

基于用户反馈的 APP 软件缺陷识别



段文静 姜 瑛

云南省计算机技术应用重点实验室 昆明 650500

昆明理工大学信息工程与自动化学院 昆明 650500

(526626071@qq.com)

摘 要 当前,APP 软件已被广泛应用,其质量越来越受到关注。高质量的软件的缺陷应尽可能少,然而软件测试并不能发现所有的缺陷,部分缺陷到用户使用阶段才被发现,因此通过分析用户反馈的信息有助于发现软件缺陷。文中提出了基于用户反馈的 APP 软件缺陷识别方法,通过定义 APP 软件缺陷抽取规则挖掘用户反馈中的软件缺陷,并在挖掘软件缺陷的过程中动态更新抽取规则,最后对抽取出的 APP 软件缺陷进行分类及严重程度分析。实验表明,所提方法是有效的,提取含有软件缺陷的 APP 软件用户评论的准确率达 85.19%,缺陷分类准确率达 83.23%。

关键词: APP 软件;用户反馈;缺陷抽取规则;缺陷分类;缺陷严重程度

中图法分类号 TP311.5

Defect Recognition of APP Software Based on User Feedback

DUAN Wen-jing and JIANG Ying

Computer Technology Application Key Lab of Yunnan Province, Kunming 650500, China

Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

Abstract At present, APP software has been widely used, and its quality has been widely concerned. The high quality software defects should be fewer. However, software testing cannot find all defects. Some software defects can not be found until the user uses the software. This paper puts forward a method of software defect recognition based on user feedback. By defining the APP software defect extraction rule, software defects in user feedback are mined. During the mining of APP software's defect, the extraction rule is dynamically updated. Then the classification and severity for the extracted defects are analyzed. The experimental results show that the proposed method is effective, the accuracy of extracting user comments including APP software's defects is 85.19%, and the accuracy of defect classification is 83.23%.

Keywords APP software, User feedback, Defect extraction rules, Defect classification, Defect severity

1 引言

软件开发是一项人为参与的智力活动,势必引入缺陷。IEEE 729-1983 将软件缺陷定义为:从产品内部看,缺陷是软件产品开发或维护过程中存在的错误、毛病等各种问题;从产品外部看,缺陷是系统所需要实现的某种功能的失效或违背^[1]。质量不佳的软件产品不仅会影响公司信誉,还可能产生其他的责任风险,在一些关键的应用领域,如金融、军事和航空航天等,甚至会造成灾难性的后果。

软件一旦失效,将对人类的工作生活造成严重后果。随着软件工程的快速发展,软件缺陷已经不仅局限于代码编写过程中。而软件测试并不能发现所有的软件缺陷,因此用户在使用软件时,仍可能发现大量的软件缺陷。随着 APP 软件的广泛应用,其质量问题也日益受到关注。在 APP 软件用户

针对软件使用情况给出的用户反馈中,通常包含用户评论及评分星级。其中,用户评论是真实体验信息的重要来源,包含了大量来自不同类别用户的不同使用结果反馈。与用户评论相比,用户评分星级包含的信息较少,但对软件的缺陷识别仍具有一定价值。为了从用户反馈中获取 APP 软件的相关缺陷信息,本文提出了基于用户反馈的 APP 软件缺陷识别方法。本文主要贡献如下:(1)提出了基于用户反馈的 APP 软件缺陷抽取规则,用于识别用户反馈中的软件缺陷,在挖掘软件缺陷的过程中不断更新抽取规则;(2)对识别出的 APP 软件缺陷进行分类及严重程度分析。

2 相关工作

用户评论是一种用户反馈,与一般文本相比,用户评论具有海量、简短、低质等特点。传统的文本挖掘方法应用到用户

到稿日期:2019-11-18 返修日期:2020-04-13 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家重点研发计划项目(2018YFB1003904);国家自然科学基金(61462049,61063006,60703116);云南省应用基础研究计划重点项目(2017FA033)

This work was supported by the National Key Research and Development Program of China (2018YFB1003904), National Natural Science Foundation of China (61462049, 61063006, 60703116) and Key Project of Yunnan Applied Basic Research (2017FA033).

