计算机科学
COMPUTER SCIENCE

用户数据驱动的App消退功能研究

贾经冬, 侯鑫, 王哲, 黄坚

引用本文

贾经冬, 侯鑫, 王哲, 黄坚. 用户数据驱动的App消退功能研究[J]. 计算机科学, 2026, 53(1): 262-270.

JIA Jingdong, HOU Xin, WANG Zhe, HUANG Jian. Research on User Data-driven App Fading Functions [J]. Computer Science, 2026, 53(1): 262-270.

相似文章推荐（请使用火狐或 IE 浏览器查看文章）

Similar articles recommended (Please use Firefox or IE to view the article)

基于可信平台控制模块的信任评估系统研究

Study on Trust Evaluation System Based on Trusted Platform Control Module

计算机科学, 2024, 51(11A): 240200109-6. https://doi.org/10.11896/jsjkx.240200109

融入类别标签和主题信息的用户兴趣识别方法

User Interest Recognition Method Incorporating Category Labels and Topic Information

计算机科学, 2024, 51(6A): 230500169-8. https://doi.org/10.11896/jsjkx.230500169

一种结合代码片段和混合主题模型的软件数据聚类方法

Software Data Clustering Method Combining Code Snippets and Hybrid Topic Models

计算机科学, 2024, 51(6): 44-51. https://doi.org/10.11896/jsjkx.230300091

接诉即办智能派单业务调度算法研究

Study on Scheduling Algorithm of Intelligent Order Dispatching

计算机科学, 2023, 50(11A): 230300029-7. https://doi.org/10.11896/jsjkx.230300029

开源软件中社区文档应用与维护的实证研究

Empirical Study on Application and Maintenance of OSS Community Profile Documentation

计算机科学, 2023, 50(6A): 220600221-8. https://doi.org/10.11896/jsjkx.220600221

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.250100070

用户数据驱动的 App 消退功能研究

贾经冬　侯　鑫　王　哲　黄　坚
北京航空航天大学软件学院　北京 100191

摘　要　为有效促进 App 功能迭代,现有大量研究通过挖掘用户评论来改善或增加新功能以促进版本更新,但忽视了从用户评论中识别应该消退的功能。针对此问题,提出了用户数据驱动的 App 消退功能分析方法。首先从应用市场采集用户评论,构建关键字模板过滤出含消退功能的评论,应用语法范式从中挖掘功能短语,并训练分类器识别功能短语以提取出待研究的消退功能,从而构建消退功能数据集。根据版本更新日志和用户评论回溯找到消退功能的生命周期。然后进行消退功能生命周期的用户评论分析。基于文本情感分析,提出字数权重阈值法对虚假评分进行检测和更正,运用 BERT 进行评论文本分类,提出 BERTopic-Corex 主题模型产生主题词,结合之前的分析结果和评论字数识别出关键用户评论,实现了从用户评论中有效分析和识别消退功能。实验结果和实例证明了所提方法的可行性和有效性。

关键词:消退功能;用户评论;评论分类;虚假评分;主题模型
中图分类号　TP391

Research on User Data-driven App Fading Functions

JIA Jingdong,HOU Xin,WANG Zhe and HUANG Jian
School of Software,Beihang University,Beijing 100191,China

Abstract　In order to effectively promote App function iteration,most existing studies generally focus on improving existing functions or adding new functions to promote version update by mining user requirements in user reviews,while neglecting to identify functions that should be eliminated from user reviews. To address the issue,an analysis method about user data-driven App fading functions is proposed. User reviews from App market are collected. Keyword templates are built to filter out reviews that contain fading functions. From these reviews,function phrases are mined by syntax paradigms. A classifier is trained by these phrases to identify fading functions to be studied,so the dataset of fading functions is built. Lifecycle of a fading function is found based on version update log and user reviews backtracking. Then,user reviews for the lifecycle of fading functions are analyzed. A word weight threshold method is proposed to detect and correct false ratings based on text sentiment analysis. BERT algorithm is used to classify the review data. BERTopic-Corex topic model is proposed to generate theme words of reviews. Key user reviews are identified based on the previous analysis results and the word count of reviews. Thus,fading functions can be effectively identified and analyzed from user reviews. Experimental results and examples prove the feasibility and effectiveness of the proposed method.

Keywords　Fading function,User review,Review classification,Fake rating,Topic model

1　引言

App 软件已经成为人们日常生活中不可或缺的工具,App 的定期更新是一种保持应用程序吸引力的必要手段。App 用户评论直接或间接地反映了用户意图,其包含着丰富的有价值的信息,能帮助软件开发者理解用户的需求,设计和改进软件产品[1]。目前大量的研究通过分析用户评论来挖掘用户需求,从而更好地满足用户期望[2]。Wang 等[3]重点挖掘待添加功能、待改进功能以及性能、可用性、可靠性 5 类需求。Hu 等[4]分析了评论中“软件满足的需求”“软件存在的问题”和“软件未达到的期望”3 种情况,基于后两者给出 App 改善建议。也有研究[5-6]重点挖掘 App 的非功能需求,如可靠性、稳定性等。这些研究更多地关注挖掘 App 中从用户角度需要新增和改善的功能,以更好地满足用户需求,提高用户体验[7]。

然而,用户关注的功能被移除或不受欢迎的功能未被移除,同样会影响 App 用户的使用体验。如果在 App 演化中只新增功能而不消退功能,那么 App 的功能将变得愈加复杂。研究表明,功能过多的复杂 App 会有碍于用户对应用的理解,进而影响用户对 App 的使用体验[8]。此外,一些花哨的

到稿日期:2025-01-13　返修日期:2025-05-04
基金项目:国家重点研发计划(2022YFB2602104)
This work was supported by the National Key R&D Program of China(2022YFB2602104).
通信作者:贾经冬(jiajingdong@buaa.edu.cn)

功能可能会影响 App 主要功能的使用。例如，2019 年微信下线了朋友圈表情包评论功能，原因之一就是过载的评论信息使得用户的朋友圈体验变差。同年，微信关闭了漂流瓶功能，后续观察发现此举提升了用户体验和平台安全性。因此，研究 App 中哪些功能应该及时消退同样重要。

De Lima 等[9]虽然通过分析评论中的负面观点，识别出影响 App 未来评级的需求，但并未对这些需求未来是否值得消退提供建议。事实上，在应用功能演化过程中，消退功能占据了较大的比重。Nayebi 等[10]调研了 106 位开发者，其中 78.3%开发者认为删除功能与添加功能同样重要。因此，有必要通过分析用户数据（用户评论和评分）找出需要消退的功能，以帮助 App 设计和开发者决策功能的消退，以提高 App 质量和用户满意度。

