软件学报 ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*,2019,30(10):3168−3185 [doi: 10.13328/j.cnki.jos.005794]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

# 体现使用反馈的 APP 软件用户评论挖掘*

胡甜媛[1,2], 姜 瑛[1,2]

[1](云南省计算机技术应用重点实验室,云南 昆明 650500)

[2](昆明理工大学 信息工程与自动化学院,云南 昆明 650500)

通讯作者: 姜瑛, E-mail: jy_910@163.com

**摘 要**: 随着 APP 软件应用的普及,针对 APP 软件的用户评论数量急剧增加,基于用户评论挖掘有价值的软件使用反馈,可以帮助开发人员有针对性地维护和改进 APP 软件.针对不同类型的 APP 软件使用反馈,提出了评价对象和评价观点抽取规则,给出了评论模式和评论种子的定义;应用评论种子挖掘与之相同或相似的体现相同使用反馈类型的用户评论,基于人工标注的少量初始评论种子持续构建候选评论模式库;应用半监督自学习的方式,基于候选评论模式库动态扩充评论种子库;通过循环挖掘的方式,动态扩大挖掘体现不同使用反馈类型的 APP 软件用户评论的范围.实验结果表明,所提方法可以有效地挖掘体现使用反馈的 APP 软件用户评论,平均挖掘率达到 77.82%.

**关键词**: APP 软件;用户评论;使用反馈;评论模式;评论种子;抽取规则;循环挖掘

**中图法分类号**: TP311

中文引用格式: 胡甜媛,姜瑛.体现使用反馈的 APP 软件用户评论挖掘.软件学报,2019,30(10):3168−3185. http://www.jos.org.cn/1000-9825/5794.htm

英文引用格式: Hu TY, Jiang Y. Mining of user’s comments reflecting usage feedback for APP software. Ruan Jian Xue Bao/Journal of Software, 2019,30(10):3168−3185 (in Chinese). http://www.jos.org.cn/1000-9825/5794.htm

## Mining of User’s Comments Reflecting Usage Feedback for APP Software

HU Tian-Yuan[1,2], JIANG Ying[1,2]

[1](Yunnan Key Laboratory of Computer Technology Application, Kunming 650500, China)

[2](Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China)

**Abstract**: With the popularity of App software applications, the number of user’s comments for App software has increased dramatically. Mining valuable software usage feedback based on user’s comments can help developers to maintain and improve App software pertinently. Aimed at different types of usage feedback for App software, this study proposes the extracting rules of evaluation object and evaluation opinion. Moreover, the comment modes and comment seeds are defined. User’s comments that are same or similar to comment seeds reflecting usage feedback are mined. Based on the initial comment seeds labeled manually, a candidate comment mode library is built continuously. A semi-supervised learning method is used to dynamically expand the comment seed library based on the candidate comment mode library. The scope of mining user’s comments reflecting usage feedback is expanded by interactive mining process. Finally, the experimental results show that the proposed method can effectively mine App software user’s comments reflecting usage feedback with an average mining rate of 77.82%.

**Key words**: App software; user’s comments; usage feedback; comment mode; comment seed; extracting rule; interactive mining

---

* 基金项目: 国家自然科学基金(61462049, 60703116, 61063006); 云南省应用基础研究计划(2017FA033); 云南省教育厅科学研究基金(2018Y016)

Foundation item: National Natural Science Foundation of China (61462049, 60703116, 61063006); Key Project of Yunnan Applied Basic Research (2017FA033); Scientific Research Fund Project of the Yunnan Education Department (2018Y016)

本文由“面向 DevOps 的软件工程新技术”专题特约编辑荣国平、白晓颖、岳涛推荐.

收稿时间: 2018-09-01; 修改时间: 2018-10-31; 采用时间: 2018-12-14; jos 在线出版时间: 2019-05-22

CNKI 网络优先出版: 2019-05-22 15:25:54, http://kns.cnki.net/kcms/detail/11.2560.TP.20190522.1525.003.html

© 中国科学院软件研究所 http://www.jos.org.cn

随着智能终端的广泛应用,APP 软件发展迅速.APP 软件的特点是开发周期短,更新速度快,同类型产品竞争激烈.在日益激烈的市场竞争中,用户驱动的软件演化更具有实际意义.Panago 等人[1]指出,由于软件开发者和用户互不认识,用户使用反馈,如建议意见和系统异常等对开发者更加重要.因此,获取有价值的软件使用反馈可以让用户参与到软件的设计和维护中,是软件开发商在竞争激烈的软件市场取得更好收益的重要保证.Jiang 等人[2]认为,APP 软件的在线用户评论凭借覆盖用户广泛、内容丰富、时效性强等优势,成为软件使用反馈获取的重要资源.Lu 等人[3]认为,针对产品的在线用户评论可以及时地体现用户关注的产品属性、同类型其他产品的优势与自身产品的劣势以及下一版本需要改进的地方.因此,基于 APP 软件的用户评论挖掘可以帮助获取有价值的软件使用反馈,分析用户在使用软件过程中的使用感受及用户所关注的软件特征,帮助软件开发者站在用户的视角有针对性地维护和改进软件,最终辅助软件开发者提高软件产品的竞争力.此外,挖掘体现使用反馈的用户评论有助于软件应用市场综合分析软件的实际使用情况,从而给潜在用户提供有效的参考.

然而,APP 软件用户群体广泛,用户评论内容多样.海量的用户评论中包含大量与 APP 软件使用反馈不相关的用户评论信息,例如,与所评价的 APP 软件无关的用户评论、未体现 APP 软件实际使用情况或用户使用观点的用户评论等.针对用户评论数据规模大、更新快等特点,如何过滤大规模用户评论数据中未体现使用反馈的用户评论,准确、高效地获取有建设性的 APP 软件用户的使用反馈是当前亟需解决的问题.

## 1 相关工作

针对挖掘用户评论中有价值的使用反馈,国内外学者开展了相关研究.Cui 等人[4]提出基于评论挖掘的需求获取方法 RERM(software requirement elicitation method on review mining),通过采用本体和条件随机场模型融合的特征提取方法,结合情感分析技术,对软件存在的相关问题分类汇总,如改进特征、建议意见和缺陷修改等.为了挖掘用户评论中体现软件存在的问题的使用反馈,Panichella 等人[5]应用一种自然语言解析器(stanford typed dependencies parser)进行特征抽取,通过对依赖关系的分析,检测评论文本结构,分析特定的关键字所对应的精确语法角色和特定语法结构,从而判断用户评论是否与软件缺陷或用户功能请求等方面的使用反馈相关.Grano 等人[6]提供了 APP 软件的用户反馈类型的概述,并记录了相关代码度量标准的演变.

部分研究通过分类、聚类来挖掘用户评论.Guzman 等人[7]为了识别用户评论中体现使用反馈的细粒度的功能特征,利用主题建模技术对细粒度功能进行分组,将其转化为更有意义的高级特征.Keertipati 等人[8] 为了挖掘用户评论中体现的软件需要改进的特征,应用文献[7]中主题模型的方法抽取用户评论中的相关特征,站在 APP 软件开发者的角度,对挖掘出的软件使用过程中存在的问题或用户期望进行优先级排序.Gao 等人[9]提出的主题排序和评论排序计划可以对用户评论进行优先级排序,从而帮助开发者应用最合适的主题模型发现软件使用过程中存在的最新问题.Palomba 等人[10]围绕相似的体现软件未达到期望的用户评论,通过主题模型 LDA(latent Dirichlet allocation)等 3 种聚类技术对评论中的用户请求进行分组.Guzman 等人[11]应用多个分类器对用户评论进行更详细的种类划分,将 APP 软件用户评论划分为 7 个类别:缺陷报告、功能优点、功能缺点、用户请求、表扬、抱怨和使用场景,不同类型的用户评论可以反馈出用户对软件的满意程度或软件存在的问题等.Maalej 等人[12]采用文本分类、自然语言处理、多种情感分析等技术将用户评论分为问题报告、功能请求、用户体验和评分 4 种类型.Villarroel 等人[13]设计了一种分类、聚类的方法——CLAP(crowd listener for release planning),对体现软件缺陷、特征请求等用户评论进行分类,同时针对体现相似软件缺陷的用户评论进行聚类.Gu 等人[14]根据预定义的句式结构直接抽取用户评论句子中的对象-观点对,通过对象-观点进行相同方面的聚类来总结用户评论中体现的使用反馈.

通过分析现有研究我们发现:(1) 在经过预处理后的用户评论数据中,仍然包含大量无用数据,应用分类技术挖掘用户评论,大量无关的用户评论数据会影响有价值的使用反馈数据的挖掘;(2) 基于已有的知识库的方式只能匹配知识库中已有的内容,无法识别知识库外有价值的信息;(3) 应用有限的软件特征词挖掘体现使用反馈的用户评论,忽略了句式结构的重要性;已有针对用户评论进行使用反馈挖掘的研究对象大部分为用户评论内容,而忽略了用户评论的句式结构在用户表达使用反馈时的规律.此外,由于在线网络环境下,用户评论数

© 中国科学院软件研究所 http://www.jos.org.cn

量大、更新快且类型丰富,部分研究采用有监督的学习方式,需要建立完备的数据资源,且随着时间的推移,人工标注的数据资源需要不断补充甚至重新构建,难以满足挖掘需求.如何综合分析 APP 软件用户评论的评论内容和评论句式结构的特点,通过半监督自学习的方式,自动挖掘出体现不同类型的使用反馈的 APP 软件用户评论是本文的研究重点.

## 2 APP 软件使用反馈的类型

由于软件的维护和演化应该严格地由用户评论中包含的用户请求来引导,所以软件用户的使用反馈分析尤其重要[15].通过分析用户针对 APP 软件发表的不同类型的使用反馈,可以帮助软件开发者获得有价值的软件信息,发现用户主要关注的软件特征和常见的软件存在的问题.文献[16]认为,从用户的使用反馈中获取用户需求的主要任务是从海量、非结构化、有噪声、不确定的评论中抽取用户共同关注的软件特征及其群体观点.在挖掘体现使用反馈的用户评论过程中,本文认为,用户所关注的软件特征即评价对象,用户所表达的观点即评价观点.评价对象就是观点持有者表达情感的目标实体,通常由一个或多个单词组成[17].评价观点指的是能够表达用户自身观点的带有情感倾向的词语,是判断用户对评价对象情感的根本依据[18].因此,用户针对不同的评价对象发表不同的观点,对评价观点进行归类可以挖掘出不同类型的用户使用反馈.通过分析大量体现使用反馈的 APP 软件用户评论后发现,有价值的 APP 软件使用反馈往往针对该 APP 软件的实际使用情况发表满意或者不满意的观点,不满意的用户评论观点一般体现出软件在实际使用过程中存在的问题.此外,用户评论体现的建议或意见对于 APP 软件的维护和改进也具有较大的参考价值.因此,本文将用户评论体现的 APP 软件使用反馈总结为以下 3 种类型.