通信作者:姜瑛(jy_910@163.com)

评论这类短文本数据时,具有复杂度高、准确率低、噪声大的缺点。针对这一问题,大多研究通过外部链接或知识库来扩展词汇的语义,或者利用后缀树模型构建短语,从而改善短文本稀疏性问题。例如,以 WordNet 或 HowNet 等语义词典为基础丰富词汇信息。Zhang 等提出了一个弱点查找器专家系统,采用基于分词的方法和 Hownet 的相似度算法提取产品特征,对显性特征进行分类,并利用搭配选择的方法对隐性特征进行识别和分类;然后利用情感分析的方法确认句子极性,寻找产品弱点^[2]。Abraham 提出基于产品缺陷识别的文本特征构建体系,将用户评论归属分论坛的属性名作为汽车评论的分类标志,得到若干二元产品评论分类器,并以汽车和电子产品为实现对象检验该方法的准确性^[3]。Zhang 等提出了互联网环境下的手机缺陷识别研究方法,其中利用支持向量机分类算法需要大量的人工标注^[4]。Jiang 等提出在中文互联网环境下运用半监督分类算法识别汽车故障,在缺陷率为 30% 的汽车论坛评论使用半监督 Tri-training 方法,大大减少了人工标记数量,但该方法的分类准确率不够高^[5]。Xia 等提出一种基于评论长度的特征提取方法,设计了评论样本自动标注方法,并构建了评论有效性分类模型,以改进分类效率,但所构建的通用领域词概念模型与 APP 软件特征不符,因此将其运用到 APP 软件用户反馈中分类不准确^[6]。Pagano 分析了 APP 软件用户评论中的用户对该 APP 软件的正反态度,并以此作为开发人员改进用户对 APP 软件需求的依据^[7]。Zhao 等采用 Canopy+K-means 方法对海淘 APP 用户在线评论数据的属性特征进行聚类,确定用户满意度评价维度,进而应用 CNN-SVM 情感分析模型得到各维度的用户满意度评分^[8]。Fan 等在情感分析的基础上构建了基于网络用户评论情感计算的产品用户痛点分析模型,从用户关注程度和用户情感两方面来测度用户痛点,提出用户痛点指数计算公式,在一定程度上克服了传统用户评分的主观性偏差^[9]。Hu 等针对不同类型的 APP 软件使用反馈,提出了评价对象和评论观点抽取规则,将用户评论体现的 APP 软件使用反馈总结为软件满足的需求、软件存在的问题和软件未达到的期望 3 种类型,应用半监督自学习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^[10]。

在软件缺陷分类方面,缺陷分类的方法繁多,不同的方法因目的不同具有不同的分类过程、复杂程度、准确性及应用领域^[11]。经典的软件缺陷分类方法有正交缺陷分类法(Orthogonal Defect Classification, ODC)、软件异常分类标准(IEEE standard classification for Anomalies 1044-2009)、Thayer 分类法及 Roger 分类法等。正交缺陷分类法由 IBM 公司提出,提供了一套用于捕获缺陷数据关键特性的方案,并就如何对分类的缺陷数据进行分析给予了指导^[12]。软件异常分类标准为出现在项目中、产品中或者系统生命周期内的软件异常提供了一种统一的分类方法,定义了缺陷分类方法和失效分类方法和大量可供选择的属性类型及其参考取值集合。该分类标准灵活度高,可针对实际需求进行适当的裁剪或扩展^[13]。Thayer 分类法根据错误的性质进行分类,分类的信息源自软件测试和使用中填写和反馈的问题报告,不局限于软件本身的错误,还包括系统软件错误、人员操作错误等^[14]。Roger 分类法根据缺陷引入原因将其分为 12 种类

型,但所提供的信息十分笼统^[15]。Chen 提出了一种基于支持向量机和主题模型的评论分析方法 RASL(Review Analysis method based on SVM and LDA),从移动应用软件的用户抱怨评论中提取特征,并构建用户评论的多标签分类模型,进行用户评论分类,然后对每个类别下的评论进行主题挖掘,但主题分类结果较多^[16]。

上述研究存在如下不足:

(1)当前基于用户反馈的缺陷识别大多面向某一产品或基于英文语境。由于 APP 软件与一般产品相比具有抽象性、非可视性的特点,通用领域词差异较大,因此,适用于一般产品的用户反馈缺陷识别方法难以完全适用于 APP 软件缺陷识别。基于 APP 用户反馈的软件缺陷研究多是从用户需求、满意度等角度提及。

(2)正交缺陷分类法、软件异常分类标准和 Roger 分类法是基于软件过程的分类方法,Thayer 分类法的数据源是问题报告。以上方法与由于分类对象与角度的不同,运用到用户反馈的缺陷分类和分析中范围过大,因此不适用。

本文针对以上问题,基于 APP 软件的用户反馈,在 APP 软件缺陷抽取规则、缺陷识别、缺陷分类及严重程度分析等方面进行了研究。

3 挖掘用户反馈中的软件缺陷

为了识别和分析 APP 软件用户反馈中的软件缺陷,本文提出了基于用户反馈的 APP 软件缺陷抽取规则。首先筛选出含有缺陷的用户评论;其次抽取 APP 用户评论中的软件缺陷;最后判断缺陷类别及严重程度,为 APP 软件的后期维护人员提供可靠、客观的分析结果。基于用户反馈的 APP 软件缺陷识别流程如图 1 所示。