针对此问题，本文提出了数据驱动的 App 消退功能研究方法。本文的主要贡献如下：

1）基于用户评论提出了用关键词过滤、语法范式和文本分类从用户评论中获取消退功能的方法；

2）提出了一种利用字数权重阈值法对虚假评分进行检测和更正的方法；

3）基于情感分析、文本分类、主题模型技术，提出了分析消退功能生命周期和识别关键评论的方法；

4）通过实验证明了本文方法的整体有效性，其能为开发者有效识别消退功能提供决策依据，更好地优化 App 的更新。

2 相关工作

用户评论是反映用户需求的重要数据，现有研究大多通过文本分类、文本聚类、自然语言处理、字典和情感分析等手段进行应用评论分析。

Maalej 等[11]将评论划分为错误报告、功能请求、用户体验和评分 4 类。此后，很多研究关注于设计不同的方法来提高评论分类的准确性[12-13]。Memon 等[14]从 Google 和 Apple 应用商店提取了 30 个不同 App 的 2500 条评论，然后用 AR-DOC 分类器从中抽取出“功能请求”类评论进行主题建模，再通过 Naïve Bayes 分类器将主题精准归类为功能和非功能需求两类。Suprayogi 等[15]联合使用文本分析、情感分析和主题建模来提取用户评论信息，并采用支持向量机将评论的内容分为问题、改进、要求和其他 4 类。此外，很多研究侧重于从用户评论中提取有效信息以促进 App 的更新。Tang 等[16]分析目标 App 用户评论，判断其同类应用程序中是否存在匹配的 bug 报告，如果存在则为目标 App 推荐可能存在的 bug。Keertipati 等[17]设计了 3 种方法帮助开发者对从用户评论中抽取的待改进功能进行优先级排序。Chen 等[18]提出了名为 AR-Miner 的评论挖掘框架对评论进行分组和排序。Palomba 等[19]分析用户评论中包含的句子的结构、语义和情感，从维护的角度提取有用的关于软件更新的用户反馈，据此确定需要维护的代码工件。Gao 等[20]通过分析和 App 广告相关的用户评论，为开发者提供实用的广告整合策略。Gao 等[21]构建了一个深度细粒度分类器来识别类似应用程序，然后采用基于情感分析的关键词提取，从类似应用程序的评论中挖掘分析与许可权限相关的评论，以帮助开发人员发现可能的与权限相关的用户需求。

以上对用户评论的研究更多地关注如何从评论中提取 App 的新功能或者待改进的功能，而没有关注 App 的消退功能。为了帮助开发者识别 App 功能变化趋势，Sarro 等[22]提出了应用功能生命周期理论，且发现应用程序的消退功能占比很高。App 中功能太多容易影响可用性，增加资源消耗和维护工作量，因此在 App 演化中有必要关注消退功能。

Murphy-Hill 等[23]从修复 bug 的角度定义功能删除为删除一个特性，而且他们的研究结果表明 75%的开发人员会删除功能来解决 bug，但是并没有把功能删除和用户评论建立关联。Nayebi 等[10]首次对移动应用中的消退功能展开研究，基于开源 Android 应用的代码变化来分析应用的消退功能。他们的研究结果表明，近 1/3 的应用在连续发布的过程中有功能消退现象，而且发现 14.63%的提交说明表明功能消退会受到用户负面反馈的影响。此外，他们调查的 106 位开发者也强调令人恼火的评论和低评级的评论会影响功能的消退。然而，该研究对消退功能的获取是基于代码更新来捕捉的，且并未考虑用户对功能的直接反馈。

App 的用户评论中不仅包含正面反馈，还有对功能不满和使用体验的负面反馈，这些消极意见可为开发者提供消退功能的参考依据[24-25]。然而，基于用户评论的分析研究中缺乏对 App 消退功能的研究，Nayebi 等[26]的最新研究也陈述了这个事实。基于他们之前的工作[10]，Nayebi 等[26]对消退功能进一步展开研究。首先通过对个体调研，发现功能消退会导致负面的用户体验且可能引起用户流失；其次介绍了基于用户评论推荐消退功能的方法，但他们只关注了 App 中的用户界面功能的消退，对用户数据的分析不够全面，例如不包括用户评分，所采用的方法也相对简单且未使用当前相对新的主题模型技术。此外，对推荐结果的验证是基于用户和开发者的调研完成的。

3 总体研究框架设计

本文总体框架如图 1 所示，主要包括 3 个步骤。

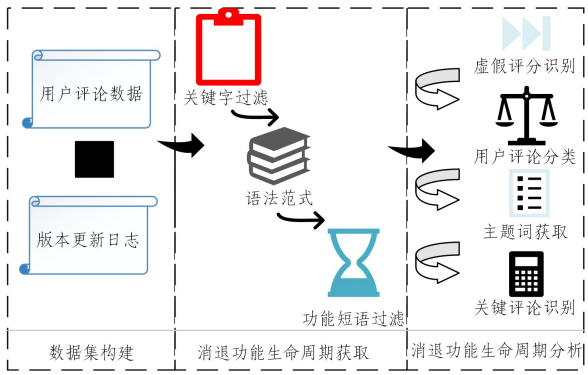

图 1 总体研究框架

Fig. 1 Overall research framework

第一步是数据集构建，主要包括中文的用户评论数据集构建和应用版本更新日志数据集的构建。第二步是 App 消退功能生命周期获取，首先基于关键字过滤、语法范式和文本

分类的方法提取消退功能，然后结合版本更新日志获取消退功能生命周期。第三步是消退功能生命周期分析，通过文本情感分析识别虚假评分，对用户评论进行分类，通过文本聚类和主题模型获得主题词，进而识别关键评论，从用户的视角探究消退功能的原因和生命周期的变化。

4 数据集构建

4.1 数据来源

本文使用的应用市场数据集包括用户评论数据集和版本更新日志数据集。用户评论来自于中文苹果应用商店的用户评论，选择的应用有微信、QQ、淘宝、抖音以及百度。这些应用在各自类别中名列前茅，且包含充足的应用评论数量和足够长的时间跨度。本文收集了上述应用的版本号和时间范围内的用户评论数据，如表 1 所列。

表 1 用户评论数据集概述

Table 1 Overview of user reviews datasets

应用名称	版本号范围	时间范围
微信	7.0.0－8.0.11	2018.12.21－2021.08.23
百度	12.0.0.11－12.20.6	2020.09.10－2021.07.30
抖音	14.0.0－16.9.0	2020.12.17－2021.07.27
淘宝	9.0.0－9.26.0	2019.09.29－2021.05.23
QQ	8.0.0－8.7.8	2019.04.13－2021.06.01