(1) 软件满足的需求:描述了用户在实际使用软件的过程中,软件的某些特征或软件整体使用户具有较好的使用体验.

(2) 软件存在的问题:描述了用户在实际使用软件的过程中,软件存在的问题,例如崩溃、错误或性能问题.

(3) 软件未达到的期望:表达了用户希望达到的功能(例如其他同类型软件提供的功能),分享如何通过添加或更改特性来改进未来版本软件的想法.

针对上述 3 种不同类型的 APP 软件使用反馈,本文进行用户评论的挖掘,主要针对一条用户评论中的评价对象和评价观点进行分析,排除用户评论中大量无关信息的干扰,旨在以评价对象和评价观点作为一条用户评论的核心内容,从海量用户评论中挖掘出体现不同使用反馈类型的用户评论.

## 3 评论种子及抽取规则

针对 APP 软件的使用反馈进行用户评论挖掘,可以通过分析评价对象和评价观点判断用户评论是否体现了某种类型的使用反馈.此外,通过分析大量用户评论,本文发现,针对体现不同类型的使用反馈,特定的句式结构可以辅助挖掘体现不同使用反馈类型的用户评论.例如:“安装/v 不/d 了/y”和“登/v 不/d 上/vf”(其中,v 表示动词、d 表示副词、y 表示语气词、vf 表示趋向动词),其评价对象不同,分别为“安装/v”和“登/v”,但两者句式结构相似,均为“v+d(否定副词)”,该句式结构特点为:动词 v 表示评价对象,否定副词 d 表示评价观点,且该句式结构均体现“软件存在的问题”这一使用反馈类型.由此可以看出,评价对象、评价观点、句式结构这 3 个方面在挖掘用户评论时都至关重要.由于词体现评价内容,词性体现句式结构,所以综合分析词和词性是挖掘体现使用反馈的用户评论的重要手段.

本节定义评价对象和评价观点的抽取规则,通过该抽取规则分析一条用户评论的核心内容;结合 APP 软件使用反馈类型定义了评论种子,通过定义评论种子,综合应用一条用户评论的词和词性可以挖掘与之相同或相似的体现相同使用反馈类型的用户评论;利用评价对象和评价观点的抽取规则辅助分析评论种子的评价对象和评价观点,抽取新的评论种子,从而获得评论种子的核心表达内容、排除无关内容的干扰.

© 中国科学院软件研究所 http://www.jos.org.cn

### 3.1 评价对象和评价观点的抽取规则

为了判断用户评论中的核心内容是否体现使用反馈,需抽取用户评论中的评价对象和评价观点.

评价对象多为名词或者名词短语[19].此外,由于本文针对体现使用反馈的用户评论进行挖掘,通过人工分析大量体现使用反馈的用户评论的评价对象发现,存在用户针对软件功能性的行为动作进行评价的情况,如评价对象为“下载/v”和“退出/v 播放/v”等.因此,本文主要针对名词、动词及其组合形式建立评价对象词性组合规则以抽取评价对象,抽取规则见表 1.

**Table 1** Extracting rules of evaluation object

**表 1** 评价对象抽取规则

| 评价对象类型 | 词性组合 | 优先级 | 评价对象 | 用户评论示例 |
|---|---|---|---|---|
| 多行为限定对象 | v+v/vi/vn+n | 1 | v+v/vi/vn+n 为评价对象 | 可以/v 抢/v 红包/n 更/d 好/a |
| 组合名词对象 | n+n | 2 | n+n 为评价对象 | QQ/n 浏览器/n 越来越/d 差劲/a |
| 组合行为对象 | v+ v/vi/vn | 3 | v+v/vi/vn 为评价对象 | 不/d 能/v 下载/v 退出/v |
| 主体行为对象 | n+v/vi/vn | 4 | n+v/vi/vn 为评价对象 | 广告/n 植入/v |
| 行为限定式对象 | v/vi/vn+n | 5 | v/vi/vn+n 为评价对象 | 推荐/v 内容/n 好/a |
| 前置评价对象 | n+a | 6 | 前置 n 被后置 a 修饰,n 为前置评价对象 | 内存/n 很小/a |
| 后置评价对象 | a+n | 7 | 后置 n 被前置 a 修饰,n 为后置评价对象 | 傻/a 软件/n |
| 名词独立对象 | n | 8 | n 为评价对象 | 喜欢/vi QQ/n |
| 动词独立对象 | v/vi/vn | 9 | v/vi/vn 为评价对象 | 改版/vi 不/d 好/a; |

其中,v 表示动词、vi 表示不及物动词、vn 表示动名词、a 表示形容词、n 表示名词.

文献[20]指出,形容词或动词可以作为判别句子中情感倾向的依据,副词作为形容词以及动词的修饰词,起到了增强情感强度的作用,而形容词、动词能够更好地指示其情感倾向.但是,通过人工分析大量体现使用反馈的用户评论的评价对象后发现,否定副词在修饰形容词或动词表示评价观点时起到否定修饰的作用而不是增强情感,如“不/d 喜欢/v”.因此,本文主要针对形容词、动词、否定副词及其组合形式建立评价观点词性规则以抽取评价观点,抽取规则见表 2.

**Table 2** Extracting rules of evaluation opinion

**表 2** 评价观点抽取规则

| 评价观点类型 | 词性组合 | 优先级 | 评价观点 | 用户评论示例 |
|---|---|---|---|---|
| 并列观点 | a+a | 1 | a+a 作为评价观点 | 不错/a 好/a ./wj |
| 否定修饰观点 | d+a | 2 | d 修饰 a,若 d 为否定副词,则 d+a 为评价观点词短语;否则,a 为评价观点 | 不/d 好/a 用/v<br>广告/n 太/d 长/a |
| 后置修饰观点 | n+a | 3 | a 后置修饰 n,a 为评价观点 | 内存/n 小/a |
| 前置修饰观点 | a+n | 4 | a 前置修饰 n,a 为评价观点 | 好/a 东西/n |
| 伴随行为观点 | d+v+b+n | 5 | d+v+b+n 为评价观点 | 越/d 更新/v 越/b 垃圾/n |
| 双重否定修饰行为观点 | d+v+d+v/vi/vg | 6 | d+v+d 修饰 v/vi/vg,若 2 个 d 均为否定副词,则 v/vi/vg 为评价观点 | 不/d 是/v 不/d 能/v |
| 双重否定副词修饰行为观点 | d+d+v/vi/vg | 7 | d+d 修饰 v/vi/vg,若 2 个 d 均为否定副词,则 v/vi/vg 为评价观点 | 没有/d 不/d 喜欢/vi |
| 否定修饰行为观点 | d+v | 8 | d 修饰 v,若 d 为否定副词,则 d+v 为评价观点 | 不/d 限制/v 时间/n 流量/n |
| 否定副词独立观点 | d | 9 | 否定副词 d 修饰无实际意义词,如语气词 y,则 d 为评价观点 | 登/v 不/d 了/y |
| 语素独立观点 | vg/ng/ag | 10 | vg/ng/ag 为评价观点 | 棒/ng 棒/ng 哒/o |
| 形容词独立观点 | a/an/al | 11 | a/an/al 为评价观点 | 好/a;实惠/an |
| 动词独立观点 | v/vi | 12 | v/vi 为评价观点 | 支持/v;喜欢/vi |

其中,an 表示名形词,al 表示形容词性惯用语,vg 表示动词性语素,ag 表示形容词性语素,ng 表示名词性语素,y 表示语气词.

表 2 中“否定修饰观点”涉及否定副词的概念,本文参考文献[21],抽取了如下否定副词:“没有、不、非、匪、弗、否、靡、蔑、莫、末、蒯、微、未、无、毋、勿、别、没、休、白、空、徒、枉”.如表 2 中的用户评论示例“广告/n 太/d 长/a”,该评论包含词性组合“d+a”,其中的 d 为“太”,由于“太”不是否定副词,所以该用户评论

© 中国科学院软件研究所 http://www.jos.org.cn

的评价观点为“长”.

如果一条用户评论满足多条评价对象(评价观点)抽取规则,就会引发冲突,此时需要为抽取规则定义相应的优先级.主要存在以下两种情况.

(1) 用户评论同时满足多条评价对象(评价观点)抽取规则,且每条规则不相关.例如:“棒/ng ,/wd 不过/c 皮肤/n 不/d 够/v”,该用户评论通过“名词独立对象”规则抽取的评价对象为“皮肤”.然而,在抽取评价观点的过程中,该用户评论同时满足“否定修饰行为观点”规则和“语素独立观点”规则.通过“否定修饰行为观点”规则抽取出的评价观点为“不够”,通过“语素独立观点”规则抽取出的评价观点为“棒”.评价观点“不够”修饰评价对象“皮肤”,且该评价观点对分析用户使用反馈更有建设性,因此,“否定修饰行为观点”规则的优先级高于“语素独立观点”规则.

(2) 用户评论同时满足多条评价对象(评价观点)抽取规则,且其中一条规则的词性组合包含另一条规则的词性组合.例如,评论“不/d 好/a”在抽取评价观点的过程中,可能同时满足表 2 中“否定修饰观点”和“形容词独立观点”规则,且“否定修饰观点”规则的词性组合“d+a”包含“形容词独立观点”规则的词性组合“a”,但通过“否定修饰观点”规则抽取出的评价观点才是表达正确的评价观点,因此,“否定修饰观点”规则的优先级高于“形容词独立观点”规则.

针对上述两种评价对象(评价观点)抽取规则的冲突情况,我们通过分析大量体现使用反馈的用户评论数据,总结了不同抽取规则在分析用户评论的评价对象和评价观点时的有效性,对每个抽取规则定义了不同的优先级.优先级体现了抽取规则的优先程度,优先级越高,抽取规则的优先程度越高.

按照表 1 和表 2 中抽取规则中的优先级分别抽取评价对象和评价观点,主要遵循以下原则.

(1) 在抽取评价对象(评价观点)时,若用户评论同时包含多种优先级的评价对象(评价观点)类型,则按优先级最高的评价对象(评价观点)类型的规则进行抽取,不考虑其他规则.

(2) 在抽取评价对象的过程中,若用户评论中同时包含同一优先级的评价对象类型,则根据该评价对象类型的词性组合出现的先后顺序,抽取位置较前的词性组合所对应的评价对象.