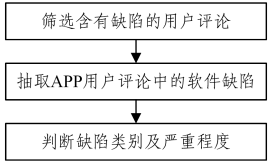


图 1 基于用户反馈的 APP 软件缺陷识别流程
Fig. 1 Process of defect recognition of app software based on user feedback

3.1 筛选含有缺陷的用户评论

用户反馈是在用户使用 APP 软件的过程中给出的关于使用情况的反馈,包括用户评论和用户评分星级。用户评论是用户真实体验信息的重要来源,针对不同产品的用户评论具有不同的特点。APP 软件用户评论具有评论文本简短、主观性强、随意性强及口语化现象普遍等特点。APP 软件相对商品用户评论,包含较少的特征词,不同的使用人群和使用平台可能因其特定的身份属性和设备给出不同的使用结果。因此,从大量主观的用户评论中得到相对客观的分析结果是有难度的。

通过分析大量 APP 用户反馈,我们发现在用户评分星级中,1—3 星的评论大概率地含有软件缺陷。含有缺陷的评论包含大量通用的缺陷句型,例如,“为什么”后常常紧跟软件缺陷的描述,“如果……就好了”中包含用户对软件提出的建议。基于大量 APP 软件用户评论,本文总结了相关的缺陷句型,如表 1 所列。

表 1 缺陷句型
Table 1 Defect sentences

序号	缺陷句型	说明
1	为什么……/为啥……/ 怎么……/……怎么回事	用户对软件的疑惑,其后常紧跟 软件缺陷的描述
2	没有……	通常指某缺陷对象的缺失
3	无(没)法……/不能……/ 不可以(不行)……	通常指某缺陷对象无法运行
4	希望……/建议……/ 如果……就好了	包含用户对软件提出的建议
5	只是……/但是……	表转折,其后常常紧跟软件缺陷 的描述
6	总是……	用户表述某软件缺陷的频繁发生
7	连……都……	描述某缺陷对象的运行错误
8	……不了……	通常指某软件功能的失效

此外,在含有缺陷的评论中包含大量的缺陷词,如崩溃、闪退、故障、黑屏、速度慢等。因此,为了识别 APP 软件缺陷,本文采用缺陷词、缺陷句型与评分星级相结合的方式,从用户反馈中识别含有软件缺陷的评论。

首先基于表 1 中的缺陷句型和部分缺陷词,筛选出含有缺陷的用户评论;再从剩余评论中筛选出用户评分星级为 1—3 星的用户评论,作为含有缺陷的 APP 用户评论。具体步骤如图 2 所示。

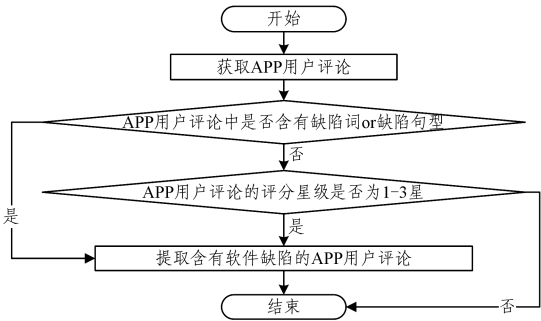


图 2 筛选含有软件缺陷的 APP 用户评论的流程

Fig. 2 Process to filter APP user comments including software defects

3.2 抽取 APP 用户评论中的软件缺陷

为了抽取 APP 软件用户评论中的软件缺陷,需要在含有

缺陷的用户评论中准确定位评论对象及其评论内容。本文基于 3.1 节筛选出的含有缺陷的 APP 用户评论,建立了 APP 用户评论软件缺陷抽取规则,用于识别用户评论中的缺陷对象和缺陷表现。其中,我们将 APP 用户评论中,对软件欠缺或不完备的评论目标称为“缺陷对象”,对某一个缺陷对象的表现描述称为“缺陷表现”。具体步骤如图 3 所示。

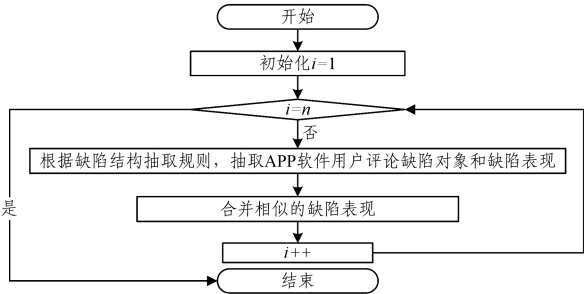


图 3 识别 APP 用户评论中的软件缺陷的流程

Fig. 3 Process of software defects recognition in APP user comments

3.2.1 建立缺陷结构

缺陷结构是识别 APP 软件用户评论中软件缺陷的依据,3.1 节缺陷句型在不同的 APP 用户评论中通常具有相似的表述,具体表现为在句型关键词附近具有相近的词性组合,可用于识别用户评论中的缺陷对象和缺陷表现。因此,缺陷结构由句型关键词与词性组合构成。