所收集的用户评论数据包含评论时间、用户名、评论标题、评论内容和评分。版本更新日志数据集与所选应用的版本号范围相对应，其来源包括应用商店的更新日志和官方网站的发布说明等官方渠道。

4.2 数据预处理

上述获取的用户评论中包含大量噪声数据，如文本错别字、重复单词、非中文单词，以及不符合语法规则的句子等[27]。此外，用户评论的质量参差不齐，有些评论仅包含无意义内容，例如评论文本“1”。因此，需要对评论数据进行预处理，从版本更新日志和用户评论中清除噪声句子和字符。

用户评论通常简短且含有非正式或混合词汇。对于每条用户评论的预处理过程如图 2 所示，分为 3 步。

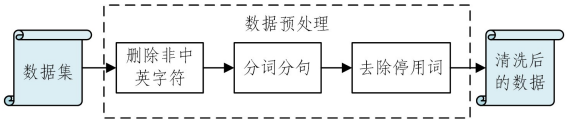

图 2 数据预处理流程

Fig. 2 Process of data preprocessing

首先，构建正则匹配的规则，根据正则表达式删除非中文和非英文字符，以减少对未来分析的干扰。

其次，由于汉语中的单词之间没有明确的边界，因此本文使用哈工大研发的 LTP 工具（语言技术平台）对评论进行分句和分词。LTP 还提供了词性标注和句法分析等功能。

最后，利用经过筛选和过滤的停用词列表删除评论中的无意义词，如语气助词等。如果句子在去除停用词后不含任何词语，则该句子将被删除。

为了更清晰地说明数据预处理效果，给出了清洗实例，如表 2 所列。

表 2 用户评论清洗实例

Table 2 Examples of user reviews cleaning

原评论	清洗后评论	处理类型
希望视频的时候微信[看]自带美颜的功能。	希望视频的时候微信自带美颜的功能。	删除非中英文字符
为什么现在没有提取文字功能？就是从图片里面提取出来。一定要处理啊啊	为什么现在没有提取文字功能？从图片里面提取出来。一定要处理	去停用词

5 消退功能生命周期获取

5.1 消退功能生命周期定义

定义一个功能特性 f，若从 t_0 时刻起，f 属于应用 C 的应用数据集 D，则将其表示为 $\{C|f\in C_{D\{t_0\}}\}$。若 t_1 时刻（其晚于 t_0 时刻）有 $\{C|f\notin C_{D\{t_1\}}\}$，并且在 t_0 到 t_1 之间的任何时刻 t，都有 $\{C|f\in C_{D\{t\}}\}$，则说明从 t_1 时刻起，功能特性 f 已经不属于应用 C 的应用数据集 D，则将区间 (t_0,t_1) 定义为应用 C 的消退功能 f 的生命周期。

5.2 消退功能生命周期获取

从用户评论中获取消退功能的流程如图 3 所示。首先使用关键字过滤的方法定位可能出现消退功能的评论，然后采用基于语法范式的方法从用户评论中提取候选功能短语，最后构建分类器，确定候选功能短语中与应用特征描述相关的短语。

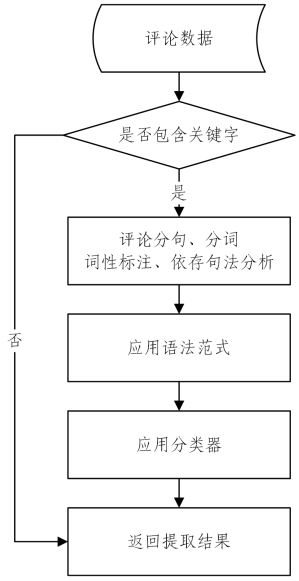

图 3 消退功能短语获取流程

Fig. 3 Process of obtaining fading function phrases

首先，通过人工分析用户评论获取一些常见的关键词，如“消失”“不见”等，然后综合文献中提到的关键词进行中英文替换，最后应用中文同义词和近义词的检索方法扩充关键词库。最终构建过滤评论的关键词集合，如表 3 所列。

表 3 关键词集合

Table 3 Keywords set

关键词
消失，消退，消亡，消除，删除，没有，不见，去哪，移除，去除，找不到，取消，不能，还我，还给我，下架，改回，去掉，删掉，还原

随后，构建语法范式来挖掘过滤后的评论，以找到其包含

的消退功能短语。在构建语法范式时,本文综合分析了现有的语言模板,构建语法分析树来评估其是否可以用于处理中文文本,随后挑选并优化语法范式,确保有效提取功能语法短语。最终,本文定义了提取功能短语的 12 种语法范式模板(见表 4),主要包括词性特征以及依存句法特征。

表 4 功能获取语法范式

Table 4 Function acquisition syntax paradigm

序号	语言模板	示例	功能短语
1	na	求解呀!我的微信到现在也没有视频号的。	视频号
2	v	为什么没有支付了?	支付
3	v+na	设置中没有聊天选项了。全键盘下没法换行了。一条道输入到底。	聊天选项
4	na+v	视频美颜消失了。	视频美颜
5	na+na	还我视频动态!	视频动态
6	v+v	最近分享朋友圈的音乐不能直接播放了,一定要二次点击链接才能播放,而且没有悬浮播放。	悬浮播放
7	a/b+na	我的微信版本是最新的但却没有深色模式这是怎么回事?	深色模式
8	d+v+na	最近总是出现信息铃声,打开后却没有未读消息。	未读消息
9	na+v+v	为何 ios 版没有消息免打扰功能?	消息免打扰
10	v+v+na	卸载重新安装能不能保留聊天数据	保留聊天数据
11	SBV	提示完全都没有	提示
12	ATT+SBV	语音提示完全没有	语音提示

表 4 中,na 是本文定义的名词集合,包括名词(n)、方位名词(nd)等;v 代表动词;a 代表形容词;b 代表其他名词修饰语;d 代表副词;SBV 是依存句法关系中的主谓关系;ATT 是依存句法关系中的定中关系。

以评论“8.01 版本小程序浮窗消失”为例,阐述第二步过程。首先通过分词将评论句子拆分为短语,然后为分词后的短语标注上具体的词性,再对整体进行依存句法分析,具体过程如图 4 所示。经历上述过程后,再运用表 3 中的关键字模板和表 4 中的语法范式 4 和 12 分别提取到“程序浮窗”和“小程序浮窗”的候选短语。