(3) 在完成评价对象抽取之后进行评价观点抽取时,若用户评论中同时包含同一优先级的评价观点类型,则根据该评价观点类型的词性组合与已抽取的评价对象的距离,抽取与评价对象距离较近的词性组合所对应的评价观点.

应用本节构建的评价对象和评价观点的抽取规则可以自动抽取一条用户评论中的评价对象和评价观点,通过该方式可挖掘一条用户评论中的核心内容,并通过核心内容分析用户评论是否体现使用反馈及其使用反馈类型.

### 3.2 评论模式和评论种子的定义

为了更准确地挖掘出体现使用反馈的 APP 软件用户评论,针对不同使用反馈类型,本文从综合考虑词和词性的角度出发,结合评价对象和评价观点的抽取规则提出了评论模式和评论种子的概念.首先,为了便于挖掘用户评论,本文将用户评论的内容定义为评论模式.

**定义 1**. 评论模式(mode):用户评论的有代表性的表达方式,包含词、词性、权重和使用反馈类型 4 个属性.评论库中共有 $r$ 条用户评论,对应 $r$ 个评论模式:$mode=\{mode_1,\ldots,mode_m,\ldots,mode_r\}(1\leqslant m\leqslant r)$,$mode_m=\langle word_{m1}+\ldots+word_{mn}+\ldots+word_{mq},speech_{m1}+\ldots+speech_{mn}+\ldots+speech_{mq},weight_{m1}+\ldots+weight_{mn}+\ldots+weight_{mq},feedback_type_m\rangle(1\leqslant n\leqslant q)$.

其中,$word$ 代表词;$speech$ 代表词性;$weight$ 代表词及其词性对应的权重;+代表词/词性/权重的连接;$q$ 代表评论模式中词(词性)的数量;$feedback_type_m$ 代表用户评论 $comment_m$ 体现的使用反馈类型,未对评论 $comment_m$ 进行使用反馈类型判断时,其值为 *unknown*.

针对用户评论表达不规则、数量大、更新快的特点,本文借鉴了种子的相关思想.种子的概念被广泛应用于知识获取的半监督机器学习[22]中,将人工标注的少量语料作为评论种子,并通过半监督自学习的方式从大量未标注的语料中自动迭代扩大评论种子集.由于评价对象、评价观点、句式结构这 3 个方面在挖掘用户评论时

© 中国科学院软件研究所 http://www.jos.org.cn

都至关重要,本文通过词性的方式体现句式结构,通过词和权重的方式体现评价对象和评价观点这两个核心评价内容,并设置使用反馈类型标识评论种子体现的使用反馈类型,具体的评论种子定义如下.

**定义 2**. 评论种子(seed):体现使用反馈的具有代表性的评论表达方式,包含词、词性、权重、距离和使用反馈类型 5 个属性.评论种子库中共有 $s$ 个评论种子.

$seed=\langle seed_1,\ldots,seed_i,\ldots,seed_s\rangle(1\leqslant i\leqslant s)$;

$seed_i=\langle word_{i1}+\ldots+word_{ij}+\ldots+word_{ip},speech_{i1}+\ldots+speech_{ij}+\ldots+speech_{ip},weight_{i1}+\ldots+weight_{ij}+\ldots+weight_{ip},dis_i,feedback_type_i\rangle(1\leqslant j\leqslant p)$.

其中,$word$ 代表词;$speech$ 代表词性;$p$ 代表评论种子中词(词性)的数量;$weight$ 代表词及其词性对应的权重,可体现评价对象和评价观点;+代表词/词性/权重的连接;$dis_i$ 为该评论种子的距离,其值为评价对象和评价观点之间可扩展的最大距离以及评论种子中评价对象和评价观点的词(词性)的数量之和,评价对象和评价观点之间可扩展的最大距离表示评价对象和评价观点之间可能出现不影响评论表达含义的其他词的最大数量;$feedback_type_i$ 表示该评论种子体现的使用反馈类型.应用第 3.1 节中的评价对象和评价观点抽取规则,抽取评论种子中的评价对象和评价观点,评论种子的词及其词性对应的权重 $weight$ 的计算如公式(1)所示:

$$weight=\begin{cases}\dfrac{1}{numObjectOpinion}, & word_{ij},speech_{ij}\text{为评价对象或评价观点}\\ 0, & word_{ij},speech_{ij}\text{不为评价对象或评价观点}\end{cases}\tag{1}$$

其中,$numObjectOpinion$ 代表该评论种子的评价对象和评价观点的总个数.例如,评论种子“〈内存+太+大,n+d+a,0.5+0+0.5,6,软件存在的问题〉”,应用“独立名词对象”的词性组合规则抽取出该评论种子的评价对象为“内存”,应用“否定修饰观点”的词性组合规则抽取出该评论种子的评价观点为“大”.该评论种子的评价对象和评价观点的词(词性)的数量为 2,针对该评论种子的表达方式,评价对象和评价观点之间一般由副词修饰评价观点,且副词数量最大为 4[23],因此,该评论种子的距离为评价对象和评价观点之间可扩展的最大距离与评论种子中评价对象和评价观点的词(词性)的数量之和,即 4+2=6.

基于评论模式和评论种子的定义,可以结合评价对象和评价观点的抽取规则设置评论种子的词及其词性的权重,通过权重的方式反映出该评论种子中体现使用反馈的核心评论内容.针对评论模式和评论种子中的词和词性,从评论内容和句式结构两方面挖掘用户评论,挖掘出与评论种子相同或相似的用户评论与评论种子体现相同的使用反馈类型.

## 4 挖掘体现使用反馈的用户评论

由于体现 APP 软件使用反馈类型的用户评论变化多样,人工标注的数据资源无法满足挖掘体现使用反馈的用户评论的要求.因此,本文采用半监督学习的方式,主要通过循环挖掘过程中动态扩充评论种子库,扩大挖掘体现使用反馈的用户评论的范围,在人工标注数据有限的情况下提高挖掘体现使用反馈的用户评论的能力.循环挖掘过程包括以下 4 个步骤.

(1) 通过评论种子挖掘与评论种子相同或相似的体现使用反馈的用户评论:不局限于特定的评价对象、评价观点或固定的评论句式结构,应用评论种子定义的词、词性、权重及距离这 4 个属性共同挖掘与评论种子相同或相似的用户评论.

(2) 通过能愿动词挖掘体现“软件未达到的期望”的用户评论:评论种子的数量限定了能被挖掘出的体现使用反馈的用户评论是有限的,但与评论种子匹配失败的用户评论中仍然可能包含体现使用反馈的用户评论,通过明确体现“用户期望”的能愿动词挖掘用户评论,此类用户评论可以体现“软件未达到的期望”这一使用反馈类型,抽取其评论模式进入候选评论模式库.

(3) 通过软件简介和情感分析挖掘体现“软件存在的问题”和“软件满足的需求”的用户评论:由于能愿动词有较强的针对性,用户评论数目相对较少,剩余的用户评论中仍然可能包含体现使用反馈的用户评论,因此通过软件简介判断该用户评论是否与评价软件相关,针对与评价软件相关的用户评论通过情感分析的方式判断该

© 中国科学院软件研究所 http://www.jos.org.cn

用户表达出的情感观点正负,确定该用户评论体现的使用反馈类型——“软件存在的问题”或“软件满足的需求”,抽取其评论模式进入候选评论模式库.

(4) 通过候选评论模式抽取新评论种子:通过评论种子可以挖掘与之相同或相似的、体现使用反馈的用户评论,动态扩充评论种子可以适应评论数据量大、更新快的特点,实现体现使用反馈的用户评论的动态挖掘.本文针对每次循环挖掘,建立不同的候选评论模式库,该候选评论模式库中包含未挖掘出的、体现使用反馈的用户评论,基于候选评论模式库抽取新的具有代表性的评论种子,进入下一次的循环,挖掘剩余评论数据中体现使用反馈的用户评论.

挖掘体现使用反馈的用户评论的流程图如图 1 所示.

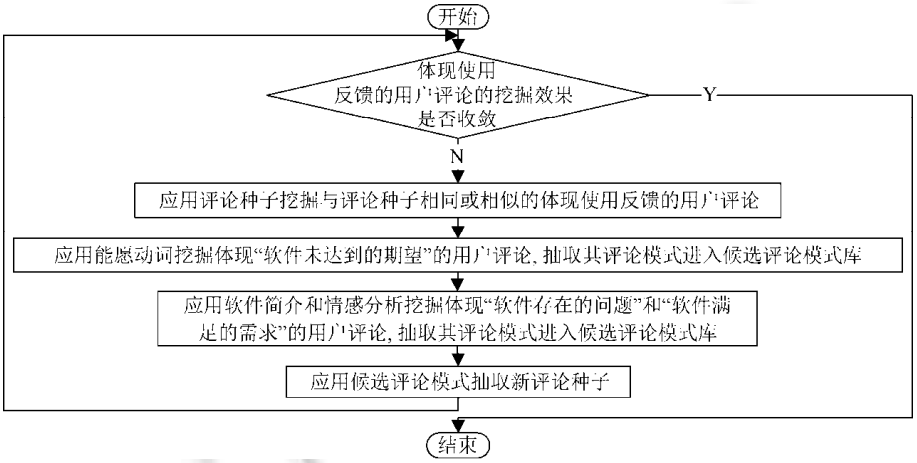


Fig.1 The flow diagram of mining user’s comments reflecting usage feedback

图 1 挖掘体现使用反馈的用户评论流程图

本文针对循环挖掘体现使用反馈的用户评论定义挖掘收敛条件,针对第 *time* 次循环,应用 *Levenshtein* 相似度计算方法[24]计算新评论种子的评价对象和评价观点与已有评论种子的评价对象和评价观点之间的文本相似度.为了确定文本相似度的阈值,本文随机选取了一批体现 APP 软件使用反馈的评论种子,多次计算任意两个评论种子对应的评价对象和评价观点之间的文本相似度.我们发现,当词性组合完全相同且词完全不同时,文本相似度为 0.4.如果两个评论种子的评价对象和评价观点的词性组合完全相同且存在相同词,文本相似度会大于 0.4,此时,两个评论种子体现的使用反馈相似,例如评论种子“〈软件+功能+好,n+n+a,0.33+0.33+0.33,7,软件满足的需求〉”和“〈视频+界面+好,n+n+a,0.33+0.33+0.33,7,软件满足的需求〉”对应的评价对象和评价观点之间的文本相似度为 0.5,且这两者均体现“软件满足的需求”这一使用反馈类型.根据上述原因,我们计算与已有评论种子的文本相似度的最大值大于 0.4 的新评论种子数 $numNewSeed'_{time}$,当 $numNewSeed'_{time}$ 与该次循环产生新评论种子总数 $numNewSeed_{time}$ 相同时,体现使用反馈的新评论种子与已有评论种子完全相似,应用新评论种子挖掘相似的用户评论的数量显著降低,停止扩充评论种子库,挖掘过程结束.体现使用反馈的用户评论的挖掘效果收敛判断依据见公式(2):

$$\frac{numNewSeed'_{time}}{numNewSeed_{time}}=1 \tag{2}$$

### 4.1 应用评论种子挖掘用户评论

在图 1 中,首先需要将一条用户评论的评论模式与某个评论种子匹配,判断该用户评论是否与某个体现使用反馈的评论种子相似,与评论种子相似的用户评论与评论种子体现相同的使用反馈类型.评论种子中包含 5 个属性:词、词性、权重、距离和使用反馈类型,其中,词、词性、权重是判断一条用户评论是否与评论种子的词(词性)匹配的关键,距离是综合判断一条用户评论与评论种子是否匹配的关键.因此,假设某条用户评论

© 中国科学院软件研究所 http://www.jos.org.cn

$comment_m$ 的评论模式为 $mode_m$,某评论种子为 $seed_i$,判断 $comment_m$ 与 $seed_i$ 是否匹配的过程主要包括两个部分:计算词(词性)匹配值和计算综合匹配值.