研究发现,一些句型更为贴近缺陷对象和缺陷表现,比如“这软件连头像都换不了”,符合表 2 中的规则 2 和规则 5,我们希望抽取的目标是缺陷对象“头像”和缺陷表现“换不了”,显然表 2 中的结构 2.2“n+d+v+‘不了’”更贴近我们的目标。因此,为了得到更好的匹配效果,本文定义了句型优先级和结构优先级。句型优先级是在抽取过程中,优先抽取优先级为 1 的句型。结构优先级是优先匹配了优先级为 1 的句型后为缺陷结构定义的优先抽取规则。初始的缺陷结构可分为十大类,38 种缺陷结构,部分缺陷结构如表 2 所列(全部缺陷结构请参见首码 OSID 码)。

表 2 缺陷句型结构及其优先级

Table 2 Sentence structure and priority of Defect

序号	句型规则	句型规则优先级	缺陷结构	结构优先级
1.1	要是……就好了	优先级 1	n+‘要是’,……,‘就好了’	
1.2	如果……就好了		n+‘如果’,……,‘就好了’	
2.1	……不了……	优先级 1	v+‘不了’+n	2.1=2.2=2.3=2.4>2.5
2.2			n+d+v+‘不了’	
2.3			n+v+‘不了’+n	
2.4			vn+d+v+‘不了’	
2.5			v+‘不了’	
3.1	……不能……	优先级 1	‘不能’+v+n	3.3=3.4=3.5>3.1=3.2>3.6
3.2			n+n+‘不能’+v	
3.3			n+n+‘不能’+v+n	
3.4			‘不能’+v+n+n	
3.5			v+n+‘不能’+v	
3.6			‘不能’+v	
4.1	……有问题	优先级 1	n+‘有问题’	4.2>4.1
4.2			n+n+‘有问题’	
5.1	连……都……	优先级 2	‘连’+n+‘都’+v+d+v	
5.2			‘连’+n+‘都’+v+d+y	

3.2.2 识别 APP 用户评论中的软件缺陷

在定义缺陷结构之后,可以根据缺陷结构在含有缺陷的 APP 软件用户评论中提取缺陷对象和缺陷表现。在 APP 用户评论中,对于同一缺陷对象,用户描述的缺陷表现往往是不同的。例如在 QQ 中,同一缺陷内容“头像”的缺陷表现包括“换不了”“不能换”“加载不出来”。其中“换不了”和“不能换”的缺陷表现是相似的,所以需要将相似的缺陷表现进行合并。本文利用知网词典,对同一缺陷对象下的不同缺陷表现进行相似度计算,将相似度高(大于 0.6)的缺陷表现进行合并。

3.2.3 更新 APP 软件缺陷结构的抽取规则

表 2 的缺陷结构是我们根据 APP 用户评论的缺陷句型特征总结出来的,只能识别一部分与缺陷句型相关的软件缺陷。通过分析大量 APP 软件用户评论,我们总结了部分缺陷词,如表 3 所列。其中,字母 n 代表名词,v 代表动词,d 代表副词,vn 代表动名词,q 代表量词,y 代表语气词。

表 3 初始缺陷词
Table 3 Initial defect words

序号	词性组合	缺陷词
1	v/vi	失败、崩溃、耗电
2	n	错误、故障、卡
3	a	差、臃肿
4	v+n	有问题、无法、耗流量、死机
5	d+v	不兼容、不匹配、不行
6	n+n	垃圾玩意、网页错误
7	v+v	闪退
8	n+a	速度慢
9	v+a	下载慢
10	n+d+a	速度太慢
11	v+d+a	下载太慢
12	v+d+v	关不掉
13	v+d+y	发不了
14	d+v+v	不能改

由于 APP 用户评论数据量十分庞大,缺陷词数量较多,人工枚举所有缺陷词并建立缺陷结构显然是不可行的,因此需要给出缺陷词与缺陷结构更新的方法,即在挖掘用户评论的过程中,根据初始缺陷词表自动识别出新的缺陷词,并将其定义为新的缺陷结构,更新到已有结构中。

(1)识别新缺陷词

缺陷词位于含有缺陷的用户评论中,存在一定的表达规律:1)缺陷词有相似的词性组合规律;2)部分缺陷词中包含否定词或为负面词;3)缺陷词往往位于含有缺陷的用户评论的句尾,如“更新之后闪退!”“动态权限设置,加好友设置打不开”。其中,“闪退”和“打不开”是典型的 APP 软件缺陷词。

