文章编号: 1671-6833(2023)03-0056-06

一种面向用户反馈的智能分析与服务设计方法

汪 烨[1], 周思源[1], 翁知远[2], 陈骏武[1]

(1.浙江工商大学 计算机与信息工程学院,浙江 杭州 310018; 2.内布拉斯加大学林肯分校 计算机科学与工程系,美国内布拉斯加州 林肯市 68508)

摘 要: 针对用户评论数据,提出了一种面向用户反馈的智能分析与服务设计方法。该方法选取了 IOS 平台多个 App 的用户评论数据,对其进行智能挖掘和分类,分析其中的潜在需求。首先,分析用户需求类别,将划分的 10 个需求进行具体定义。其次,对用户数据进行爬取、清洗和标注,形成软件分类数据集。通过实验检验 TextCNN、BiL-STM_Attention 和 BERT 对用户评论数据智能分类的效果,将分类结果进行优先级排序。最后,将该方法封装成一种可重用的智能服务供使用者远程调用。实验结果表明: TextCNN 模型综合效果最好,在单一指标 *Precision* 上,BERT 模型效果最好; BERT 模型利用并行计算优化训练过程,使其可拓展到大规模项目,在数据量大、精确性要求比较高的情况下,推荐 BERT 模型; 反之,在应对数据小、时限紧的情况时,推荐 TextCNN 模型。

关键词: 需求分析; 服务设计; 服务计算; 深度学习; 用户反馈

中图分类号: TP311 **文献标志码:** A **doi:** 10.13705/j.issn.1671-6833.2022.06.004

近年来,智能服务受到了广大研究者的重视,实现了飞速发展。互联网应用市场是近年来增长最快的领域之一,例如苹果应用商店在两年半内就有超过 30 万的应用上线[1],而早在 2011 年,Henze 等[2]分析了以苹果应用为主的多个实例,发现对于应用来说,版本迭代是最能有效提高应用在商店中排名的策略。因此,互联网应用程序只有充分、准确地挖掘用户需求,正确把握迭代方向,才能充分展现应用的活力与竞争力。

应用中的用户评论可以提炼出大量的需求信息[3],包括功能性、操作性、Bug 反馈等类别的需求。这些需求可以充分帮助应用进行迭代与更新[4]。因此,用户评论在软件需求的探索中显得格外重要。对这些评论数据进行智能分析与统计,通过对其描述内容进行处理,删除无效信息,将相似的内容或需求进行总结分类,可得到用户对该应用的显性或隐性需求。

因此,本文引入服务计算的架构,提出一种对用户反馈进行智能挖掘、分析与统计的方法,并将其设计成可重复使用的智能服务,从而更好地帮助应用软件进行开发与快速迭代。该方法首先从 IOS 平台爬取数据质量较高的用户评论数据,其次采用深度学习技术对其进行需求分类,最后对分类结果进行优先级排序,并将各类需求中优先级最高的反馈意见展示给用户。

在将深度学习方法引入到软件需求挖掘和分析过程后,除了可以充分分析用户的直接意见和漏洞反馈等显性需求外,还可以进一步挖掘出用户的习惯、喜好关联的隐性需求。尤其是对独立开发人员或者缺乏专业需求分析部门的小型开发团队来说,提供用户反馈收集与分析等功能的智能服务具有非常实际的价值。

1 相关工作

1.1 研究背景

用户反馈分析可以在构建与迭代软件过程中帮助开发人员获取用户实际感受,减少对用户需求的误解以避免设计缺陷。目前国内外许多研究者将用户评论数据作为挖掘用户反馈的重要信息来源[4-6]。这些研究从数据本身的特征出发,对需求

收稿日期: 2022-03-15; 修订日期: 2022-07-10

基金项目: 浙江省自然科学基金资助项目(LY21F020011, LY20F020027); 浙江省重点研发计划(2021C01162)

作者简介: 汪烨(1986—),女,安徽滁州人,浙江工商大学副教授,博士,主要从事软件工程、人工智能、自然语言处理、服务计算相关研究,E-mail: yewang@zjgsu.edu.cn。

引用本文: 汪烨, 周思源, 翁知远, 等. 一种面向用户反馈的智能分析与服务设计方法[J]. 郑州大学学报(工学版),2023,44(3):56-61.(WANG Y, ZHOU S Y, WENG Z Y, et al. An intelligent analysis and service design method for user feedback[J]. Journal of Zhengzhou University (Engineering Science),2023,44(3):56-61.)

