolation Method* |
| **Input:**  *G*: a bipatite graph representing revewers, reviews and products  *s*: similarity threshold  *k*: The initial group size of Clique Percolation Method  *MINSPAM*: minimum spam threshold for a group  *MAXSIZE*: maximum size of a group  **Output:**  Spammer groups ordered by spamicity score  **Description:**   1. initialize the reviewer relationship graph using reviewer set of *G*, set all weights of edge to 0; 2. **for** each reviewer pair(i, j) in G **do**  5. set the weight of edge(i,j) to 6. **if** ＜s **then** 7. remove edge(i,j) of *G* 8. **end if** 9. **end for** 10. **end for** 11. *gi*, *SS(gi)*=*ICPM(G)* 12. Ranking *SpammerGroups gi* ordered by *SS(gi)* |

算法2(*ICPM(G)*)的输入是一个评论者二分图，首先在图G中执行派系过滤算法*CPM(k)*，得到连通的垃圾用户群，其中*k*值需要人为确定。为了能够检测到较小的垃圾用户群，在实验中我们将*k*设置为2；然后，对于每个垃圾用户群*gi*，计算其垃圾程度分数*SS(gi)*，如果垃圾用户群的垃圾程度分数*SS(gi)*小于设置的最小垃圾阈值*MINSPAM*，或者垃圾用户群规模大于设置的最大规模*MAXSIZE*，那么采用最小切割算法将此垃圾用户群分为*n1*,*n2*两部分，最后*n1*,*n2*由算法2递归地计算。

|  |
| --- |
| **Algorithm 2** *ICPM(G)* |
| **Output:**  *SpammerGroups gi* and it’s *SS(g)*  **Description:**   1. excute *CPM(k)* on G, get *SpammerGroups* 2. **for each** *gi* in*SpammerGroups* **do** 3. compute *SS(gi)* for *gi* 4. **if** *SS(gi)＜MINSPAM* or *gi*.size＞*MAXSIZE* **then** 5. *n1*,*n2*=MinCut(*gi*) 6. *ICPM(n1)* 7. *ICPM(n2)* 8. **else** 9. Output *gi*, *SS(gi)* 10. **end if** 11. **end for** |

算法1的最后一步是排名检测到的垃圾用户群，这一步的作用是通过分析不同的垃圾指标改善算法的精度。一个比较方便的方法是通过计算各垃圾指标的均值来排名检测到的垃圾用户群。可是不同的垃圾指标拥有检测垃圾用户群的不同能力，我们可以通过线性回归来确定每个垃圾指标的权重。线性回归是一种监督模型，需要标记了垃圾用户群的数据，标记大型数据十分的耗时，而且人工标记的准确率一般不高。所以，综合考虑，本方法通过计算各垃圾指标的均值来排名检测到的垃圾用户群。

本方法采用CPM算法搜索评论者二分图中极大团，其时间复杂度为，其中和为常数。然后，的极大团必须分成两个更小的团，以进一步调查其垃圾程度。这里我们采用最小切割算法，通过移除一些边来拆分极大团，使得它们的权重之和最小。通过使用优先级队列[29]，最小切割算法的时间复杂度为，其中和分别是图的顶点集和边集。幸运的是，在真实世界评论数据集中，大多数极大团拥有少于个节点。所以，最终的运行时间受到图中边数的影响最大。本方法中，我们通过指定一个足够大的*s*（相似度阈值）来控制边数。

# 5.实验研究

## 5.1.数据集

在本次研究中，我们使用了来自美国最大的商业评论网站Yelp的数据集，表1显示了这个数据集的统计情况。这个数据集由[2]收集，数据集包含芝加哥地区餐馆和酒店的评论。

表1 本研究中使用的评论数据集

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Domain | fake | Non-fake | %fake | total#reviews | #reviewers |
| Hotel | 802 | 4876 | 14.1% | 5678 | 5124 |
| Restaurant | 8368 | 50149 | 14.3% | 58517 | 35593 |