本文的缺陷词抽取规则是根据表 3 中的缺陷词与初始缺陷词的相似度、位于 APP 用户评论中的位置和缺陷词词性组合来定义的。具体步骤如下。

Step1 提取第 i 条评论中的所有词性组合,放入 $DefectWordseed$ 集合中,初始化缺陷词系数 $coefficient=1$;

Step2 计算 $DefectWordseed$ 中元素与人工词性表中同词性的相似度 $Similarity$;

Step3 计算 $DefectWordseed$ 中元素与句尾(标点符号和句尾)的距离,进行数据归一化处理得到 $Distance[0,1]$;

Step4 判断 $DefectWordseed$ 中元素是否是否定词或负面词,如果是 $coefficient\times 1.2$;

Step5 计算 $DefectWordseed$ 中元素缺陷词的可能性:

$Possibility=Coefficient\times (1-Distance)\times Similarity$ (1)

若该值大于阈值 0.6,且位于含有缺陷的评论内,则定义该词为新缺陷词。

Step3 中为了将缺陷词位于 APP 缺陷评论中的位置转换为式(1)中的权重,本文计算了缺陷词与句尾的距离,对其进行离差标准归一化处理,如式(2)所示:

$Distance=\frac{(Distance-min)}{(max-min)}$ (2)

其中, $Distance$ 表示缺陷词到标点符号和句尾的距离, max 表示样本数据的最大值, min 表示样本数据的最小值。

将距离数值进行离差归一化处理后,得到的 $Distance$ 的取值范围为 $[0,1)$ 。 $Distance$ 的数值越接近 0 表示缺陷词距离句尾越近。由于在 APP 软件用户评论中缺陷词往往位于含有缺陷的用户评论句尾,本文使用 $1-Distance(0,1]$ 代表距离权重。

(2)缺陷结构的更新

在找到新的缺陷词后,需要在此基础上建立新的缺陷结构,并将其补充到初始的缺陷结构中。在 APP 用户评论中,缺陷词是对缺陷对象的描述,与缺陷表现相似,存在于缺陷对象附近。为了将缺陷词转换为缺陷结构,需要在含有缺陷的 APP 用户评论内,先定位缺陷对象和缺陷词,然后将其转换为词性组合的结构,具体步骤如下。

Step1 从 APP 用户评论的缺陷结构中提取出缺陷对象,在其周围寻找缺陷词,选取缺陷对象和缺陷词的距离小于 3 的词性结构作为候选缺陷结构。

Step2 在含有缺陷的 APP 用户评论中,计算候选缺陷结构在用户评论中的共现度。

Step3 将共现度高的候选缺陷结构补充到缺陷结构中。

实现缺陷词及缺陷结构的不断更新,将有助于不断提高识别含有缺陷的 APP 软件用户评论的准确率。

3.3 判断缺陷类别及严重程度

对 APP 用户评论中的软件缺陷进行分类的目的在于分析 APP 用户评论中软件缺陷的严重程度。分析用户在使用过程中遇到的软件缺陷的严重程度,可以帮助维护人员了解 APP 用户反馈的建议和软件缺陷,提供合理的软件缺陷解决顺序。

传统的软件缺陷分类是从软件过程的角度对软件缺陷进行分类,是与软件的生命周期的有机结合。与传统的软件缺陷分类不同,针对 APP 用户评论的软件缺陷分类是在产品上市后,面向 APP 用户评论的,需要针对 APP 用户评论的具体内容进行分析。因此,缺陷分类应从用户评论的角度出发,包含所有用户评论中存在的软件缺陷类型。基于 APP 用户评论的软件缺陷分类应准确地对用户评论内发现的软件缺陷进行分类,且软件缺陷的分类之间没有重叠。本文根据 APP 用户评论中的具体内容,将 APP 用户评论内的软件缺陷分为 7 类,即安装缺陷、功能性缺陷、兼容性缺陷、资源性缺陷、可靠性缺陷、有效性缺陷和用户建议。APP 软件缺陷的严重程度根据上述软件缺陷的类型进行分级,具体如表 4 所列。

本文在进行软件缺陷分类时发现,属于缺陷类型 1,3,4,5,6 的 APP 用户评论中包含一定的关键词,而功能性缺陷和用户建议缺陷也有相应的句型表达方式。因此基于表 4 的缺陷类型,对含有缺陷的 APP 软件评论进行类型划分。具体划

分规则如表 5 所列。

表 4 软件缺陷类型及严重程度
Table 4 Types and severity of defects

序号	缺陷类型	描述	严重程度
1	安装缺陷	APP 软件在安装阶段出现问题:如安装不上等	1
2	功能性缺陷	在具体功能上的缺陷	4
3	兼容性缺陷	在不同手机上由于兼容性问题造成的图片显示不了、页面布局出现问题	3
4	资源性缺陷	内存占用较多,耗电,耗流量	6
5	可靠性缺陷	APP 软件在使用时崩溃,闪退,黑屏,无响应	2
6	有效性缺陷	APP 软件在使用时卡顿	5
7	用户建议	用户对 APP 软件提出的建议	7