类别进行定义,结合文本分析和机器学习等技术,在一定程度上提高了需求挖掘的效率和质量,提取到了有价值的用户反馈信息。但仍然存在一些不足: ①在需求服务领域工作的侧重面还是处于功能性需求,非功能性需求定义范围较为模糊; ②目前针对用户反馈分析的主要研究大多基于传统机器学习模型,其效果通常局限于单一适用领域,在不同领域的有效性有待验证。

1.2　方法框架

针对以上不足,本文整体方法框架包括:

(1) 对面向应用软件的用户评论数据进行需求类别定义[5-6]。分析软件用户的评论数据,定义了 9 种功能性和 1 种非功能性的知识类别。

(2) 面向应用软件用户评论数据的获取与构建。主要以用户评论的数据质量作为参考标准,选取了两大主流平台之一的 IOS 系统应用商店作为爬取对象,以排名、熟知度和用户数据质量为导向,选择了社交类、商务类、导航类、工具类、健康健美类、效率类下共 7 个热门 App,爬取了共计 5 万多条原生 App 用户评论数据后,构建基础资源库,制定 App 用户评论数据标注策略,安排相关领域标注者进行数据标注,构建实验数据集。

(3) 深度学习模型分类阶段。选择 TextCNN[7]、BiLSTM_Attention[8]、BERT[9] 3 种典型深度学习方法,从 *Precision*、*Recall* 和 *F*1-*measure* 方面对深度学习方法进行判定。

(4) 应用阶段。将上述方法获得的分类结果进行排序,应用至软件开发与迭代更新中,使得开发人员更方便地获取高频的用户需求。并将上述方法设计为一种有效的面向用户反馈的智能需求分析服务。

2　方法

2.1　应用需求分类

用户反馈表达了用户在功能性和非功能性方面的需求,但是绝大多数用户并非直接地表达对该软件的需求,往往掺杂着一些情感的表达,例如 “这个软件的广告真的很烦人”。综合考虑现有的需求分类[10-11],定义了 10 个需求类别。

(1) 期望需求(功能性需求): 迫切期望实现的功能的需求。

(2) 非期望需求(功能性需求): 与期望需求相背离的功能的需求。

(3) 观感需求: 主要描述了用户对产品外观和风格的期望。

(4) 易用性需求: 用户界面的易于使用,以及对面向用户的文档和培训资料等方面的需求。

(5) 性能需求: 用户在软件响应速度、结果精度、运行时资源消耗量等方面的需求。

(6) 安全性需求: 软件消除潜在风险的能力和对风险的承受能力。

(7) 可靠性需求: 用户在软件失效的频率、严重程度、易恢复性,以及故障可预测性等方面的需求。

(8) 可保障性需求: 用户在软件可配置性、可扩展性、可维护性、可移植性、可适配性等方面的需求。

(9) 合法性需求: 用户在是否合乎法律或法规方面的需求。

(10) 其他: 任何与功能性需求和非功能性需求无关的类型,比如对应用的评价、情感的表达。

2.2　应用需求数据集构建

调研已有的应用市场和系统平台,以排名、熟知度和用户数据质量为导向,选取了 IOS 应用商店中 7 个热门 App 的用户评论数据作为爬取对象构建数据集。

由于用户评论数据的随意性,存在数据格式不统一、表达多样化等特点,在初步爬取数据后,需要进行数据清洗与过滤: ①去除评论中的特殊符号,如“#”和表情符号等; ②对文中出现的英文数据进行小写转换和删除多余空格,方便后续处理; ③删除开发人员回复,在一定程度上减少输入文本的特征维度,提高需求分类的效率; ④删除重复性用户评论数据,主要是针对过短文本,如“太赞了!”。