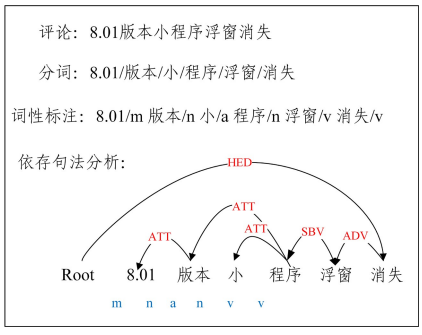

图 4 语法范式分析示意图

Fig. 4 Schematic diagram of syntax paradigm analysis

接下来,构建文本分类器,过滤出实际的功能短语,因为经过关键字过滤、语法范式挖掘获取的候选功能短语仍然包含很多非功能性短语和无意义的词语(见表 5)。

从候选功能短语中过滤出消退功能属于短文本分类任务,本文主要采用基于机器学习和基于预训练的深度学习方法。使用词袋(BoW)、TF-IDF 和 Word2Vec 算法的特征工程方法,结合在评论分类中常用的支持向量机(SVM)、逻辑回归(LR)以及决策树(DT)机器学习算法进行分类。在深度学习的预训练模型中,本文应用 BERT 算法,因为 BERT 同时考虑文本中词汇的前后文信息[28],有利于提高消退功能识别的准确性和效率。

表 5 功能短语中噪声数据示例

Table 5 Example of noise data in functional phrases

评论数据	语法范式提取结果	非功能短语
为什么现在没有了提取文字功能。就是从图片里面提取出来的。	[‘现在’,‘提取’,‘提取文字’]	[‘现在’,‘提取’]

获取消退功能后,遍历用户评论和版本更新日志以确定其生命周期。在此过程中,需要结合应用更新日志与评论数据进行回溯,找到该功能首次出现的时间和消退版本号对应的时间。这样便可明确某消退功能从首次出现到消退的完整生命周期。

6 消退功能生命周期变化分析

确定消退功能后,为基于用户评论和评分来分析其生命周期变化,本文构建消退功能生命周期分析框架,利用情感分析对虚假评分进行检测更正,采取文本分类将评论数据分类,通过文本聚类和主题模型为评论数据生成主题关键字,并识别关键评论。

6.1 虚假评分识别与更正

在应用市场中,用户除了进行文字评论外,还可以对应用进行打分(通常为 1~5 分)。用户评分是用户喜好的直观反馈,但存在虚假的评论和评分[29-30]。笔者认为,用户评分与用户评论所表达的情感色彩不符时,这种评分为虚假评分。

本文将用户评论的情感分析得分与用户评分进行对比,从而识别虚假评论。首先对用户评分线性归一化,然后使用百度情感倾向分析工具计算评论的情感得分,其中 0 代表负面,1 代表正面。定义第 i 条评论的归一化后的评分为 $rate_i$,情感分析得分为 sen_i,由此定义 α_i 为评分与情感分析得分之差,如式(1)所示:

$$\alpha_i = |rate_i - sen_i| \tag{1}$$

本文对 α_i 在不同阈值区间进行实验,探究仅使用阈值时的效果,同时考虑到评论字数对情感分数起补充作用,评论字数越多表达的语气往往越强烈。因此,获取评论数据集中评论字数的最大值 $len_{\max}$ 与最小值 $len_{\min}$,构成数据集的评论字数区间$[len_{\max}, len_{\min}]$,采用线性映射的方法将评论字数映射到区间$[a,b]$生成强度系数 β_i,再将其与 α_i 相乘得到强度差值 α_i',如式(2)所示:

$$\begin{aligned} \beta_i &= a + \frac{b-a}{len_{\max} - len_{\min}}(len_i - len_{\min}) \\ \alpha_i' &= \beta_i \times \alpha_i \end{aligned} \tag{2}$$

本文将这种方法称为字数权重阈值法,通过调整映射区间来探究其是否可以提高虚假评分检测的效率。

如果检测到虚假评分,就按照文本情感分析分数对虚假评分进行更正,更正后的评分会是准确的,符合用户评论感情色彩的。

6.2 评论分类

对用户评论分类可以了解哪类评论中容易出现消退功能，为功能迭代提供参考。而现有的评论分类是多样的[31]，从用户角度综合考虑相关研究后，将用户评论分为 5 类，如表 6 所列。

表 6 用户评论类别

Table 6 User reviews category

序号	类别名称	类别说明
1	Bug	用户反馈的应用出现的问题
2	用户体验	用户对于应用的使用评价以及感受等
3	信息询问	用户对应用相关功能以及相关信息的提问与咨询
4	资源利用	与应用资源以及消耗相关，比如内存、温度等
5	功能请求	用户对应用功能的建议以及已有功能的改善要求等

然后，对包含消退功能生命周期的评论进行分类。本文采用 3 种基于预训练模型的文本分类模型 BERT，ERNIE 和 RoBERTa 对评论数据进行分类，它们在中文数据集中有较好的表现。BERT 是一种基于 Transformer 的双向预训练语言模型，通过同时考虑上下文信息来生成丰富的文本表示。Xiao等[2]用 BERT 对评论进行多分类，其效果优于机器学习的文本分类方法。ERNIE 是一种知识增强的预训练语言模型，通过整合外部知识库的信息来提升语言表示能力。RoBERTa 是 BERT 的优化版本，通过更大规模的数据和更长的训练时间进行预训练，以提高模型性能。

6.3 用户评论聚类与主题提取

仅对评论进行分类是不够的，因为同一类别下仍然有很多评论，所以需要通过聚类和主题模型对评论进行更深细粒度的挖掘与分析。基于 BERTopic[32] 和 Anchored CorEx[33] 方法，本文构建 BERTopic-Corex 主题模型用于生成消退功能对应的主题词，模型结构如图 5 所示。

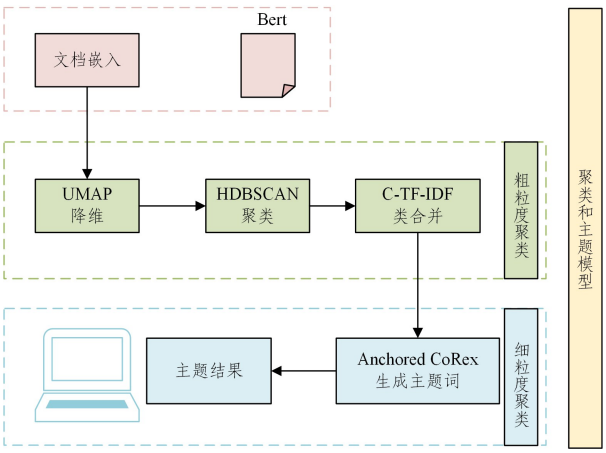

图 5 BERTopic-Corex 结构图

Fig. 5 Structure diagram of BERTopic-Corex

第一步 利用 Sentence Transformer 从一组文档中创建文档嵌入。

第二步 用 UMAP 算法进行向量降维，因为 UMAP 很好地保留了文本嵌入的局部和全局结构，而且 UMAP 对嵌入维度没有计算限制，所以可以供跨不同维度空间的语言模型使用。