表 5 根据特征词和句型划分缺陷类型

Table 5 Classification of defect types according to feature words and sentence patterns

缺陷类型	缺陷句型和特征词
安装缺陷	安装
功能性缺陷	……不了……/……不能……/连……都/……不可以……
兼容性缺陷	不兼容,不匹配
资源性缺陷	内存、耗(费)电、耗(费)流量、臃肿、存储
可靠性缺陷	崩溃、闪退、黑屏、无(没)响应、死机、停止运行、不(没)经过同意、没(无)反应
有效性缺陷	卡、速度(慢)
用户建议	如果……就好了/要是……就好了/希望……/……望改进/建议……

软件缺陷类型中的功能性缺陷没有具体的特征词,比较难以判断。但是在缺陷句型中,部分句型是与功能性缺陷直接相关的,例如“……不了……”“……不能……”等。这些缺陷结构描述的是 APP 某一功能的运行问题。因缺陷句型的局限、覆盖率不足等问题,大部分的功能性缺陷仍难以分类。

表 7 APP 用户评论软件缺陷抽取规则的实验结果

Table 7 Experimental results of defect extraction rules for APP user comments

分词前	分词后	缺陷结构	缺陷对象	缺陷表现
内存要是少就好了	内存/n要是/c少/d就/d好/a了/y	内存/n*要是'少/d*就好了'	内存	
为什么 QQ 发不了信息了	为什么/ryv QQ/n发/v不/d了/u!e信息/n了/y	QQ/n发/v*不了'信息/n	QQ 信息	发不了
为什么录音权限老是打不开呢	为什么/ryv 录音/vn权限/n老/a是/vshi打/v不/d开/v呢/y	录音/vn权限/n*老是'打/v不/d开/v	录音权限	打不开
权限都开了为什么还是不能发语音	权限/n都/d开/v了/y为什么/ryv还/d是/vshi不/d能/v发/v语音/n	'不能'发/v语音/n	语音	不能发

4.2 实验一

根据 3.1 节的方法,在 56102 条用户评论中筛选出含有缺陷的用户评论,结果如表 8 所列。可以看出,针对不同的 APP 评论,含有缺陷的评论在其中所占比例不同。在 56102

因此,本文将软件简介与评论内容进行对比来分析功能性缺陷。软件简介是对 APP 的简要介绍,其中包含 APP 官方给出的软件功能介绍。通过比较 APP 简介中的功能点与用户评论,可以弥补利用单一的缺陷结构对功能性缺陷分类不准确的问题。

4 实验与分析

4.1 实验数据来源

为了验证本文方法的有效性,从应用宝¹⁾上获取真实的用户评论作为数据集进行实验,随机选取 10 个不同的 APP 软件共计 56102 条用户评论,并对所有用户评论是否包含缺陷内容进行了人工标注。具体内容如表 6 所列。

表 6 APP 名称与用户评论条数

Table 6 APP name and number of user comments

APP 软件名称	版本号	用户评论条数
QQ	V7.8.2	14652
新浪新闻	V6.5.8	4721
懒人听书	V6.1.6	2673
QQ 音乐	V7.9.1.7	6870
QQ 空间	V7.5.1.288	5865
欢乐斗地主	V5.83.002	4632
QQ 邮箱	V5.3.6	2526
京东	V6.5.2	9867
蘑菇街	V10.1.1.8315	1523
百度新闻	V6.7.0.2	2773

本文将 ICTCLAS 作为数据预处理的工具,完成分词及词性标注,并根据 3.2 节定义的 APP 用户评论软件缺陷抽取规则抽取 APP 软件用户评论的缺陷结构和其中的缺陷对象、缺陷表现。部分数据的处理结果如表 7 所列。

表 8 提取含有缺陷的用户评论实验结果表

Table 8 Experimental results of user comments extraction with defects

APP 软件名称	本文方法识别的含有缺陷的用户评论条数	人工标注的含有缺陷的用户评论条数	本文方法识别的含有缺陷的用户评论所占比例/%	人工标注的含有缺陷的用户评论所占比例/%	准确率/%
QQ	7429	7217	50.70	49.26	88.33
新浪新闻	753	681	15.95	14.42	80.62
懒人听书	133	128	4.98	4.79	79.69
QQ 音乐	2456	2290	35.75	33.33	85.81
QQ 空间	1974	1883	33.66	32.11	85.81
欢乐斗地主	702	689	15.16	14.87	89.70
QQ 邮箱	632	586	25.02	23.20	76.11
京东	1964	1830	19.90	18.55	80.82
蘑菇街	216	199	14.18	13.07	81.41
百度新闻	392	368	14.14	13.27	85.87