4.1.1 计算词(词性)匹配值

首先,针对每条用户评论 $comment_m$ 与评论种子 $seed_i$ 的初始化词(词性)匹配值 $word_match_{mi}$($speech_match_{mi}$)为 0.进行词匹配时,将评论种子 $seed_i$ 的词 $word_{ij}$ 和 $comment_m$ 的词依次进行匹配,每一次词匹配开始的位置都是 $comment_m$ 中上一次匹配成功的词的位置,若 $word_{mn}$ 匹配 $word_{ij}$ 成功,则记录评论中词的位置 $n$ 到词匹配成功位置集合 $pos_word$ 中,且词匹配标记 $word_flag_{ij}$ 为 1,否则为 0,所有词匹配完成后,$word_match_{mi}$ 的计算如公式(3)所示:

$$word_match_{mi} = \sum_{j=1}^{p} word_flag_{ij} \times weight_{ij} \tag{3}$$

计算词匹配值的具体流程如图 2 所示.

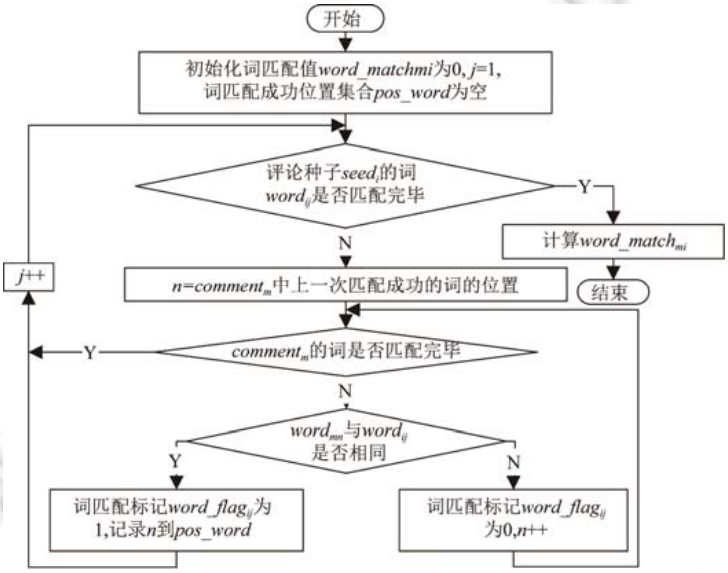


Fig.2 The flow diagram of calculating word-matching value

图 2 词匹配值计算流程

词性匹配值计算流程与词匹配值计算流程相似,不同之处在于:进行第 1 次词性匹配时,从 $comment_m$ 匹配成功的第 1 个词的位置开始,将评论种子 $seed_i$ 的词性 $speech_{ij}$ 和用户评论 $comment_m$ 的词性依次进行匹配,第 1 次词性匹配之后,每一次词性匹配开始的位置都是 $comment_m$ 中上一次匹配成功的词性的位置.若 $speech_{mn}$ 匹配 $speech_{ij}$ 成功,则记录用户评论中词性的位置 $n$ 到词性匹配成功位置集合 $pos_speech$ 中,且词性匹配标记 $speech_flag_{ij}$ 为 1,否则,$speech_flag_{ij}$ 为 0,所有词性匹配完成后,$speech_match_{mi}$ 的计算如公式(4)所示:

$$speech_match_{mi} = \sum_{j=1}^{p} speech_flag_{ij} \times weight_{ij} \tag{4}$$

4.1.2 计算综合匹配值

完成词(词性)匹配值计算后需要进行综合判断,由于词和词性对于判断用户评论是否与评论种子匹配都至关重要,所以通过计算 $comment_m$ 与 $seed_i$ 的综合匹配值 $both_match_{mi}$ 来判断用户评论与评论种子是否匹配.由于用户评论与评论种子匹配成功的词(词性)的最大距离反映了该用户评论是否满足评论种子的语法表达规则,若匹配成功的词(词性)的最大距离大于或等于评论种子的距离,则表明用户评论与评论种子匹配成功的词(词性)不在评价对象和评价观点之间可扩展的最大距离定义的位置范围内,不符合评论种子所定义的语法表达规则,反之,则满足.因此,为判断用户评论与评论种子匹配成功的词(词性)是否符合评论种子定义的语法距离限定,需

© 中国科学院软件研究所 http://www.jos.org.cn

要计算 $comment_m$ 和 $seed_i$ 匹配成功的词(词性)之间的最大距离 $dis_word_{mi}$($dis_speech_{mi}$)是否小于评论种子的距离 $dis_i$.由于词(词性)匹配都是从上一次词(词性)匹配成功的位置开始的,因此集合 $pos_word$($pos_speech$)是递增序列,故用公式(5)和公式(6)计算 $dis_word_{mi}$ 和 $dis_speech_{mi}$:

$$dis_word_{mi}=\mathrm{MAX}\{pos_word\}-\mathrm{MIN}\{pos_word\} \tag{5}$$

$$dis_speech_{mi}=\mathrm{MAX}\{pos_speech\}-\mathrm{MIN}\{pos_speech\} \tag{6}$$

在用户评论的词或词性不满足评论种子所定义的语法表达规则时,不论用户评论与评论种子的词(词性)匹配值为多少,综合匹配值都应该为 0;否则,根据词匹配值和词性匹配值共同计算综合匹配值 $both_match_{mi}$,计算如公式(7)所示:

$$both_match_{mi}=\begin{cases}0, & dis_i \leqslant dis_word_{mi}\\ 0, & dis_i \leqslant dis_speech_{mi}\\ 0.5word_match_{mi}+0.5speech_match_{mi}, & dis_i > dis_word_{mi}, dis_i > dis_speech_{mi}\end{cases} \tag{7}$$

用户评论 $comment_m$ 需要与所有评论种子依次进行匹配,最后抽取其与第 $g$ 个评论种子的最大综合匹配值 $both_match_{mg}$,判断是否匹配成功,抽取最大综合匹配值如公式(8)所示:

$$both_match_{mg}=\mathrm{MAX}\{both_match_{m1},\ldots,both_match_{mg},\ldots,both_match_{ms}\}(1\leqslant g\leqslant s) \tag{8}$$

$comment_m$ 和 $seed_g$ 的综合匹配值越高,该用户评论与该评论种子匹配成功的可能性越大,且使用评论种子挖掘出的正确的用户评论条数越多.这里需要设定一个阈值,只有当综合匹配值大于该阈值时,该用户评论与评论种子 $seed_g$ 才能匹配成功.在用户评论与评论种子匹配的过程中,词匹配成功可以保证其对应词性匹配成功,而词性匹配成功不能保证词匹配成功.因此,即使在词性完全匹配的情况下(其词性匹配值为 1),仍无法保证该用户评论中存在与评论种子匹配成功的词,即无法保证该用户评论与评论种子的评价对象或评价观点相关.为了保证与评论种子匹配成功的用户评论中包含体现该评论种子使用反馈类型的评价对象或评价观点,本文将阈值设置为 0.5.

当最大的综合匹配值 $both_match_{mg}>0.5$ 时,$comment_m$ 与 $seed_g$ 的评论内容和句式结构比较相似,$comment_m$ 与 $seed_g$ 匹配成功,$comment_m$ 与 $seed_g$ 体现相同的使用反馈类型;否则,$comment_m$ 与评论种子库中所有评论种子匹配失败.其中,若最大的综合匹配值 $both_match_{mg}=1$,$comment_m$ 与 $seed_g$ 的评论内容和句式结构完全相同.

例如,用户评论“播放/v 界面/n 真的/d 非常/d 丑/a”与评论种子“〈界面+很+丑,n+d+a,0.5+0+0.5,6,软件存在的问题〉”进行词匹配,匹配成功的词为“界面”和“丑”,$word_match_{mi}$ 为 1;进行词性匹配,匹配成功的词性为“n”和“a”,$speech_match_{mi}$ 为 1.在以上用户评论中,匹配成功的词“界面”和“丑”的位置分别为 2 和 5,匹配成功的词性之间的最大距离为 3;匹配成功的词性为“n”和“a”的位置分别为 2 和 5,匹配成功的词之间的最大距离为 3.由于词(词性)匹配值均小于 6,所以,$both_match_{mi}$ 为 1,评论文本与评论种子匹配成功,评论文本与评论种子体现相同的使用反馈——“软件存在的问题”.

在用户评论与评论种子匹配的过程中,由于评论种子库中的评论种子并不能代表所有评论,部分体现 APP 软件使用反馈的用户评论与评论种子匹配失败,这些用户评论包含与已有评论种子差异较大的、体现使用反馈的评论内容,应该从中抽取出新的具有代表性的评论种子,以便于后续挖掘出更多与新评论种子相似的、体现使用反馈的用户评论.为了保证新抽取的评论种子能够体现使用反馈,需要挖掘出与评论种子匹配失败的用户评论中可以体现“软件满足的需求”“软件存在的问题”或“软件未达到的期望”这 3 个不同使用反馈类型的用户评论.本文主要应用能愿动词挖掘体现“软件未达到的期望”的用户评论,应用软件简介及情感分析挖掘体现“软件满足的需求”和“软件存在的问题”的用户评论,通过上述两种方式挖掘出的体现使用反馈的用户评论建立候选评论模式库,在候选评论模式库的基础上抽取新评论种子.由于针对不同使用反馈类型,代表性和抽象性高的评论种子的挖掘效果更好,因此,针对体现相同使用反馈的评论模式进行文本相似度计算,以此抽取体现不同使用反馈类型的具有代表性的新评论种子.