经数据清洗、过滤后,对数据进行标注。根据需求类别设置,通过 5 名相关领域的标注者进行数据标注,并通过一致性检验,最终选取了 6 200 条数据作为实验数据。

2.3　基于深度学习模型的需求分类

2.3.1　基于 TextCNN 的分类模型

TextCNN 主要是通过不同长度的卷积核捕捉到不同个数的相邻词的相关特征以进行特征拼接。基于 TextCNN 的模型如图 1 所示。

用户评论语句 $S=\{w_1,w_2,\cdots,w_n\}$,n 表示用户评论语句长度,w_n 表示语句中第 n 个词。首先,将用户评论语句通过词编码形成词向量矩阵 $M\in\mathbf{R}^{n\times d}$,$d$ 表示嵌入层编码维度。

其次,使用不同尺度的卷积核进行特征提取:

$$ConvM_1=Conv1D_{k1}(M); \quad (1)$$

$$ConvM_2=Conv1D_{k2}(M); \quad (2)$$

$$ConvM_3=Conv1D_{k3}(M); \quad (3)$$

$$ConvM=Cont[ConvM_1,ConvM_2,ConvM_3]。 \quad (4)$$

式中: $Conv1D_k$ 表示卷积核尺度为 k 的一维卷积层; $Cont$ Ⅰ 表示拼接操作。将经过卷积操作后的结果进行拼接,得到 $ConvM \in \mathbf{R}^{n\times 3b}$,$b$ 表示单个卷积层的输出通道数。

经过卷积操作后,再进行最大池化操作,将全部的特征压缩至单维的向量空间,最后再输入由使用全连接网络和 softmax 激活函数组成的预测层进行预测,输出类别。

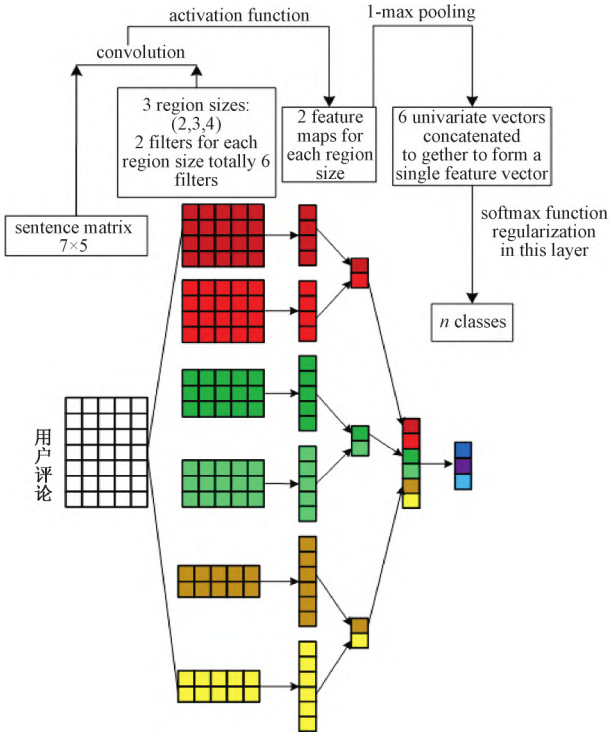

图 1 TextCNN 模型结构图

Figure 1 Model structure of TextCNN

2.3.2 基于 BiLSTM 和注意力机制的分类模型

本文使用双向长短期记忆网络(bi-directional long short-term memory,BiLSTM)处理序列类型的数据,可以充分考虑数据中的序列关系。注意力机制可以自动聚焦于对用户评论分类有决定性影响的词语,捕捉句子中最重要的语义信息。BiLSTM_Attention 模型如图 2 所示。