¹⁾ <https://sj.qq.com/>

4.3 实验二

为了分析含有缺陷的 APP 软件用户评论与评分星级的关系,本文分析了 15871 条含有缺陷的 APP 软件用户评论中各个星级内的软件缺陷的分布情况,结果如表 9 所列。

表 9 含有缺陷的用户评论与评分星级关系

Table 9 Relationship between user comments with defects and ratings

APP 软件名称	人工标注不同评分星级中的 含有缺陷的用户评论条数					1—3 星中含有缺陷的 用户评论条数占比/%
	1	2	3	4	5	
QQ	2479	1976	1958	389	415	89.00
新浪新闻	197	134	179	25	146	74.89
懒人听书	67	21	16	8	16	81.25
QQ 音乐	970	480	512	201	127	85.68
QQ 空间	865	379	241	221	177	78.86
欢乐斗地主	146	371	42	73	57	81.32
QQ 邮箱	273	78	104	89	42	77.65
京东	629	677	198	67	259	82.19
蘑菇街	24	86	54	17	18	82.12
百度新闻	138	58	126	11	35	87.50

从表 9 可以看出,评分星级为 1—3 的用户评论中含有较多的与软件缺陷相关的内容。10 个 APP 软件 1—3 星中含有缺陷的用户评论条数平均占比 84.92%,说明大量有关软件缺陷的描述集中在 1—3 星的 APP 软件用户评论中。因此,在提取含有缺陷的 APP 软件用户评论时,应重点关注 1—3 星的 APP 软件用户评论。

为了验证表 1 缺陷句型的有效性,本文从上述 56102 条用户评论中提取 16651 条含有缺陷的 APP 用户评论并进行分析,得出了不同句型在含有缺陷的用户评论中的分布情况,结果如表 10 所列。实验表明,在含有软件缺陷 APP 用户评论中,句型“为什么……”“没有……”“不……”“……不了……”所占比例较大。由于不同缺陷句型在具体用户评论

中互有重叠,经过整体去重,在 16651 条含有缺陷的用户评论中,共有 8216 条含有缺陷的用户评论被识别,准确率为 49.34%,表明缺陷句型可以筛选部分含有缺陷的 APP 用户评论。但是,缺陷句型不能作为判断 APP 用户评论中是否含有缺陷的唯一方法,还需要在将来的研究中进一步扩充相关句型。

表 10 验证缺陷句型有效性的实验结果

Table 10 Experimental results of verifying validity of defect sentence

序号	缺陷句型	含有本类句型的 用户评论条数	本类句型在含有缺陷的 用户评论所占比例/%
1	为什么……	1783	10.71
	为啥……	193	1.16
	怎么……	1169	7.02
	……怎么回事	131	0.78
2	没有……	1800	10.81
	无(没)法……	319	1.92
3	不能……	1764	10.59
	不可以(不行)……	367	2.20
	希望……	1231	7.39
4	建议……	188	1.13
	如果……就好了	390	2.34
5	只是……	176	1.06
	但是……	602	3.62
6	总是……	209	1.26
7	连……都……	336	2.02
8	……不了	1938	11.64

4.4 实验三

为了验证缺陷分类方法的有效性,本文用 3.3 节的方法对 15871 条含有缺陷的 APP 软件用户评论进行分类,分类结果如表 11 所列。其中 P 代表使用本文方法分类的相关 APP 软件用户评论条数,M 代表人工标注的相关 APP 软件用户评论条数。

表 11 APP 软件缺陷分类结果

Table 11 Classification results of APP software defects

APP 软件 名称	含有相关软件缺陷的用户评论条数															
	安装缺陷		功能性缺陷		兼容性缺陷		资源性缺陷		可靠性缺陷		有效性缺陷		用户建议			
	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>	<i>P</i>	<i>M</i>
QQ	267	251	3579	3642	451	427	296	274	375	405	468	437	862	871		
新浪新闻	44	38	332	369	42	31	26	29	34	41	15	16	27	27		
懒人听书	6	6	35	37	4	4	21	24	1	0	18	18	14	14		
QQ 音乐	19	19	1211	1345	8	8	65	69	53	55	63	52	428	431		
QQ 空间	2	2	1427	1325	23	25	53	51	1	1	9	8	107	103		
欢乐斗地主 (腾讯)	13	13	163	172	2	2	74	71	2	3	86	74	124	137		
QQ 邮箱	5	5	231	246	3	3	11	10	15	17	24	21	197	205		
京东	17	18	1005	1121	4	4	204	207	1	1	45	37	316	312		
蘑菇街	1	1	121	134	2	2	31	29	2	1	21	17	16	13		
百度新闻	2	2	156	162	0	0	17	21	14	13	58	51	49	47		