### 4.2 应用能愿动词挖掘用户评论并抽取评论模式

通过分析大量用户评论后发现,体现 APP 软件未达到的期望的用户评论数量较少,且该类型用户评论可能

© 中国科学院软件研究所 http://www.jos.org.cn

包含其他针对软件总体或部分已有功能等方面的评价.为了挖掘出该类用户评论,针对评论种子匹配失败的用户评论,本文首先应用能愿动词挖掘出体现“软件未达到的期望”的用户评论,根据文献[25],抽取出如下针对体现使用反馈的能愿动词:“应该、应当、须得、必得、乐意、愿、愿意、情愿、想、想要、要、要想、希望、企图、好意思、乐得、高兴、乐于、敢于、勇于、甘于、苦于、懒得、便于、有助于、难于、易于、善于、适于、宜于”.通过上述能愿动词抽取体现“软件未达到的期望”的用户评论的评论模式进入候选评论模式库.

### 4.3 应用软件简介及情感分析挖掘用户评论并抽取评论模式

在挖掘体现“软件未达到的期望”的用户评论之后,剩余的用户评论中包含针对该软件特征进行评论的信息,本文通过应用软件简介的方式判断用户评论是否针对该软件进行评价,并通过情感值计算的方式判断用户评论针对该软件表达的情感正负,其中,情感表达为正的用户评论体现“软件满足的需求”,情感表达为负的评论体现“软件存在的问题”.

APP 软件简介由开发者官方发布并具有权威性,这些信息描述了该 APP 软件的大部分特征,可以作为判断用户评论是否针对该软件进行评价的依据,因此,本文提取 APP 软件简介中的名词和动词系列的相关词作为软件特征词,通过判断用户评论中是否包含这些特征词分析用户评论是否体现该软件的用户反馈.在挖掘出的针对该软件的用户评论的基础上,应用评价对象和评价观点抽取规则,抽取该用户评论的评价观点,针对评价观点通过构建的情感极值表计算该用户评论的情感值,挖掘体现使用反馈的用户评论进入候选评论模式库.

体现使用反馈的用户评论类型包括以下两种.

(1) 若该情感值为正,则表明该用户评论针对软件表达正面情感,用户实际使用该软件过程中该软件满足某个功能特征或软件的整体使用感受较好,即体现“软件满足的需求”;

(2) 若该情感值为负,则表明该用户评论针对软件表达负面情感,用户实际使用该软件过程中该软件未满足某功能特征或软件的整体使用感受较差,即该用户评论体现“软件存在的问题”.

通过应用软件简介及情感分析挖掘出体现“软件满足的需求”或“软件存在的问题”的用户评论,并抽取其评论模式构建候选评论模式库,保证候选评论模式库中在评论数据较为充足的情况下有体现多个使用反馈类型的评论模式,以便抽取出代表不同使用反馈类型的新评论种子.

### 4.4 应用候选评论模式抽取新评论种子

在用户评论与评论种子匹配的过程中,由于评论种子库中的评论种子数是有限的,部分用户评论会出现与评论种子匹配失败的情况.这些用户评论可能体现了 APP 软件使用反馈,此外,这些评论实际上蕴藏着评论种子库中不存在的内容,应该从与已有评论种子匹配失败的用户评论中抽取出新的评论种子,以便于后续挖掘出更多与之相似的体现使用反馈的用户评论.通过第 4.2 节和第 4.3 节中的方法,挖掘出与评论种子匹配失败的用户评论中体现“软件满足的需求”“软件存在的问题”和“软件未达到的期望”这 3 种不同使用反馈类型的用户评论,并建立相应的候选评论模式库.进入候选评论模式库的评论模式各自体现了不同的使用反馈类型,且体现相同使用反馈类型的评论模式在表达内容和句式结构上更具有相似性,所以,针对体现相同使用反馈的评论模式进行文本相似度计算,以抽取体现不同使用反馈类型的具有代表性的新评论种子.

由于候选评论模式库中包含体现 3 种不同使用反馈类型的候选评论模式,所以本文参考文献[26]这一专利——基于候选评论模式库抽取反映不同软件使用质量属性的新评论种子的方法,主要针对体现相同使用反馈类型的评论模式进行文本相似度计算从而抽取出新的、体现不同使用反馈类型、具有代表性的新评论种子.应用评论模式库抽取新评论种子主要通过以下 3 个步骤来实现.

(1) 抽取体现不同使用反馈类型具有代表性的评论模式

参考文献[26]这一专利应用 Levenshtein 相似度[24]的计算方法,本文在不同的使用反馈类型下,计算评论模式与其他评论模式的词文本相似度和词性文本相似度,针对 3 种不同使用反馈类型,每次循环过程中计算出至多 3 个综合文本相似度最大的、具有代表性的评论模式.

(2) 抽取评论模式对应的评论种子

© 中国科学院软件研究所 http://www.jos.org.cn

基于至多 3 个代表不同使用反馈类型的文本相似度最大的评论模式,结合本文第 3.1 节中的评价对象和评价观点抽取规则,确定评论模式的评价对象和评价观点,根据评价对象和评价观点确定该评论模式中的词及其词性所对应的权重,即该评论模式对应的评论种子的权重,最终抽取出这 3 个评论模式的评论种子.

(3) 更新新评论种子的距离

根据评论种子距离的定义,为了保证即将进入评论种子库中的评论种子的距离可以代表评价对象和评价观点之间可扩展的最大距离,本文应用文献[26]这一专利中更新评论种子的距离的方法,抽取其中的最大值作为该评论种子的距离.

通过上述方法,在人工标注的初始评论种子有限的情况下,首先应用评论模式和评论种子,综合评论种子的多个属性,将用户评论与评论种子进行匹配,匹配成功的用户评论可以体现与评论种子相同的使用反馈;其次,通过能愿动词、软件简介和情感分析挖掘出与评论种子匹配失败的用户评论中的体现使用反馈的用户评论,并构建候选评论模式库;最后,针对每次循环挖掘过程中构建的候选评论模式库抽取具有代表性的新评论种子,用于下一次的循环挖掘,实现体现使用反馈的用户评论的循环挖掘.

## 5 实验结果及分析

### 5.1 实验数据及来源

为了验证本文方法的有效性,我们使用 Java 语言开发了一个体现使用反馈的 APP 软件用户评论挖掘原型工具(mining user’s comment based on seed,简称 MUCBS)进行实验.我们从安卓电子市场(http://apk.hiapk.com/apps)随机爬取了多款 APP 软件的用户评论,并据此建立了 APP 软件用户评论库.本文从 APP 软件用户评论库中随机选取娱乐类、视频类、购物类、社交类和工具类的 92 811 条用户评论,使用 ICTCLAS 2018 作为数据预处理的工具,完成了用户评论分词及词性标注.由于网络用户评论往往存在大量的网络词汇和拼写错误,可能导致分词结果存在误差,目前,本文对此没有进行处理.在完成分词的用户评论数据的基础上,过滤其中不包含评价对象(即词性中不包含 n 或 v)的用户评论,最终针对 50 072 条用户评论数据进行挖掘(经人工标记,其中有 38 789 条体现使用反馈的用户评论).

### 5.2 体现使用反馈的用户评论挖掘

在本实验中,通过分析常见的体现使用反馈的用户评论,我们随机设置了 11 条比较具有代表性的初始评论种子.每次循环挖掘过程中,当次循环过程中与评论种子匹配成功的评论不再进入下一次循环挖掘,应用当次循环新抽取的评论种子进行下一次循环挖掘.

#### 5.2.1 候选评论模式库的建立

首先,建立体现使用反馈的候选评论模式库以便抽取可以合理体现使用反馈的新评论种子.由于与评论种子匹配成功的评论不再进入下一次循环挖掘,所以第 1 次循环挖掘结束时抽取的候选评论模式数量多、覆盖广,本文针对 50 072 条用户评论进行第 1 次循环挖掘后建立的候选评论模式库进行了分析,分析结果见表 3.

表 3 中,通过应用能愿动词挖掘体现“软件未达到的期望”的用户评论、应用软件简介及情感分析挖掘体现“软件存在的问题”及“软件满足的需求”的用户评论建立候选评论模式库.例如,表 3 中候选评论模式“〈非常+棒+平台,d+a+n,0.0+0.5+0.5,软件满足的需求〉”,依据第 2 节中 APP 软件使用反馈的定义,用户评论“非常棒平台”体现出软件整体给用户较好的使用体验,因此,该候选评论模式体现了“软件满足的需求”这一使用反馈类型.通过综合分析,抽取的候选评论模式的平均准确率为 88.71%,表明本文抽取的候选评论模式库是有效的.但是,根据抽取的部分用户评论的候选评论模式可以看出,在挖掘体现使用反馈的用户评论过程中仍存在一些错误.

(1) 应用能愿动词挖掘出的用户评论的评论模式为“〈建议+大家+下载,n+rr+v,0.5+0.0+0.5,软件未达到的期望〉”,该用户评论中能愿动词“建议”表示该评论用户给其他软件用户的建议,无法体现“软件未达到的期望”.

(2) 通过情感分析挖掘出的负面评论模式“〈好+到+不+行,a+v+d+vi,0.0+0.0+0.5+0.5,软件存在的问题〉”,针对“不/d 行/vi”进行情感分析后结果为负面,而该用户是通过夸张手法表达的是正面情感,本文方法在进行情感

© 中国科学院软件研究所 http://www.jos.org.cn

分析的过程中针对“夸张”“讽刺”等表达方式挖掘效果较差.

建立体现使用反馈的候选评论模式库后,可以基于候选评论模式库进一步扩充评论种子库,从而实现体现使用反馈的用户评论的循环挖掘.