第三步 使用 HDBSCAN，算法聚类。HDBSCAN 是一种层次密度聚类方法，相比 DBSCAN 提出了新的距离计算公式，能更有效地反映数据点之间的分布关系。

第四步 因为聚类后仍然可能存在类很多且有些类规模很小的情况，所以使用 BERTopic 框架提出的 C-TF-IDF 方法进行类合并。该方法把聚类产生的每个类当作一个文档，在文档上用 TF-IDF 获取特征，如式(3)所示：

$$W_{x,c}=tf_{x,c}\times\log\left(1+\frac{A}{f_x}\right) \tag{3}$$

其中，$tf_{x,c}$是词 x 在类 c 中的频率，f_x是词 x 在所有类中出现的频率，A 是每个类的平均词数。

第五步 提取每个类的主题词。虽然经过第四步可以产生关键词，但会输出很多出现频次高但与功能特性无关的主题。为提高主题模型最后的输出的质量，可以加入先验知识将任务转变为半监督的方式。Gallagher 等[33]提出一种可以预先设定先验知识的主题模型 Anchored CorEx。该算法以信息熵为基础，通过最大化相关性解释来确定文档的关键字。定义 $\boldsymbol{X}$ 是离散的随机变量，$H(\boldsymbol{X})$是其信息熵的表示，$\boldsymbol{X}_G$是 $\boldsymbol{X}$ 的一个子集，$TC(\boldsymbol{X}_G)$是相关性解释，如式(4)和式(5)所示：

$$H(\boldsymbol{X})=E_x[-\log p(x)] \tag{4}$$

$$TC(\boldsymbol{X}_G)=\sum_{i\in G}H(\boldsymbol{X}_i)-H(\boldsymbol{X}_G) \tag{5}$$

接下来定义条件信息熵，如式(6)所示：

$$TC(\boldsymbol{X}|\boldsymbol{Y})=\sum_i H(\boldsymbol{X}_i|\boldsymbol{Y})-\boldsymbol{H}(\boldsymbol{X}|\boldsymbol{Y}) \tag{6}$$

利用 TC 和条件信息熵之差可以表示某一变量对于数据的相关性的贡献，如式(7)所示：

$$\begin{aligned}TC(\boldsymbol{X}_G;\boldsymbol{Y})&=TC(\boldsymbol{X}_G)-TC(\boldsymbol{X}_G|\boldsymbol{Y})\\&=\sum_{i\in G}I(\boldsymbol{X}_i;\boldsymbol{Y})-I(\boldsymbol{X}_G;\boldsymbol{Y})\end{aligned} \tag{7}$$

在主题模型中，$\boldsymbol{X}_G$表示一系列文档，而 $\boldsymbol{Y}$ 表示主题词，通过最大化相关性解释，找到对文档解释程度最高的主题词，通过迭代更新连通矩阵 $\boldsymbol{\alpha}_{i,j}^l$来优化参数，如式(8)所示：

$$\boldsymbol{\alpha}_{i,j}^l=\exp(\lambda^l(I(\boldsymbol{X}_i;\boldsymbol{Y}_j)-\max_j(I(\boldsymbol{X}_i;\boldsymbol{Y}_j)))) \tag{8}$$

使用式(7)中的替代形式重写目标引入指标变量 $\alpha_{i,j}^l$，其中 t 表示迭代次数，λ 控制 softmax 函数的尖锐度。每次迭代计算式(9)和式(10)来计算边界，通过不断更新的 $\alpha_{i,j}^l$来计算式(11)，直到收敛为止。

$$p_t(y_j)=\sum_{\bar{x}}p_t(y_j|\bar{x})p(\bar{x}) \tag{9}$$

$$p_t(x_i|y_j)=\frac{\sum_{\bar{x}}p_t(y_j|\bar{x})p(\bar{x})\mathbb{I}[\bar{x}_i=x_i]}{p_t(y_j)} \tag{10}$$

$$\log p_{t+1}(y_j|x^l)=\log p_t(y_j)+\sum_{i=1}^n\alpha_{i,j}^l\log\frac{p_t(x_i^l|y_j)}{p(x_i^l)}-\log\mathbb{Z}_j(x^l) \tag{11}$$

正常的 $\alpha_{i,j}^l$在[0,1]之间，而将第 i 号词固定在第 j 号主题可以令 $\alpha_{i,j}^l=\beta_{i,j}$，其中 $\beta_{i,j}$代表强度。将连通矩阵的某些值固定即可将算法输出的主题词固定，这样固定的主题就形成了先验知识。

通过以上方法，可以得到与消退功能相关的主题词。优质的主题有助于开发者更加直观地看到用户反馈的与消退功能相关的焦点话题。

7 实验结果与分析

7.1 评价指标

本文使用机器学习和深度学习的算法，故采用的评估指标为精确率(Precision)、召回率(Recall)、F-measure 以及准确率(Accuracy)。这些指标计算是基于分类预测的混淆矩阵，如表 7 所列。

表 7 混淆矩阵

Table 7 Confusion matrix

样本	预测为正	预测为负
正样本	TP	FN
负样本	FP	TN

在本文的分类任务中，当评估某一样本类别时，该类别的样本被视为正样本，其余类别的样本被视为负样本。4 个评价指标的计算如式(12)－式(15)所示。这些指标的值越大，表明模型的性能越好。

$$Precision=\frac{TP}{TP+FP} \tag{12}$$

$$Recall=\frac{TP}{TP+FN} \tag{13}$$

$$F\text{-}measure=\frac{2\times precision\times recall}{precision+recall} \tag{14}$$

$$Accuracy=\frac{TP+TN}{TP+FP+TN+FN} \tag{15}$$

7.2 消退功能获取实验结果

通过关键字和语法范式过滤后，得到一个包含 2400 条候选功能短语的数据集，按照该候选功能短语是否为应用功能进行人工标注，分类结果如表 8 所列，加粗数据为最优结果。从表中与看出，BERT 算法相比其他分类算法，其各项指标均达到了最佳表现，可以很好地从候选功能短语中识别出功能短语。这是因为 BERT 基于 Transformer 的双向预训练机制，能够深入理解上下文语义，更准确地识别功能短语。