如表 11 所列,使用本文方法对 15871 条含有缺陷的 APP 软件用户评论进行分类,有 13418 条评论被成功分类,缺陷分类准确率为 83.23%。但是,在某些含有缺陷的 APP 软件用户评论中,对缺陷对象与缺陷表现的描述简单、模糊,从而导致被成功划分缺陷的类型。在分析软件缺陷在各个缺陷类别的分布情况后,可以看出,在 APP 软件用户评论中功能性缺陷被大量提及,用户关注度最高。其次,资源性缺陷和

有效性缺陷也是用户较关注的软件缺陷。用户在使用 APP 时,对软件功能、软件闪退、卡顿和软件的资源消耗量的要求较高。因此在 APP 软件开发时,在保证软件质量的前提下,需着重关注以上问题。

结束语 本文提出的基于用户反馈的 APP 软件缺陷识别方法,基于 APP 用户反馈,考虑了用户反馈中软件缺陷的存在规律,通过定义 APP 软件缺陷抽取规则挖掘用户反馈中

的软件缺陷。由于 APP 用户评论数据量大,本文在挖掘缺陷的过程中提出了缺陷结构与缺陷词的自动更新方法,最后根据 APP 用户反馈中软件缺陷的特点,对提取出的 APP 软件缺陷进行分类及严重程度分析。实验表明,本文方法是有效的;但其仍存在软件缺陷识别错误和分类不准确等问题,下一步我们将对比本文方法与机器学习的相关方法,并继续针对缺陷分析及相关规则的改进展开研究。

参 考 文 献

[1] IEEE Std 729-1983[S]. Standard Glossary of Software Engineering Terminology. IEEE,1990.

[2] ZHANG W,XU H,WAN W. Weakness Finder: Find product weakness from Chinese reviews by using aspects based sentiment analysis[J]. Expert Systems with Applications, 2012, 39(11):10283-10291.

[3] ABRAHAMS A S,JIAO J,WANG G A, et al. Vehicle defect discovery from social media[J]. Decision Support Systems, 2012,54(1):87-97.

[4] ZHANG S,WU J Y,FAN W G, et al. Defect Discovery of Phones Based on Social Media Analytics[J]. Computer Integrated Manufacturing Systems,2016,22(9):2264-2273.

[5] JIANG C Q,WANG Q L,LIU S X, et al. A Semi-Supervised Learning Method for Vehicle Defect Recognition in Chinese Social Media Environment[C]//The 16th China Management Science Annual Conference, 2014.

[6] XIA H S,YANG P,XIONG G. The Classification Model for Online Reviews' Effectiveness Based on Feature Extraction Improvement[J]. Scientific and Technical Information, 2015, 5(34):493-500.

[7] PAGANO D,MAALEJW. User feedback in the appstore:An empirical study[C]// 2013 21st IEEE International Requirements Engineering Conference(RE). IEEE,2013:125-134.

[8] ZHAO Y,LI Q Q,CHEN Y H, et al. Research on overseas online shopping APP user satisfaction based on online comment sentiment analysis[J]. Data Analysis and Knowledge Discovery, 2018,2(11):19-27.

[9] FAN W H,XU J. Users' Pain-point Analysis Based on the Sen-

timent Calculation of Web Users' Comments: Taking Cell Phone Comments as Examples[J]. Information studies: Theory & Application,2018,41(1):98-103.

[10] HU T Y,JIANG Y. Mining of User's Comments Reflecting Usage Feedback for APP Software[J]. Journal of Software, 2019,30(10):3168-3185.

[11] PLOSKI J,ROHR M,SCHWENKENBERG P. Research issues in software fault categorization[J]. ACM SIGSOFT Software Engineering Notes,2007,32(6):1-8.

[12] CHILLAREGE R,BHANDARI I S,CHAAR J K, et al. Orthogonal defect classification a concept for in-process measurements[J]. IEEE Transactions on Software Engineering,1992,18(11): 943-956.

[13] IEEE Std 1044-2009[S]. Standard Classification for Anomalies, IEEE,2009.

[14] NIE L,LIU M R. Research on Sofetware Defects Classification[J]. Application Research of Computers,2004,1(6):84-86.

[15] PRESSMAN R S. Software Engineering:a Practitioner's Approach(5th)[M]. Thomas Casson,2001:209-212.

[16] CHEN Q,ZHANG L,JIANG J, et al. Review Analysis Method Based on Support Vector Machine and Latent Dirichlet Allocation[J]. Journal of Software,2019,30(5):349-362.



DUAN Wen-jing, born in 1992, post-graduate,is a member of China Computer Federation. Her main research interests include software engineering and so on.



JIANG Ying, born in 1974, Ph.D, professor, Ph. D supervisor, is a senior member of China Computer Federation. Her main research interests include software quality assurance and testing, cloud computing, big data analysis and intelligent software engineering.