**Table 3** The example of the candidate comment mode library

**表 3** 候选评论模式库示例

| 候选评论模式抽取方法 | 使用反馈类型 | 抽取用户评论的候选评论模式(部分) | 用户评论的候选评论模式是否正确 | 候选评论模式的数量 | 候选评论模式准确率(%) |
|---|---|---|---|---|---|
| 应用能愿动词挖掘用户评论 | 软件未达到的期望 | 〈希望+开发+游戏+键盘,v+vn+n+n,0.334+0.0+0.33+0.33,软件未达到的期望〉<br>〈建议+大家+下载,n+rr+v,0.5+0.0+0.5,软件未达到的期望〉 | 是<br>否 | 752 | 90.69 |
| 应用软件简介及情感分析挖掘用户评论 | 软件存在的问题 | 〈广告+很+烦人,n+d+a,0.5+0.0+0.5,软件存在的问题〉<br>〈好+到+不+行,a+v+d+vi,0.0+0.0+0.5+0.5,软件存在的问题〉 | 是<br>否 | 2 148 | 88.46 |
| | 软件满足的需求 | 〈非常+棒+平台,d+a+n,0.0+0.5+0.5,软件满足的需求〉<br>〈广告+太+多,n+d+a,0.5+0.0+0.5,软件满足的需求〉 | 是<br>否 | 8 417 | 86.97 |

5.2.2 应用评论种子挖掘用户评论的效果

为了验证本文提出的循环挖掘的有效性,在建立体现使用反馈的候选评论模式库的基础上,应当抽取出能够合理体现使用反馈的新评论种子,并且应用该评论种子可以挖掘体现使用反馈的用户评论.针对第 5.2.1 节的 50 072 条用户评论进行多次循环挖掘,直到针对体现使用反馈的用户评论挖掘收敛,参与循环挖掘的评论种子情况见表 4.

**Table 4** Comment seeds in interative mining process

**表 4** 循环挖掘过程中的评论种子

| 循环次数 | 用于本次循环挖掘的评论种子 | 评论种子数量 |
|---|---|---|
| 1 | 〈安装+不+了,v+d+y,0.5+0.5+0,3,软件存在的问题〉,〈内存+太+大,n+d+a,0.5+0+0.5,6,软件存在的问题〉,〈下载+速度+真的+很+慢,v+n+d+d+a,0.33+0.33+0+0+0.33,7,软件存在的问题〉,〈怎么+不+能+安装,ryv+d+v+v,0+0.33+0.33+0.33,5,软件存在的问题〉,〈打+不+开,v+d+v,0.33+0.33+0.33,5,软件存在的问题〉,〈不错+软件,a+n,0.5+0.5,5,软件满足的需求〉,〈没有+播放+功能,v+v+n,0.33+0.33+0.33,5,软件未达到的期望〉,〈不+喜欢+界面,d+v+n,0.33+0.33+0.33,5,软件未达到的期望〉,〈新+版本+不+好+用,a+n+d+a+v,0+0.33+0.33+0.33+0,7,软件未达到的期望〉,〈软件+很+好,n+d+a,0.5+0+0.5,6,软件满足的需求〉,〈不+可+用,d+v+v,0.33+0.33+0.33,5,软件存在的问题〉 | 11 |
| 2 | 〈希望+越来越+好,v+d+a,0.5+0+0.5,8,软件未达到的期望〉,〈很+好+用+,,d+a+v+wd,0+0.5+0.5+0,8,软件满足的需求〉,〈为什么+看+不+了+,+会黑屏,ryv+v+d+y+wd+nr,0+0.5+0.5+0+0+0,14,软件存在的问题〉 | 3 |
| 3 | 〈非常+好+,+功能+很多+,+我+很+喜欢,d+a+wd+n+m+wd+rr+d+vi,0+0.5+0+0.5+0+0+0+0+0,20,软件满足的需求〉,〈希望+能+做+更+好,v+v+v+d+a,0.33+0.33+0.0+0.0+0.33,10,软件未达到的期望〉,〈QQ+空间+直播+,+不+了+,+差+评,n+n+vn+wd+d+y+wd+v+v,0.33+0.33+0+0+0.33+0+0+0+0,13,软件存在的问题〉 | 3 |
| 4 | 〈没有+五+笔+,+不+好,v+m+q+wd+d+a,0.33+0+0+0+0.33+0.33,10,软件存在的问题〉,〈可以+,+很+好,v+wd+d+a,0.5+0+0+0.5,8,软件满足的需求〉,〈太+棒+,+建议+大家+下载,d+a+wd+v+rr+v,0+0.5+0+0.5+0+0,12,软件未达到的期望〉 | 3 |
| 5 | 〈没用+了+,+用+不+了,vi+y+wd+p+d+y,0.5+0+0+0+0.5+0,8,软件存在的问题〉,〈希望+,+没有+广告,v+wd+v+n,0.33+0+0.33+0.33,7,软件未达到的期望〉,〈好玩+,+我+喜欢,a+wd+rr+vi,0.5+0+0+0.5,12,软件满足的需求〉 | 3 |

在表 4 中,由于第 5 次循环抽取的新评论种子中,与已有评论种子的文本相似度大于 0.4 的新评论种子数为 3,与该次循环产生新评论种子的总数相同.因此,第 5 次循环挖掘后,抽取的新评论种子与已有评论种子相似,循环挖掘效果收敛,停止扩充评论种子库,体现使用反馈的用户评论的循环挖掘停止.根据表 4 中的评论种子可以看出,随着循环次数的增加,部分新评论种子的抽象性逐渐降低.例如,新评论种子“〈QQ+空间+直播+,+不+了+,+差+评,n+n+vn+wd+d+y+wd+v+v,0.33+0.33+0+0+0.33+0+0+0+0,13,软件存在的问题〉”针对具体的评价对象,即“QQ 空间”.

此外,由于循环次数增加,针对某一使用反馈类别,部分新评论种子对应的评论模式越来越复杂,即评论模式对应的用户评论中可能包含多个分句,且分句语义不连贯.针对这一情况,抽取新评论种子的评价对象和评价观点的准确度降低.例如,针对“软件存在的问题”这一使用反馈类别,新评论种子对应的用户评论中往往包含多个分句,在第 5 次循环挖掘过程中,新评论种子“〈没用+了+,+用+不+了,vi+y+wd+p+d+y,0.5+0+0+0+0.5+0,8,软件

© 中国科学院软件研究所 http://www.jos.org.cn

存在的问题〉”对应的评价对象和评价观点为“没用/vi”和“不/d”,该评论种子的评价对象和评价观点无实质含义.然而,针对“软件满足的需求”这一使用反馈类别,由于用户评论表达简洁的情况较为常见,因此,在循环挖掘过程中,体现“软件满足的需求”的新评论种子的抽象性整体较高,复杂性较低,如“〈可以+,+很+好,v+wd+d+a,0.5+0+0+0.5,8,软件满足的需求〉”和“〈好玩+,+我+喜欢,a+wd+rr+vi,0.5+0+0+0.5,12,软件满足的需求〉”.经过 5 次循环挖掘,体现使用反馈的用户评论挖掘效果如图 3 所示.

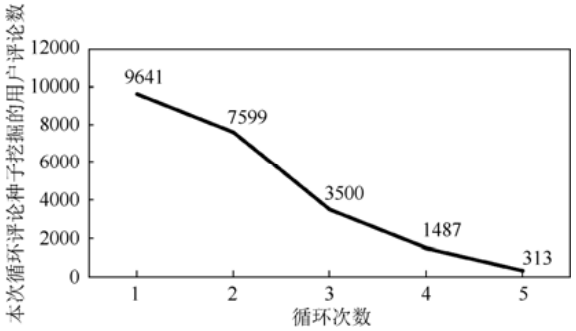


Fig.3 The mining effect of comment seed in interative mining process

图 3 循环挖掘过程中评论种子挖掘效果

图 3 中,由于参与第 1 次循环挖掘的评论种子为人工标注的初始评论种子,数量较多且抽象程度较高,所以,第 1 次循环挖掘效果较好.根据图 3 中应用评论种子挖掘出的用户评论数的趋势可以看出,循环中应用评论种子挖掘的用户评论数随着循环次数的增加而减少,第 5 次循环挖掘结果中应用新评论种子挖掘的用户评论数量显著降低,这主要是因为随着循环次数的增加,体现使用反馈的新评论种子的抽象性逐渐降低,评论数据中与新评论种子相似的用户评论较少,应用新评论种子挖掘用户评论的效果逐渐变差.

5.2.3 体现使用反馈的用户评论整体挖掘效果

经过 5 次体现用户反馈的用户评论循环挖掘后,共有 23 条评论种子参与了循环挖掘,应用本文方法共挖掘出 30 252 条体现使用反馈的用户评论,体现不同使用反馈类型的用户评论的整体挖掘结果如图 4 所示.

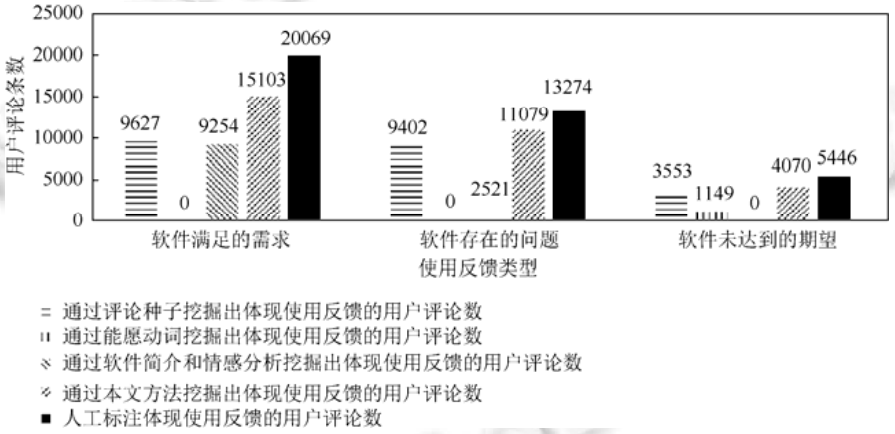


Fig.4 The overall mining effect of user’s comment reflecting usage feedback

图 4 体现使用反馈的用户评论的整体挖掘效果

图 4 中,由于本文通过能愿动词仅挖掘体现“软件未到达的期望”的用户评论,所以通过能愿动词挖掘体现“软件满足的需求”和“软件存在的问题”的用户评论数为 0.与之类似,通过软件简介和情感计算也无法挖掘出体现“软件未达到的期望”的用户评论.此外,前 4 项挖掘出的用户评论是存在重复的,通过评论种子挖掘的用户评论中包含通过能愿动词或软件简介和情感分析的方式挖掘出的部分用户评论.由图 4 可以看出,与人工标注的体现使用反馈的用户评论相比,本文的方法针对体现“软件满足的需求”和“软件未达到的期望”的用户评论挖掘率较低,分别为 75.26%和 74.73%,针对体现“软件存在的问题”这一使用反馈类型的用户评论挖掘率最高为 83.46%.由于体现“软件满足的需求”的用户评论可能是针对软件整体的评论,因此应用包含明确评价对象的评

© 中国科学院软件研究所 http://www.jos.org.cn

论种子挖掘一般性的整体评论,其挖掘率较低,如“非常/d 好/a”这种表述简洁、单一的用户评论没有被挖掘出来.与体现“软件满足的需求”的用户评论相比,体现“软件存在的问题”的用户评论表达更具有针对性,表达方式相对较少,因此,应用本文提出的评论种子概念,基于初始和循环挖掘出的体现“软件存在的问题”的评论种子可以挖掘出较多的与之相似的体现“软件存在的问题”的用户评论.此外,由于通过能愿动词挖掘出的部分用户评论不体现用户针对软件评价的“软件未达到的期望”,如用户评论“很/d 好/a ,/wd 希望/v 大家/rr 快快/d 下载/v”中“希望/v”的表达对象是其他用户,所以针对体现“软件未达到的期望”的用户评论挖掘率较低.