Word Embedding 层: 用户评论语句 $S=\{w_1,w_2,\cdots,w_n\}$,n 表示用户评论语句长度,w_n 表示语句中第 n 个词。将用户评论语句通过词编码形成词向量并输入 BiLSTM 层中进行特征提取:

$$h_i = [\overrightarrow{h_l} \oplus \overleftarrow{h_l}]。 \tag{5}$$

式中: $\overrightarrow{h_l}$ 表示正向 LSTM 的输出; $\overleftarrow{h_l}$ 表示逆向 LSTM 的输出; h_i 表示第 i 个词的输出; $\oplus$ 表示对应元素相加。

通过注意力机制捕捉语义信息,再通过 softmax 分类器进行预测类别。假设 $H=h_1,h_2,\cdots,h_r$ 表示 BiLSTM 层的输出,则有

$$M = \tan h(H); \tag{6}$$

$$\alpha = \text{softmax}(w^T M); \tag{7}$$

$$\gamma = H\alpha^T。 \tag{8}$$

式中: $H \in \mathbf{R}^{d^w\times T}$,$d^w$ 为词向量的维度,w 为需要训练的参数。句子的最终表示如式(9)所示:

$$h^* = \tan h(\gamma)。 \tag{9}$$

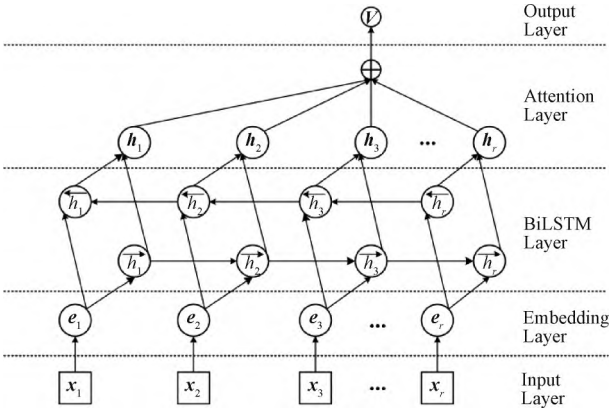

图 2 BiLSTM_Attention 模型结构图

Figure 2 Model structure of BiLSTM_Attention

2.3.3 基于 BERT 的分类模型

BERT 是基于双向的 Transformer 编码器实现的,其模型结构如图 3 所示,其中 $E_1,E_2,\cdots,E_n$ 表示字符的输入,用户评论语句的向量化主要通过 Transformer 实现。

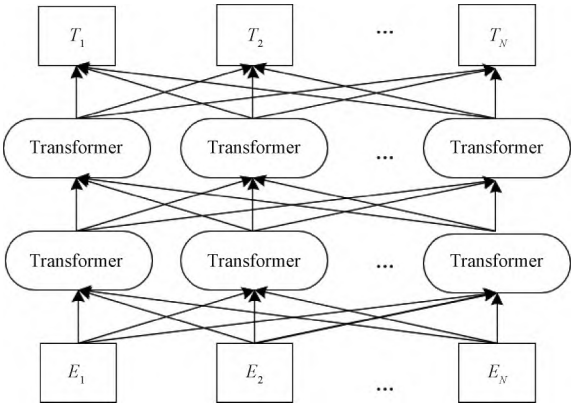

图 3 BERT 模型结构

Figure 3 Model structure of BERT

BERT 模型的输出形式主要有 2 种: 一种是字符级别的向量,即输入的用户评论语句每个字符对应的向量表示; 另一种则是句子级别的向量,即 BERT 模型输出的带[CLS]标识符的向量,该向量可以认为是代表整个句子的语义。

基于 BERT 的分类算法可以分为以下步骤:

步骤 1 构建应用软件文本训练集 $T=\{(x_1,$

y_1),(x_2,y_2),…,(x_n,y_n)},$i=1,2,\cdots,n$,其中(x_n,y_n)为第 n 个训练样本对应的文本向量,表示预处理之后第 n 个训练样本对应的类别。

步骤 2　在训练集上对 BERT 模型进行微调,BERT 模型输出得到训练集 T 对应的特征表示 $V=(v_1,v_2,\cdots,v_n)$,$i=1,2,\cdots,n$,其中 v_i 表示每位用户评论对应的句子级别的特征向量。