Yelp从2005年开始致力于假评论的过滤工作，其拥有一个可以辨别虚假/可疑评论的过滤算法，其算法成熟，准确率高，获得了业界研究的认可[30]。由于商业秘密性，Yelp.com并没有公布其算法的细节，但是公布了其对评论的过滤结果。尽管Yelp自带的过滤算法并不完美，但是基于Yelp.com过滤算法的业界认可度和多年开发的成熟性，该数据集是目前在商业真实性与ground truth性之间，做出较好平衡的一个数据集。该数据集中包括推荐评论和可疑评论，我们把他们分别视为真实评论和虚假评论。我们还把评论者分为两类：垃圾用户（假评论用户）和正常用户（无假评论用户）。

由于该数据集中包含许多空值，为了实验的正确性，我们删除了评论者ID为空或者产品ID为空的值。对于评论ID的空值，我们认为它不影响计算结果，所以我们保留了评论ID为空值的记录。

## 5.2.垃圾邮件指标的实验分析

本文中使用的所有群组垃圾指标都选自[13]，并且[13]已经用Yelp数据集验证了指标的有效性，所以在本文中对于群组垃圾邮件指标的实验分析略。使用带标记的Yelp数据集，我们可以对我们提供的3组个人垃圾指标进行实验分析。累积分布函数（CDF）用于显示垃圾指标的辨别能力。图4绘制了3组个人垃圾指标的CDF，即垃圾用户（红色实线）和非垃圾用户（蓝色虚线）的垃圾指标的累积百分比。两个分布曲线之间的差距越大，指标将达到更好的识别能力。我们可以看到，所有这3个指标都可以在一定程度上区分垃圾用户和正常用户行为。



Fig.4 CDF distribution of individual spam indicators for spam vs. nonSpam reviewers.

## 5.3.比较基线

我们把FraudEagle[31]，Wang等人[17]提出的基于图形的方法，以及GSBP[12]作为基线，进行性能比较。FraudEagle采用了基于置信度传播(MRF)的推理算法，该算法仅利用评论者和产品之间的关系结构对评论者的可疑度进行排名。通过在高度可疑的评论者-产品的诱导子图上进行图形聚类，获得垃圾用户群。Wang等提出了一个异构评论图的新概念来搜寻评论者，评论和商店之间的关系。值得注意的是，FraudEagle和Wang的方法没有生成垃圾用户群，他们仅仅将评论和评论者进行了排名。GSBP是检测垃圾用户群的方法。

## 5.4.参数影响

本方法中有许多自定义参数，它们将会影响本方法的结果。这里我们调查这些参数在Yelp数据集上的影响。

参数用来过滤两个评论之间可疑度比较小的值。我们设置为0.26,0.28,0.30，控制。对于不同的值，图5展示了其对排名前100至1000的评论者的精确度和生成的垃圾用户群的数量的影响。从图中我们可以看出，随着值的减小，会产生更多的垃圾用户群，但是精确度却越来越小。



Fig.5 The impact of different *s* on precision for top-ranked reviewers, Num. of groups and Jaccard similarity for top-ranked reviewers

同样，我们研究参数的影响，控制，设置为10，15，20。从图6中我们可以看出随着的增加，更多的垃圾用户群被检测到了，但是精确度却逐渐减小。



Fig.6 The impact of different on precision for top-ranked reviewers, Num. of groups and Jaccard similarity for top-ranked reviewers

图7研究参数*MINSPAM*的影响，我们控制，设置*MINSPAM*为0.56,0.50,0.44。从图中我们可以看出，一方面，随着*MINSPAM*的增加，垃圾用户群的数量显著下降，所以该参数对于垃圾用户群数量的影响最大。另一方面，较大的*MINSPAM*值将会带来更高的检测精度。



Fig.7 The impact of different *MINSPAM* on precision for top-ranked reviewers, Num. of groups and Jaccard similarity for top-ranked reviewers

综合考虑，我们使用参数设置作为基线，进行对比实验。

## 5.5.在数据集上的性能比较

我们使用[12]中提到的最佳设置来设置GSBP的参数。通过设置本方法的参数为，使这两种方法生成数量差不多相同的500个垃圾用户群。为了公平，我们从这两个方法中抽取了他们各自检测出来的排名前500的垃圾用户群。