最后综合分析,体现使用反馈的用户评论的平均挖掘率(即 3 种不同使用反馈类型的用户评论挖掘率的平均值)为 77.82%,说明本文提出的方法是有效的.

5.2.4 不同初始评论种子的挖掘效果

本文采用半监督自学习的方式,基于有限数量和类型的评论种子,通过多次循环的方式挖掘体现使用反馈的用户评论,这是一个评论种子逐渐扩充、挖掘效果逐渐提升的过程.为了验证本文提出的循环挖掘方法的有效性,我们针对第 5.2.1 节中的 50 072 条用户评论设计了 3 个不同的循环挖掘实验,每个实验中,初始评论种子的数量或类型是不同的.实验 1 的评论种子见表 4,实验 2 和实验 3 的初始评论种子及其整体挖掘效果见表 5.

**Table 5** Initial comment seeds and overall mining effects of Experiment 2 and Experiment 3

**表 5** 实验 2 和实验 3 的初始评论种子及其整体挖掘效果

| 实验 | 初始评论种子数 | 初始评论种子 | 循环挖掘次数 | 新评论种子数 | 新评论种子(部分) | 本文方法挖掘的体现使用反馈的用户评论数 | 挖掘率 |
|---|---|---|---|---|---|---|---|
| 2 | 11 | 〈怎么+不+能+使用,ryv+d+v+v,0+0.33+0.33+0.33,5,软件存在的问题〉,〈占+内存,v+n,0.5+0.5,4,软件存在的问题〉,〈验证+失败,v+vi,0.5+0.5,4,软件存在的问题〉,〈速度+慢,n+a,0.5+0.5,6,软件存在的问题〉,〈登+不+了,v+d+y,0.5+0.5+0,3,软件存在的问题〉,〈广告+多,n+a,0.5+0.5,6,软件存在的问题〉,〈功能+很+强大,n+d+a,0.5+0+0.5,6,软件满足的需求〉,〈使用+方便,v+a,0.5+0.5,6,软件满足的需求〉,〈不错+软件,a+n,0.5+0.5,5,软件满足的需求〉,〈不+喜欢+界面,d+v+n,0.33+0.33+0.33,5,软件未达到的期望〉,〈希望+功能+越来越+多,v+n+d+a,0.33+0.33+0+0.33,7,软件未达到的期望〉 | 6 | 15 | 〈很+好+用+,,d+a+v+wd,0+0.5+0.5+0,8,软件满足的需求〉,〈小说+看+不+了+,+垃圾,n+v+d+y+wd+n,0.33+0.33+0.33+0.0+0.0+0.0,14,软件存在的问题〉,〈希望+能+做+更+好,v+v+v+d+a,0.33+0.33+0.0+0.0+0.33,10,软件未达到的期望〉,〈没有+五+笔+,+不+好,v+m+q+wd+d+a,0.33+0.0+0.0+0.0+0.33+0.33,10,软件存在的问题〉,〈好玩+,+我+喜欢,a+wd+rr+vi,0.5+0.0+0.0+0.5,12,软件满足的需求〉,〈希望+,+没有+广告,v+wd+v+n,0.33+0.0+0.33+0.33,7,软件未达到的期望〉 | 29 970 | 77.26% |
| 3 | 8 | 〈怎么+不+能+下载,ryv+d+v+v,0+0.33+0.33+0.33,5,软件存在的问题〉,〈登+不+了,v+d+y,0.5+0.5+0,3,软件存在的问题〉,〈不+喜欢+界面,d+v+n,0.33+0.33+0.33,5,软件未达到的期望〉,〈不错+软件,a+n,0.5+0.5,5,软件满足的需求〉,〈打+不+开,v+d+v,0.33+0.33+0.33,5,软件存在的问题〉,〈希望+功能+越来越+多,v+n+d+a,0.33+0.33+0+0.33,7,软件未达到的期望〉,〈占+内存,v+n,0.5+0.5,4,软件存在的问题〉,〈速度+慢,n+a,0.5+0.5,6,软件存在的问题〉 | 5 | 12 | 〈很+好+用+,,d+a+v+wd,0+0.5+0.5+0,8,软件满足的需求〉,〈小说+看+不+了+,+垃圾,n+v+d+y+wd+n, 0.33+0.33+0.33+0.0+0.0+0.0,14,软件存在的问题〉,〈希望+能+做+更+好,v+v+v+d+a,0.33+0.33+0.0+0.0+0.33,10,软件未达到的期望〉,〈非常+好+,+功能+很多+,+我+很+喜欢,d+a+wd+n+m+wd+rr+d+vi,0+0.5+0+0.5+0+0+0+0+0,20,软件满足的需求〉,〈没有+五+笔+,+不+好,v+m+q+wd+d+a,0.33+0.0+0.0+0.0+0.33+0.33,10, 软件存在的问题〉,〈没用+了+,+用+不+了,vi+y+wd+p+d+y,0.5+0.0+0.0+0.0+0.5+0.0,9,软件存在的问题〉 | 28 659 | 73.88% |

通过分析 3 个不同实验的整体挖掘效果可以发现,应用不同数量和不同类型的初始评论种子进行体现使用反馈的用户评论挖掘,每个实验最终挖掘出的新评论种子是相似的,这些新评论种子代表了该批数据中常见

© 中国科学院软件研究所 http://www.jos.org.cn

的评论表达.在实验 1~实验 3 中,共同出现在 3 个不同的实验的新评论种子比例分别为 75.00%、60.00%和 75.00%.此外,单次循环挖掘效果取决于参与本次循环挖掘的评论种子的代表性和数量,例如实验 1 和实验 2 应用相同数量的初始评论种子挖掘出的用户评论条数分别为 9 641 和 8 113,这表明,在评论种子数量相同的情况下,评论种子的代表性越高,挖掘效果越好.实验 3 应用 8 条初始评论种子挖掘出 7 670 条用户评论,实验 2 应用 11 条初始评论种子挖掘出 8 113 条用户评论.因此,在保证评论种子代表性的前提下,增加评论种子数量可以提高用户评论的挖掘效果.通过实验 1(评论种子总数为 23)和实验 2(评论种子总数为 26)的整体挖掘效果可以看出,由于实验 1 的评论种子针对该批数据更具有代表性,因此实验 1 应用较少的评论种子的整体挖掘效果较好.总的来说,即使每个实验的初始评论种子不同,但最终挖掘出体现使用反馈的用户评论数相近,因此,本文提出的针对 APP 软件使用反馈的用户评论挖掘方法是有效的.

5.2.5 体现使用反馈的用户评论挖掘效果对比

我们开发了一个基于贝叶斯的体现使用反馈的 APP 软件用户评论挖掘原型工具(mining user’s comment based on Bayes,简称 MUCBB),使用贝叶斯网络进行用户评论分类,将用户评论划分为“软件未达到的期望”“软件存在的问题”和“软件满足的需求”这 3 种使用反馈类型.为了将 MUCBS 与 MUCBB 进行更好的比较,针对第 5.2.1 节中的 50 072 条用户评论,我们通过表 4 中体现不同使用反馈类型的评价对象和评价观点训练 MUCBB.MUCBS 与 MUCBB 的用户评论挖掘效果对比如图 5 所示.

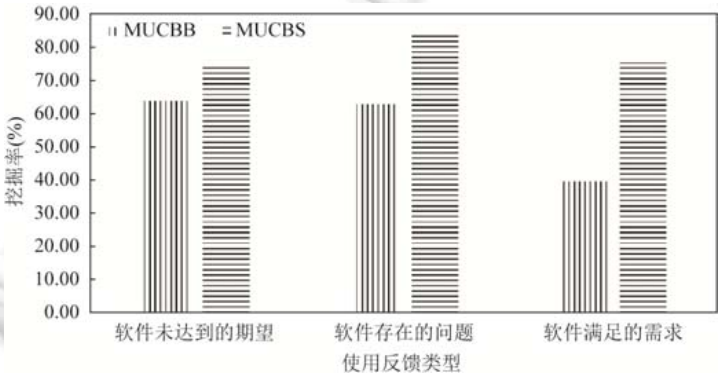


Fig.5 The comparison of overall mining effect of user’s comment reflecting usage feedback

图 5 体现使用反馈的用户评论的整体挖掘效果对比

从图 5 中 MUCBB 的用户评论挖掘效果可以看出,针对体现“软件未达到的期望”和“软件存在的问题”的用户评论挖掘率较高,分别为 63.83%和 62.77%,而体现“软件满足的需求”的用户评论挖掘率则为 39.48%.通过分析发现,由于实验使用表 4 中体现不同使用反馈类型的评论种子对应的评价对象和评价观点训练 MUCBB,“软件未达到的期望”和“软件存在的问题”类别下的训练数据较多,而“软件满足的需求”类别下的训练数据较少,训练数据不平衡导致 MUCBB 的用户评论挖掘效果较差.此外,MUCBB 在训练数据较少的情况下,受训练数据中的评价对象、评价观点的限制,MUCBB 无法挖掘出表达内容相似或句式结构相似的用户评论.例如,针对用户评论“可以/v 下载/v ,/wd 速度/n 挺/d 快/a”,MUCBB 错误地将该用户评论划分为“软件存在的问题” 这一使用反馈类型.而 MUCBS 将该用户评论与评论种子“〈可以+,+很+好, v+wd+d+a,0.5+0+0+0.5,8,软件满足的需求〉”匹配成功,该评论体现“软件满足的需求”这一使用反馈类型.因此,即使在体现不同使用反馈的初始评论种子不平衡的情况下,MUCBS 可以通过自动挖掘体现使用反馈的、具有代表性的新评论种子,在不受特定的评价对象、评价观点限制的情况下,结合句式结构的特点挖掘体现使用反馈的用户评论.但是,由于 MUCBS 需要进行循环挖掘直至挖掘效果收敛,导致 MUCBS 的挖掘效率比 MUCBB 要低.综合分析,与 MUCBB 相比,MUCBS 可以更有效地挖掘体现 APP 软件使用反馈的用户评论.