步骤 3　将特征向量 V 输入到 softmax 分类器中进行分类训练,得到分类模型。

2.4　应用阶段

通过上述步骤对应用软件的用户评论数据进行自动挖掘后,将这些步骤设计为智能分析服务以便更好地帮助开发人员进行软件更新迭代。该服务具体包含模块如图 4 所示。

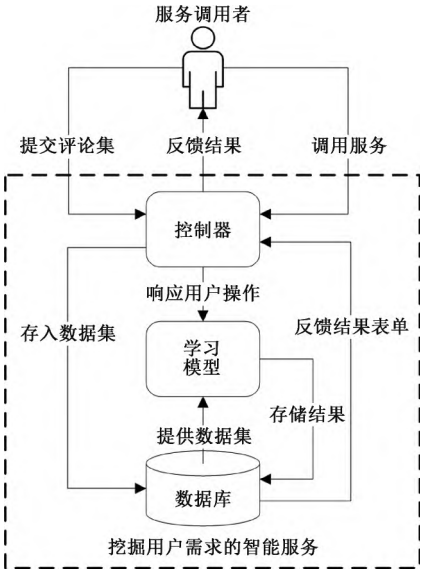

图 4　分析用户反馈的智能服务

Figure 4　Intelligetnt service that analyze user feedback

选择 Django 框架设计该智能服务应用,采用常用的 MVT 模式,将服务封装成接口。其中控制器对外负责接收用户对接口的调用请求以及返回调用后结果,对内负责与智能服务其他模块进行交互与调用。控制器包括注册、上传数据、数据清洗、处理、参考样例、返回结果等功能接口。Model 模块分为学习模型与数据库,学习模型与数据库交互,进行需求挖掘。数据库根据其核心逻辑建立 2 层数据表(评论数据表、推荐结果表),其字段设计如表 1 所示。

首先,该智能服务提供用户注册、权限确认等基础交互功能,Model 模块提供 get_data(获取评论数据),其中输入数据对应数据库中的评论数据表(见表 1)。其次,通过 data_clean(数据清洗接口)将用户提供评论数据集进行清洗,并将其关键内容存入数据库中的评论数据表的字段中,通过 data_processing(数据处理接口)确定每条评论的分类和相关度。最后,根据上文的需求分类结果,以每类需求的相关度和赞同数作为基准,取 relevancy 值和赞同数最高的前 5 个评论的内容存入 example,一起保存在推荐结果表,并标明 id。

表 1　数据库字段设计

Table 1　Database field reference

表单名	字段名	内容
评论数据表	answer_id	各条评论 id
	user_id	评论集提交用户 id
	content	评论内容
	agree_num	赞同数
	kind_id	标明分类结果
	relevancy	分类结果的相关度
推荐结果表	kind_id	各个分类 id
	kind_name	分类名称
	example	基于相关度保留其数值最高的评论标题
	example_id	保存被推荐的评论 id

在呈现给用户结果时,本文设计了 2 种方式:第 1 种是将评论数据表中的各条数据包含其分类结果直接反馈给用户,便于用户进行详细的查看;第 2 种则从方便、简易的角度,将推荐结果表中数据输出,包含各个分类中前 5 个优先级最高的需求类,以及相关度最高的前 5 个评论样例供用户参考。

3　实验与分析

3.1　实验环境与参数设置

本实验是基于深度学习框架 PyTorch 实现的。PyTorch 是一个开源的 Python 机器学习库,具有灵活易用、速度快等优点。在本实验中,模型的参数设置如下:学习率 learning rate 为 0.005、dropout rate 为 0.5、优化器为 Adam、批次大小为 64、最长文本长度为 64、迭代次数为 30。TextCNN 模型中 3 个卷积核的大小分别为 2、3、4,输出通道数设置为 256,BiLSTM_Attention 隐藏层的维度为 128,BERT 隐藏层维度为 768。