由于Yelp数据集是标记了的数据集，因此Yelp数据集中的每一个评论都被标记为假评论（过滤的）和真实评论（推荐的），我们通过检查检测到的垃圾用户群里的每一个评论（或评论者），以此来评估本方法的精确度。一个评论者是垃圾用户当且仅当他/她发布了至少一个假评论。因此我们可以根据检测到的500个垃圾用户群去检查排名靠前的评论（评论者）。[22]中公开了利用Yelp数据集，FraudEagle和Wang的方法检测到的排名前1000的评论（或评论者）的精确度，利用这些数据，我们可以将本方法的性能与其进行比较。图8展示了在Yelp数据集上，本方法和其他三个基线检测出的排名靠前的评论（者）的评论精度和评论者精度。我们可以看到，当评论（者）增大时，本方法和GSBP的精度在减小，而FraudEagle和Wang的方法的走势有点奇怪。总的来说，本方法优其他3个基线。



Fig.8 Precision comparison for top-ranked review(er)s on Yelp datasets

图9展示了在Yelp数据集上，排名前k群组的精确度和F1值。排名前k群组的精确度定义为所有前k个群组的假评论数和前k个群组包含的总评论数的比值。F1值同时考虑评论的精确度和召回率。我们可以看到，在Yelp数据集上，本方法的精确度和F1值都优于GSBP，并且前100个垃圾用户群的检测精确度远远高于GSBP，这表明本方法可以比GSBP生成更加高质量的垃圾用户群。



Fig.9 Review precision and F1@ top k groups on Yelp datasets

# 6.结论及未来工作

本文利用评论者二分图的拓扑结构，提出了一种改进的派系过滤算法用以检测评论关系图中的k极大团，以此检测出意见垃圾用户群。给定一个评论者二分图，本方法识别其垃圾分数超过给定的垃圾阈值的所有极大团。对于大型不可疑的极大团，利用最小切割算法分为两个子图，进一步利用本方法进行递归处理。实验研究表明，我们提出的方法优于几个最先进的基线。

随着网络的普及和数据的爆炸，网络中的消费者评论会越来越多。现存的检测垃圾评论的研究都是基于静态数据集的，垃圾用户发表垃圾评论到检测出垃圾用户会经历一个漫长的过程，这对于不良商家来说非常有利。因为如果垃圾用户在产品发布初期就对目标产品发布大量的虚假评论，这将直接引导消费者的情绪。但遗憾的是，目前还没有学者对垃圾评论或垃圾用户的实时标注进行研究，我们相信这将会是未来的一个研究方向。