表 8 功能过滤分类器结果

Table 8 Results of function filtering classifiers

分类方法	*Precision*	*Recall*	*F-measure*	*Accuracy*
SVM-BoW	0.816	0.821	0.814	0.821
LR-BoW	0.818	0.824	0.816	0.824
DT-BoW	0.825	0.828	0.817	0.828
SVM-TFIDF	0.814	0.824	0.813	0.821
LR-TFIDF	0.840	0.844	0.839	0.844
DT-TFIDF	0.820	0.825	0.818	0.825
SVM-Word2vec	0.856	0.854	0.852	0.854
LR-Word2vec	0.847	0.837	0.831	0.837
DT-Word2vec	0.844	0.842	0.843	0.842
BERT	**0.884**	**0.885**	**0.884**	**0.885**

为验证本文中提取消退功能方法的有效性，随机抽取 1000条评论数据，采用人工识别得出 216 个消退功能，使用本文方法提取出其中 170 个消退功能，准确率达到 0.787，说明该方法可以有效地从用户评论中获取消退功能。

7.3 消退功能生命周期分析实验结果

7.3.1 虚假评分检测实验结果

将文本情感分析用于对虚假评分的检测，对比阈值法和字数权重阈值法，检测结果如表 9 所列。可以看出，在 16 种比较区间上，字数权重阈值法在 11 种比较区间上优于阈值法(见表 9 中下划线数据)，其最高准确率可以达到 0.841，相比阈值法有很大的提升，这说明以阈值法为基础改进后的字数权重阈值法更加有效。

表 9 虚假评分检测结果

Table 9 Results of fake rating detection

阈值	映射区间 a	映射区间 b	字数权重阈值法	阈值法
0.5	0.5	1.5	**0.841**	0.769
0.5	0.6	1.4	0.827	0.769
0.5	0.7	1.3	0.788	0.769
0.5	0.8	1.2	0.778	0.769
0.6	0.5	1.5	0.795	0.779
0.6	0.6	1.4	**0.841**	0.779
0.6	0.7	1.3	0.835	0.779
0.6	0.8	1.2	0.810	0.779
0.7	0.5	1.5	0.765	0.797
0.7	0.6	1.4	0.786	0.797
0.7	0.7	1.3	0.833	0.797
0.7	0.8	1.2	0.835	0.797
0.8	0.5	1.5	0.752	0.827
0.8	0.6	1.4	0.755	0.827
0.8	0.7	1.3	0.767	0.827
0.8	0.8	1.2	0.832	0.827

基于表 9 的实验结果，近一步将虚假评分检测当作二分类问题，比较字数权重阈值法与 BERT 分类法的准确率，结果如表 10 所列。

表 10 虚假评分检测对比结果

Table 10 Comparative results of fake rating detection

方法	准确率
分类法(BERT)	0.74
字数权重阈值法	**0.84**

从表 10 可以看出，字数权重阈值法的准确性优于 BERT 方法。用分类法进行虚假评分检测，实际上仅仅是通过文本语义信息去判断，而字数权重阈值法结合了情感评分与字数两个特征，评论字数一般与评论的情感密切相关，故其实验结果更优。

7.3.2 评论分类实验结果

从评论数据库中随机抽取 4 000 条数据进行人工标注，随后采用 3 种基于预训练的文本分类模型 BERT，ERNIE 和 RoBERTa 构建用户评论分类器进行实验，在测试集上的分类实验结果如表 11 所列。显然，在 4 项评价指标上，BERT 算法的表现都明显优于 ERNIE 算法；在其中 3 项指标上，BERT 算法的表现优于 RoBERTa 算法，因此本文最终选用 BERT 算法进行评论分类。尽管 ERNIE 和 RoBERTa 都是在 BERT 基础上进行优化的，但是一方面消退功能相关的评论数量是有限的，另一方面用户评论数据集大多以短文本为主，有的表述比较口语化，文本前后文信息相对有限。因此，BERT 这种基础算法的优势更加凸显。

表 11 不同分类器的评论分类结果

Table 11 Reviews classification results of different classifiers

分类方法	*Precision*	*Recall*	*F-measure*	*Accuracy*
BERT	0.818	**0.816**	**0.809**	**0.816**
ERNIE	0.776	0.762	0.752	0.762
RoBERTa	**0.823**	0.801	0.805	0.801

评论分类的结果如表 12 所列，可以看出，用 BERT 算法构建的分类器对评论分类的准确率能够达到 0.816，可以较好地完成对用户评论意图的分类任务。

表 12 评论分类结果

Table 12 Results of reviews classification

类别	*Precision*	*Recall*	*F1-score*
Bug	0.725	0.9434	0.820
用户体验	0.835	0.8600	0.847
信息询问	0.811	0.6670	0.732
资源利用	0.500	0.6250	0.556
功能请求	0.933	0.8235	0.875
Accuracy			**0.816**

基于构造的分类器，对已识别的消退功能的用户评论进行分类，以确定哪一种评论分类更容易识别出消退功能。同时，为了识别消退功能评论分类的不同特征，将消退功能评论的分类结果与所有评论的分类结果进行对比，所有评论分类如图 6 所示。消退功能评论分类结果如图 7 所示。

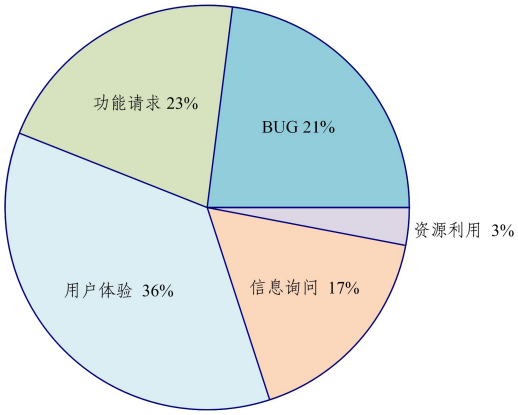

图 6 所有用户评论分类占比

Fig. 6 Proportion of all user reviews by category

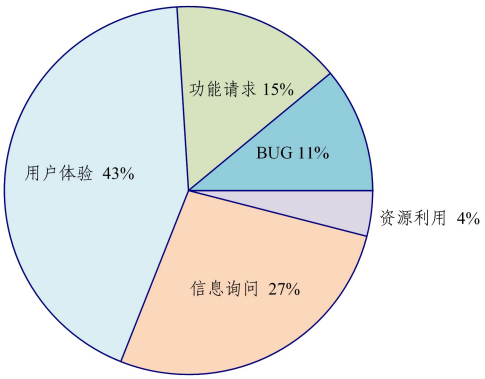

图 7 含消退功能的用户评论分类占比

Fig. 7 Proportion of user reviews with fading function

对比图 6 和图 7 可以看出，在所有评论数据集中，用户体验评论占比最大，超过了 1/3，其次是功能请求和 bug，信息咨询占比不到 20%。对于消退功能评论，用户体验类仍居首位，且占比增加了 7%。Bug 和功能请求类占比分别下降 10%和 8%，且二者不再居于优势地位。