3.2　数据集设置

模型的数据集包括训练集(4 960 条数据)、验证集(1 420 条数据)、测试集(1 420 条数据)。目前,面向中文应用领域的需求分类一般都是研究者自己构建数据集。而比较有权威的数据集,例如 PROMISE 的软件需求库和分类库,数据集的规模比较小。自采集数据集一般采用人工标注的方式进行,并且由于数据集的质量、内容等不同,形成的需

求知识类别也不同。

3.3 评价标准

需求分类的目的是在类别范围内,识别每个用户评论文本的类型。本文采用常用分类任务的评价指标 *Precision*、*Recall* 和 *F*1-*measure* 作为评估指标[12-13]。

3.4 结果与分析

各模型的分类效果对比如图5所示。由图5可知,BERT分类模型在指标 *Precision* 上效果最好,达到了0.67。这是因为BERT层数深,可表征的函数空间大,能更好地获取文本词向量和句向量,更好地抓取文本上下文特征。而在综合效果上TextCNN表现最好,其调和指标 *F*1-*measure* 达到了0.62,这是由于TextCNN对于小范围的特征提取适用性更强,故对用户评论这种短文本来说,TextCNN网络有更好的适应性,能够更好地兼顾核内上下文信息。BiLSTM_Attention在3个模型中效果最差,其 *Precision* 为0.51,*Recall* 为0.52,*F*1-*measure* 为0.51。这是因为对于短文本来说,该模型的结构复杂,增加了复杂性计算。

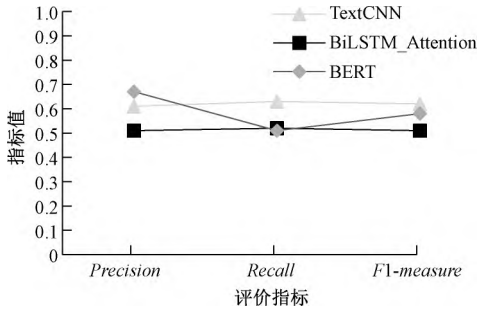

图5 分类效果对比

Figure 5 The comparison of classification result

通过实验可知,BiLSTM_Attention与BERT都擅长对文本语言做双向处理,但BERT不需要特定结构修改,只需要添加一个额外的输出层进行fine-tune,就可以在各种各样的下游任务中取得不错的表现。BiLSTM_Attention更适合捕捉长距离的依赖关系,对于短文本来说,该模型的结构较为复杂,增加了计算量。TextCNN共享卷积核的特点帮助其优化了计算量,更适合已经训练好的词向量及小范围的特征提取。

因此,当用户反馈数据量大且用户对时间需求要求不高的情况下,推荐使用BERT模型,但是其对硬件资源消耗很大,且收敛速度较慢;当用户反馈数据量较小时,BERT和TextCNN的效果相差不大,而后者由于计算量小,非常适合小样本且需要快速分析的模型设计。

4 结论

本文提出了一种针对用户反馈的智能分析和服务设计的方法。该方法通过对应用软件中的用户评论数据进行智能分析,有效地获取用户的需求反馈。

(1)将应用软件的用户评论中蕴含的潜在需求划分为功能性需求、非功能需求等10个子类别,更加细致地帮助开发人员获取不同维度的用户需求,帮助其进行应用软件的迭代与更新。

(2)TextCNN、BiLSTM_Attention、BERT分类模型对用户需求进行挖掘,发现BERT模型在 *Precision* 上的表现最好,TextCNN模型在 *Recall* 以及 *F*1_*measure* 上的表现最好。

(3)提出一种面向用户反馈的智能分析与服务设计方法,指导开发人员更系统地对用户需求进行挖掘,并将其封装成服务包以帮助进行软件的开发与迭代更新。

在未来的工作中,考虑更多类别下的应用软件以及用户评论,且对于复杂评论中的情感表达,需要进一步分析才能准确找出用户的真正意图;在需求分类的基础上,考虑下一步的研究工作方向,并形成面向后续任务的智能计算服务。

参考文献:

[1] LEE G, RAGHU T S. Product portfolio and mobile apps success: evidence from app store market [J]. Journal of Management Information Systems, 2014,31(2):133-170.