# 参考文献

1. Luca M. Reviews, reputation, and revenue: The case of Yelp. com[J]. 2016.
2. Mukherjee A, Venkataraman V, Liu Bing, et al. What yelp fake review filter might be doing?[C]//ICWSM. 2013.
3. Jindal N, Liu Bing. Opinion spam and analysis[C]//Proceedings of the 2008 International Conference on Web Search and Data Mining. ACM, 2008: 219-230.
4. Yoo K H, Gretzel U. Comparison of deceptive and truthful travel reviews[J]. Information and communication technologies in tourism 2009, 2009: 37-47.
5. Song Feng, Xing Longfei, Gogar A, et al. Distributional Footprints of Deceptive Product Reviews[J]. ICWSM, 2012, 12: 98-105.
6. Jindal N, Liu Bing, Lim E P. Finding unusual review patterns using unexpected rules[C]//Proceedings of the 19th ACM international conference on Information and knowledge management. ACM, 2010: 1549-1552.
7. Lim E P, Nguyen V A, Jindal N, et al. Detecting product review spammers using rating behaviors[C]//Proceedings of the 19th ACM international conference on Information and knowledge management. ACM, 2010: 939-948.
8. Ott M, Choi Y, Cardie C, et al. Finding deceptive opinion spam by any stretch of the imagination[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, 2011: 309-319.
9. Fei Geli, Mukherjee A, Liu Bing, et al. Exploiting Burstiness in Reviews for Review Spammer Detection[J]. ICWSM, 2013, 13: 175-184.
10. Mukherjee A, Liu Bing, Wang Junhui, et al. Detecting group review spam[C]//Proceedings of the 20th international conference companion on World wide web. ACM, 2011: 93-94.
11. Mukherjee A, Liu Bing, Glance N. Spotting fake reviewer groups in consumer reviews[C]//Proceedings of the 21st international conference on World Wide Web. ACM, 2012: 191-200.
12. Wang Zhuo, Hou Tingting, Song Dawei, et al. Detecting Review Spammer Groups via Bipartite Graph Projection[J]. Computer Journal, 2016, 59(6): 861-874.
13. Wang Zhuo, Gu Songmin, Zhao Xiangnan, et al. Graph-based review spammer group detection[J]. Knowledge and Information Systems, 2017: 1-27.
14. Do Q N T, Zhilin A, Junior C Z P, et al. A network-based approach to detect spammer groups[C]//Neural Networks (IJCNN), 2016 International Joint Conference on. IEEE, 2016: 3642-3648.
15. Wang C C, Day M Y, Lin Y R. A Real Case Analytics on Social Network of Opinion Spammers[C]//Information Reuse and Integration (IRI), 2016 IEEE 17th International Conference on. IEEE, 2016: 623-630.
16. Choo E, Yu Ting, Chi Min, et al. Revealing and incorporating implicit communities to improve recommender systems[C]//Proceedings of the fifteenth ACM conference on Economics and computation. ACM, 2014: 489-506.
17. Wang Guan, Xie Sihong, Liu Bing, et al. Review graph based online store review spammer detection[C]//Data mining (icdm), 2011 ieee 11th international conference on. IEEE, 2011: 1242-1247.
18. Mukherjee A, Kumar A, Liu Bing, et al. Spotting opinion spammers using behavioral footprints[C]//Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013: 632-640.
19. Xu Chang, Zhang Jie, Chang Kuiyu, et al. Uncovering collusive spammers in chinese review websites[C]//Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. ACM, 2013: 979-988.
20. Xu Chang, Zhang Jie. Towards collusive fraud detection in online reviews[C]//Data Mining (ICDM), 2015 IEEE International Conference on. IEEE, 2015: 1051-1056.
21. Ye Junting, Akoglu L. Discovering opinion spammer groups by network footprints[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2015: 267-282.
22. Choo E, Yu Ting, Chi Min. Detecting opinion spammer groups through community discovery and sentiment analysis[C]//IFIP Annual Conference on Data and Applications Security and Privacy. Springer, Cham, 2015: 170-187.
23. Duh A, Stiglic G, Korosak D. Enhancing identification of opinion spammer groups[C]//Proceedings of international conference on making sense of converging media. ACM, 2013: 326.
24. Gera T, Thakur D, Singh J. BILD Testing for Spotting Out Suspicious Reviews, Suspicious Reviewers and Group Spammers[C]//Communication Systems and Network Technologies (CSNT), 2015 Fifth International Conference on. IEEE, 2015: 976-981.
25. Yang Ming, Lu Ziyu, Chen Xiaojun, et al. Detecting Review Spammer Groups[C]//AAAI. 2017: 5011-5012.
26. Newman M E J, Girvan M. Finding and evaluating community structure in networks[J]. Physical review E, 2004, 69(2): 026113.
27. Palla G, Derényi I, Farkas I, et al. Uncovering the overlapping community structure of complex networks in nature and society[J]. nature, 2005, 435(7043): 814.
28. Rayana S, Akoglu L. Collective opinion spam detection: Bridging review networks and metadata[C]//Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining. ACM, 2015: 985-994.
29. Stoer M, Wagner F. A simple min-cut algorithm[J]. Journal of the ACM (JACM), 1997, 44(4):585-591.
30. Weise K. A lie detector test for online reviewers[J]. Bloomberg Business Week, 2011.
31. Akoglu L, Chandy R, Faloutsos C. Opinion Fraud Detection in Online Reviews by Network Effects[J]. ICWSM, 2013, 13: 2-11.