相对所有评论的分类，信息询问类占比显著提升，提高了 10%，处于消退功能评论的第二位。信息询问往往意味着功能对用户使用不友好，使用户产生很多疑惑。用户体验和信息询问二者共占消退功能评论的 70%，这说明消退功能生命周期内在用户体验和信息询问上带给了用户更多的问题。因此，通过用户评论分析消退功能时要重点关注用户体验和信息询问类评论，它们极可能蕴含着潜在的消退功能。

此外，虽然资源利用类在两种分类中占比都很低，但相对于全体评论，消退功能评论中资源利用类占比绝对增长 1%，相对增长 33%。这说明在消退功能中，资源利用的问题可能比一般的功能问题更加突出，因此从评论中识别消退功能时不能忽视资源利用类。

基于上述分析，开发者从用户评论识别消退功能时，应该重点关注用户体验、信息询问和资源利用这 3 类评论。

7.3.3 聚类和主题模型实验结果

针对主题模型的结果评估，对于特定的无标注数据集，人工评价是一种比较可靠的评价主题是否有意义的主观方法。本文通过统计产生的主题词与消退功能是否相关来评价主题词产生的质量，计算方法如式(16)所示：

$$Hit=\frac{\text{与功能关联的主题词}}{\text{主题词总数}} \tag{16}$$

实验在 10 个消退功能数据集上进行，分别对比 LDA，BERTopic-ctfidf 和 BERTopic-CorEx 这 3 种主题模型在不同主题词数量下的结果，如表 13 所列。

表 13 3 种主题模型的实验结果

Table 13 Experimental results of three thematic models

	LDA	BERTopic-ctfidf	BERTopic-CorEx
top10	0.21	0.33	**0.50**
top15	0.22	0.31	**0.43**
top20	0.23	0.30	**0.39**

从表 13 可以看出，本文改进后的 BERTopic-CorEx 模型所产生的主题词在不同主题词数量下均优于 LDA 和 BERTopic-ctfidf，且在 top10 和 top15 上有较大的提升，因此 BERTopic-CorEx 模型可以产生更多有意义的主题词来帮助识别功能的特性。这也说明本文提出的加入一些先验知识的主题提取是更为合理的。

以微信“时刻视频”功能为例，采用 BERTopic-CorEx 模型和 BERTopic-ctfidf 产生的 8 个主题词结果如表 14 所列。

表 14 主题模型结果示例

Table 14 Example of theme model results

模型	主题词	Hit
CTFIDF	iPad/没有/删除/不能/朋友圈/更新/建议/上传	0.375
CorEx	时间/现在/颜色/反感/比例/横屏/关闭/隐私	**0.625**

可以看到，对于“时刻视频”功能，运用 BERTopic-CorEx 产生的与功能关联的主题词多于 BERTopic-ctfidf 方法。通过主题词可以看出，与“时刻视频”功能相关的用户评论中的代表性词汇有“颜色”“时间”“比例”“隐私”等，反映出用户对该功能的这些方面格外重视且有反馈，因此开发者也应该更加关注该功能的这些特性。

7.3.4 关键评论识别

对用户评论数据进行分类或提取主题词的过程中，难免会丢失一些文本信息，而评论文本是最直接的数据源。然而由于用户评论数量过多，人工阅读耗时、耗力，因此有必要识

别某一消退功能下最有代表性的关键评论。本文综合每条评论字数、评分、文本分类结果、聚类结果以及是否包含主题词等因素,按照不同的权重为每条评论打分,根据该分数识别关键评论,具体如式(17)所示:

$$Score_i = \sum Score_c \times k_c + Score_t \times k_t + Score_{clu} \times k_{clu} + Score_{rating} \times k_{rating} + Len \times k_{len} \quad (17)$$

其中,$Score_i$ 表示对第 i 条评论的打分结果;$Score_c$,$Score_t$,$Score_{clu}$,$Score_{rating}$,Len 分别表示该评论类别、主题、聚类、更正后的评分以及长度得分;k_c,k_t,k_{clu},k_{rating},k_{len} 分别代表它们的权重。

按照式(17),以数据集中微信"深色模式"功能为例,产生的前三条关键评论如表 15 所列。

表 15 关键评论示例

Table 15 Examples of key user reviews

关键评论
1. 深色模式做的不好真的不如不做,背景不是纯黑,字体白色也不够亮,像是深色模式和夜间模式的混搭,白天看着眼睛特别累,关键还不能关了,只能在系统里关了深色模式才能解,开发者是打算让我戒了微信吗? iOS 自带软件的深色模式学学啊,不行你自家 QQ 的深色模式做得也很好啊。
2. 深色模式不好。能把深色模式去掉吗? 或者可以让用户选择开启或关闭! 有人会说去亮度设置里可以关闭苹果的夜间模式! 但是那样色彩很失真眼睛更受不了!。
3. 不小心更新了这深色模式太难受了!!!! 而且有没有考虑过眼睛闪光的人的感受! 这种黑底反白,打字都是有重影看不清的!!! 虽然可以跟随系统切换了,但是我不想系统是白色的啊! 你们微信的用户体验设计是怎么想的啊!!!

从表 15 可以看出,这些关键评论都是对"深色模式"功能的用户体验反馈,这与评论分类实验结果相吻合,即用户体验是影响消退功能的首要因素。其次,关键评论也清晰地给出了功能在哪些特性存在缺陷,如"色彩失真""没有开关"等,这些既可以作为开发者对功能改善的指导方向,也蕴含着如果实现不了则需要消退的依据。关键评论分析有助于帮助开发者从大量评论中快速聚焦,找出问题的核心。

结束语 本文对应用更新中消退功能的生命周期进行提取和分析,针对消退功能获取提出了综合关键字过滤、语法范式以及文本分类的研究方法;在消退功能生命周期分析中,提出字数权重阈值法检测和修正用户虚假评分,采用文本分类、聚类和主题模型为消退功能产生代表特征并识别关键评论,通过实验验证了本文方法的可行性。通过本文方法,App 开发者可以基于一定量的用户评论和评分分析来确定合适的需要消退的功能,有助于 App 的质量提升和维护。

然而,本文仍然存在一定的不足。首先,从用户评论中获取消退功能时,关键词集合和语法范式的构建未能完全覆盖所有相关评论,未来的研究将扩充关键词集合,并引入语义角色标注,以提高评论数据的挖掘效率。其次,除了用户评论数据之外,其他用户数据,例如用户对功能的使用频率,也可能影响 App 功能的消退,这也是未来研究消退功能的方向。

参 考 文 献

[1] DABROWSKI J, LETIER E, PERINI A, et al. Analysing App reviews for software engineering: a systematic literature review [J]. Empirical Software Engineering, 2022, 27(2): 43.