[2] HENZE N, BOLL S. Release your app on Sunday eve: finding the best time to deploy apps [C] //Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services. New York: ACM, 2011: 581-586.

[3] SCHNEIDER K. Focusing spontaneous feedback to support system evolution [C] //19th International Requirements Engineering Conference. Piscataway: IEEE, 2011: 165-174.

[4] CHEN N, LIN J L, HOI S C H, et al. AR-miner: mining informative reviews for developers from mobile app marketplace [C] //Proceedings of the 36th International Conference on Software Engineering. New York: ACM, 2014: 767-778.

[5] PANICHELLA S, SORBO A D, GUZMAN E, et al. How can i improve my app? Classifying user reviews for software maintenance and evolution [C] //IEEE International Conference on Software Maintenance & Evolution. Pisca-taway: IEEE, 2015: 281-290.

[6] 王莹,郑丽伟,张禹尧,等.面向中文APP用户评论数

据的软件需求挖掘方法[J].计算机科学,2020,47(12):56-64.
WANG Y, ZHENG L W, ZHANG Y Y, et al. Software requirement mining method for Chinese APP user review data[J].Computer Science, 2020, 47(12):56-64.

[7] KIM Y. Convolutional neural networks for sentence classification[EB/OL].(2014-08-25)[2021-12-09]. https://arxiv.org/abs/1408.5882.

[8] ZHOU P, SHI W, TIAN J, et al. Attention-based bidirectional long short-term memory networks for relation classification[C]// Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: Association for Computational Linguistics, 2016: 207-212.

[9] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[EB/OL].(2017-12-06)[2021-12-09]. https://arxiv.org/abs/1706.03762.

[10] 贾一荻, 刘璘. 中文非功能需求描述的识别与分类方法研究[J]. 软件学报, 2019, 30(10): 3115-3126.
JIA Y D, LIU L. Recognition and classification of non-functional requirements in Chinese[J]. Journal of Software, 2019, 30(10): 3115-3126.

[11] KURTANOVIC Z, MAALEJ W. Automatically classifying functional and non-functional requirements using supervised machine learning[C]//2017 IEEE 25th International Requirements Engineering Conference. Piscataway: IEEE, 2017: 490-495.

[12] HUANG Q, XIA X, XING Z C, et al. API method recommendation without worrying about the task-API knowledge gap[C]//Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering. New York: ACM, 2018: 293-304.

[13] RAHMAN M M, ROY C K, LO D. RACK: automatic API recommendation using crowdsourced knowledge[C]//2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering. Piscataway: IEEE, 2016: 349-359.

An Intelligent Analysis and Service Design Method for User Feedback

WANG Ye[1], ZHOU Siyuan[1], WENG Zhiyuan[2], CHEN Junwu[1]

(1.School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou 310018, China; 2.Department of Computer Science and Engineering, University of Nebraska-Lincoln, Lincoln 68508, U.S.)

Abstract: In order to process user comments, an intelligent analysis and service design method was proposed. The user comments from multiple iOS Apps, were mined intelligently. The potential user requirements were identified from these comments. Firstly, user requirements were summarized and classified into ten categories. Secondly, user comments were crawled, cleaned, and labeled to form a classification dataset. Then, TextCNN, BiLSTM_Attention, and BERT were used to processed these data. The classification results were prioritized. Finally, the method was packaged into a reusable intelligent service module for remote calls. Experimental results showed that the TextCNN model had the best overall performance, while the BERT model had the best performance in precision. The BERT model optimized the training process through parallel computing and could be extended to large-scale projects. Therefore, when with large data volumes, and the priority of accuracy over time, the BERT model was recommended. Conversely, the TextCNN model would be recommended when dealing with user needs with small data and short time consumption.

Keywords: requirement analysis; service design; service computing; deep learning; user feedback