应用本文方法,可以挖掘出体现 APP 软件使用反馈的用户评论.分析人员可以针对这些挖掘出的体现不同

© 中国科学院软件研究所 http://www.jos.org.cn

使用反馈类型的用户评论进行深入分析,以获取 APP 软件用户评论中有价值的信息.

## 6 总 结

本文将用户评论体现的使用反馈定义为“软件满足的需求”“软件存在的问题”和“软件未达到的期望”这 3 种类型,提出了一种挖掘体现使用反馈的 APP 软件用户评论的方法.为了针对体现使用反馈的用户评论中的核心内容,本文通过构建评价对象和评价观点抽取规则抽取一条用户评论的评价对象和评价观点.在此基础上,定义了评论种子和评论模式,应用评论种子挖掘用户评论库中与之相同或相似的、体现使用反馈的用户评论,综合应用能愿动词、软件简介和情感分析技术构建候选评论模式库,借鉴半监督自学习的思想基于候选评论模式库扩充评论种子库,实现体现 APP 软件使用反馈的用户评论的循环挖掘.通过挖掘体现不同使用反馈的用户评论,软件应用市场可以综合分析体现“软件满足的需求”和“软件存在的问题”的用户评论,给潜在用户提供软件实际使用情况的参考.此外,体现“软件存在的问题”和“软件未达到的期望”的用户评论中包含大量用户关注的软件特征,有助于软件开发人员对此进行维护和改进.

实验结果表明,本文的方法适应网络用户评论数据更新快、数量大的特点,通过半监督自学习的方式,在人工标记数量有限的情况下挖掘体现使用反馈的用户评论.本文提出的用户评论与评论种子的匹配方法的优势在于挖掘体现使用反馈的用户评论时,不受特定的评价对象、评价观点或固定的评论句式结构的限定,将三者综合考虑,从而扩大了挖掘体现相同使用反馈的、核心评价内容或评论句式结构相似的用户评论的范围.由于目前建立的评价对象和评价观点抽取规则不够全面,对表达复杂的用户评论的抽取效果较差,下一步将继续完善抽取表达复杂的用户评论的评价对象和评价观点,并针对评论文本中包含多个分句的情况进行相关研究.此外,网络用户评论中存在大量的网络词汇或者拼写错误,可能影响用户评论的挖掘效果,我们将结合这一情况展开进一步的研究.

**References**:


[1] Pagano D, Brügge B. User involvement in software evolution practice: A case study. In: Proc. of the 2013 Int’l Conf. on Software Engineering. San Francisco: IEEE Press, 2013. 953–962.

[2] Jiang W, Ruan H, Zhang L, Lew P, Jiang J. For user-driven software evolution: Requirements elicitation derived from mining online reviews. In: Tseng VS, ed. Proc. of the Pacific-Asia Conf. on Knowledge Discovery and Data Mining. Cham: Springer-Verlag, 2014. 584–595.

[3] Lu Z, Yang D, Li J. A software evaluation system based on reviews mining. Computer Applications and Software, 2014,31(7):1–4 (in Chinese with English abstract).

[4] Cui J, Yang D, Li J. RERM: A requirement elicitation method based on review mining. Computer Applications and Software, 2015,32(8):28–33 (in Chinese with English abstract).

[5] Panichella S, Di Sorbo A, Guzman E, Visaggio CA, Canfora G, Gall H. Ardoc: App reviews development oriented classifier. In: Proc. of the 24th ACM SIGSOFT Int’l Symp. on Foundations of Software Engineering. Seattle: ACM, 2016. 1023–1027.

[6] Grano G, Di Sorbo A, Mercaldo F, Visaggio CA, Canfora G, Panichella S. Android apps and user feedback: A dataset for software evolution and quality improvement. In: Proc. of the 2nd ACM SIGSOFT Int’l Workshop on App Market Analytics. Washington: ACM, 2017. 8–11.

[7] Guzman E, Maalej W. How do users like this feature? a fine grained sentiment analysis of app reviews. In: Proc. of the 22nd IEEE Int’l Requirements Engineering Conf. (RE). Karlskrona: IEEE, 2014. 153–162.

[8] Keertipati S, Savarimuthu BTR, Licorish SA. Approaches for prioritizing feature improvements extracted from app reviews. In: Proc. of the 20th Int’l Conf. on Evaluation and Assessment in Software Engineering (EASE 2016). Limerick: ACM, 2016. 1–6. [doi: 10.1145/2915970.2916003]

[9] Gao CY, Xu H, Hu JJ, *et al*. Ar-tracker: Track the dynamics of mobile apps via user review mining. In: Proc. of the 2015 IEEE Symp. on Service-oriented System Engineering (SOSE). IEEE, 2015. 284–290. [doi: 10.1109/SOSE.2015.13]


© 中国科学院软件研究所 http://www.jos.org.cn

[10] Palomba F, Salza P, Ciurumelea A, Palomba F, Salza P, Ciurumelea A, Panichella S, Gall H, Ferrucci F, De Lucia A. Recommending and localizing change requests for mobile apps based on user reviews. In: Proc. of the 39th Int'l Conf. on Software Engineering. IEEE Press, 2017. 106–117. [doi: 10.1109/ICSE.2017.18]

[11] Guzman E, El-Haliby M, Bruegge B. Ensemble methods for app review classification: An approach for software evolution (N). In: Proc. of the 30th IEEE/ACM Int'l Conf. on Automated Software Engineering (ASE). IEEE, 2015. 771–776. [doi: 10.1109/ASE.2015.88]

[12] Maalej W, Nabil H. Bug report, feature request, or simply praise? on automatically classifying app reviews. In: Proc. of the 23rd IEEE Int'l Requirements Engineering Conf. (RE). Ottawa: IEEE, 2015. 116–125.

[13] Villarroel L, Bavota G, Russo B, Villarroel L, Bavota G, Russo B, Oliveto R, Di Penta M. Release planning of mobile apps based on user reviews. In: Proc. of the 38th Int'l Conf. on Software Engineering. Austin: ACM, 2016. 14–24. [doi: 10.1145/2884781.2884818]

[14] Gu XD, Kim S. What parts of your apps are loved by users. In: Proc. of the 30th IEEE/ACM Int'l Conf. on Automated Software Engineering (ASE). IEEE, 2015. 760–770. [doi: 10.1109/ASE.2015.57]

[15] Ciurumelea A, Schaufelbühl A, Panichella S, Gall HC. Analyzing reviews and code of mobile apps for better release planning. In: Proc. of the 24th IEEE Int'l Conf. on Software Analysis, Evolution and Reengineering (SANER). Klagenfurta: IEEE, 2017. 91–102.

[16] Jiang W, Zhang L, Dai Y, Jiang J, Wang G. Analyzing helpfulness of online reviews for user requirements elicitation. Chinese Journal of Computers, 2013,36(1):119–131 (in Chhinese with English abstract).

[17] Liu K, Xu LH, Zhao J. Co-extracting opinion targets and opinion words from online reviews based on the word alignment model. IEEE Trans. on Knowledge and Data Engineering, 2015,27(3):636–650.

[18] Ren YY. Extracting product features and determine sentiment orientation from Chinese online reviews [MS. Thesis]. Hefei: University of Science and Technology China, 2015 (in Chinese with English abstract).

[19] Song XL, Wang SG, Li HX. Research on comment target recognition for specific domain products. Journal of Chinese Information Processing, 2010,24(1):89–93 (in Chinese with English abstract).

[20] Wiebe J, Wilson T, Bruce R, Bell M, Martin M. Learning subjective language. Computational Linguistics, 2004,30(3):277–308.

[21] Li QQ. The development and stylistic functions of Chinese negative adverbs [MS. Thesis]. Yinchuan: Ningxia University, 2014 (in Chinese with English abstract).

[22] Mäntylä MV, Novielli N, Lanubile F, Claes M, Kuutila M. Bootstrapping a lexicon for emotional arousal in software engineering. In: Proc of the 14th Int'l Conf. on Mining Software Repositories. IEEE, 2017. 198–202. [doi: 10.1109/MRS.2017.47]

[23] Zhang YS. Study on the Adverbs in Modern Chinese. Shanghai: Xue Lin Publishing House, 2000. 79–80 (in Chinese).

[24] Levenshtein VI. Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady, 1966,10(8):707–710.

[25] Ma QS. The conbined use of verbs. The Study of Language, 1988,(1):18–28 (in Chinese with English abstract).

[26] Jiang Y, Hu TY, Ding JM, Li LY, Wang HT. A method of user's comment mining for app software's quality-in-use. Int Cl: G06F17/30, 201710890944.4, 2018-04-06 (in Chinese).

**附中文参考文献:**

[3] 卢忠浩,杨达,李娟.基于评论挖掘的软件评价系统.计算机应用与软件,2014,31(7):1–4.

[4] 崔建苓,杨达,李娟.RERM:一种基于评论挖掘的需求获取方法.计算机应用与软件,2015,32(8):28–33.

[16] 姜巍,张莉,戴翼,蒋竞,王刚.面向用户需求获取的在线评论有用性分析.计算机学报,2013,36(1):119–131.

[18] 任远远.中文网络评论的产品特征提取及其情感倾向判定[硕士学位论文].合肥:中国科学技术大学,2015.

[19] 宋晓雷,王素格,李红霞.面向特定领域的产品评价对象自动识别研究.中文信息学报,2010,24(1):89–94.

[21] 李倩倩.汉语否定副词的发展演变及其语体功能[硕士学位论文].银川:宁夏大学,2014.

[23] 张谊生.现代汉语副词研究.上海:学林出版社,2000.79–80.

[25] 马庆株.能愿动词的连用.语言研究,1988,(1):18–28.

© 中国科学院软件研究所 http://www.jos.org.cn

[26] 姜瑛,胡甜媛,丁家满,李凌宇,汪海涛.一种针对 APP 软件使用质量的用户评论挖掘方法.Int Cl:G06F17/30,201710890944.4, 2018-04-06.

**胡甜媛**(1994－),女,江苏宿迁人,学士,CCF 学生会员,主要研究领域为软件工程,软件质量保证与测试.

**姜瑛**(1974－),女,博士,教授,博士生导师,CCF 高级会员,主要研究领域为软件质量保证与测试,云计算,大数据分析,软件工程.

www.jos.org.cn
www.jos.org.cn