[2] XIAO J M, CHEN S Z, FENG Z Y, et al. Anautomatic analysis of user revews method for App evolution and maintenance [J]. Chinese Journal of Computers, 2020, 43(11): 2184-2202.

[3] WANG Y, ZHENG L W, ZHANG Y Y, et al. Softwarerequirements mining method for chinese App user review data [J]. Computer Science, 2020, 47(12): 56-64.

[4] 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.

[5] YAO Y M, JIANG W Y, WANG Y L, et al. Non-functional requirements analysis based on application reviews in the Android App market [J]. Information Resources Management Journal, 2022, 35(2): 1-17.

[6] JHA N, MAHMOUD A. Mining non-functional requirements from App store reviews [J]. Empirical Software Engineering, 2019, 24(5): 1-37.

[7] FU S H, XUE K K, YANG M Y, et al. An exploratory study on users' resestance to mobile App updates: Using netnography and fsQCA [J]. Technological Forecasting and Social Change, 2023, 191(6): 122479.

[8] HUNGER T, ARNOLD M, PESTINGER R. Risks and requirements in sustainable App development-a review [J]. Sustainability, 2023, 15(8): 7018.

[9] DE LIMA V M A, DE ARAUJO A F, MARCACINI R M. Temporal dynamics of requirements engineering from mobile App reviews [J]. PeerJ Computer Science, 2022, 8(2): e874.

[10] NAYEBI M, KUZNETSOV K, CHEN P. Anatomy of functionality deletion: an exploratory study on mobile Apps[C]// International Conference on Mining Software Repositories (MSR). 2018: 243-253.

[11] MAALEJ W, KERTANOVIC Z, NABIL H, et al. On the automatic classification of App reviews [J]. Requirements Engineering, 2016, 21: 311-331.

[12] BISWAS M, ANISH P R, GHAISAS S. Interpretable Appreview classification with tansformers[C]// International Requirements Engineering Conference Workshops(RE). 2024: 26-34.

[13] AL KILANI N, TAILAKH R, HANANI A. Automatic classification of Apps reviews for requirement engineering: exploring the customers need from healthcare Applications[C]// International Conference on Social Networks Analysis, Management and Security(SNAMS). 2019: 541-548.

[14] MEMON Z A, MUNAWAR N, KAMAL M. App store mining for feature extraction: analyzing user reviews [J]. Acta Scientiarum Technology, 2023, 46(1): e62867.

[15] SUPRAYOGI E, BUDI I, MAHENDRA R. Information Extraction for Mobile Application User Review [C] // International Conference on Advanced Computer Science and Information Systems(ICACSIS). 2018: 343-348.

[16] TANG X Z, TIAN H Y, KONG P F, et al. App review driven collaborative bug finding [J]. Empirical Software Engineering, 2024, 29(5): 124.

[17] KEERTIPATI S, SAVARIMUTHU B T R, LICORISH S A. Approaches for prioritizing feature improvements extracted from App reviews[C]// International Conference on Evaluation and Assessment in Software Engineering(EASE). 2016: 1-6.

[18] CHEN N, LIN J, HOI S C H, et al. AR-miner: mining informative reviews for developers from mobile App marketplace[C]// International Conference on Software Engineering(ICSE). 2014: 767-778.

[19] PALOMBA F, SALZA P, CIURUMELEA A, et al. Recommending and localizing change requests for mobile Apps based on user reviews[C]// International Conference on Software Engineering(ICSE). , 2017: 106-117.

[20] GAO C Y, ZENG J C, LO D, et al. Understanding in-App advertising issues based on large scale App review analysis [J]. Information and Software Technology, 2022, 142(1): 106741.

[21] GAO H C, GUO C K, BAI G D, et al. Sharing runtime permission issues for developers based on similar-App review mining [J]. Journal of Systems and Software, 2022, 184(1): 111118.

[22] SARRO F, AI-SUBAIHIN A A, HARMAN M, et al. Feature lifecycles as they spread, migrate, remain, and die in App stores [C] // International Requirements Engineering Conference (RE). 2015: 76-85.

[23] MURPHY-HILL E, ZIMMERMANN T, BIRD C, et al. The design of bug fixes[C]// International Conference on Software Engineering. IEEE, 2013: 332-341.

[24] GUZMAN E, OLIVEIRA L, STEINER Y, et al. User feedback in the App store: a cross-cultural study[C]// International Conference on Software Engineering. 2018: 13-22.

[25] MALGAONKAR S, LICORISH S A, SAVARIMUTHU B T R. Prioritizinguser concerns in App reviews: a study of requests for new features enhancements and bug fixes [J]. Information and Software Technology, 2022, 142(1): 106798.

[26] NAYEBI M, KUZNETSOV K, ZELLER A, et al. Recommending and release planning of user-driven functionality deletion for mobile apps [J]. Requirements Engineering, 2024, 29: 459-480.

[27] GU X, KIM S. What parts of your Apps are loved by users? [C]// International Conference on Automated Software Engineering(ASE). 2015: 760-770.

[28] WU H Y, DENG W J, NIU X T, et al. Identifying key features from App user reviews [C]// International Conference on Software Engineering(ICSE). 2021: 922-932.

[29] MARTENS D, MAALEJ W. Towards understanding and detecting fake reviews in App stores [J]. Empirical Software Engineering, 2019, 24(6): 3316-3355.

[30] HE D J, PAN M H, HONG K, et al. Fakereview detection based on pu learning and behavior density [J]. IEEE Network, 2020, 34(4): 298-303.

[31] WANG X H, ZHANG T, TAN Y H, et al. How to effectively mine App reviews concerning software ecosystem? A survey of review characteristics [J]. Journal of Systems and Software, 2024, 213(1): 112040.

[32] GROOTENDORST M. BERTopic: neural topic modeling with a class-based TF-IDF procedure [J]. arXiv: 2203. 05794, 2022.

[33] GALLAGHER R J, REING K, KALE D, et al. Anchored correlation explanation: Topic modeling with minimal domain knowledge [J]. Transactions of the Association for Computational Linguistics, 2017, 5(5): 529-542.

JIA Jingdong, born in 1975. Ph.D, associate professor, master supervisor, is a member of CCF (No. 77150M). Her main research interests include artificial intelligence and software engineering.

(责任编辑:何杨)