

基于软件需求规范的项目级复用研究

巴元秀,赵逢禹,刘 亚

(上海理工大学 光电信息与计算机工程学院,上海 200093)

摘 要:目前的软件复用技术主要围绕软件代码的复用进行研究。而随着开源项目的增多,基于待开发项目的需求文档分析,实现项目级的复用就显得非常有价值。当开发人员获取项目的软件需求后,通常需要对其分析并构建解决方案,然后进行设计与实施。如果能根据项目的软件需求找到相似的历史项目进行复用,可以大大节省项目设计与实施时间。因此,在现有的项目级复用研究基础上,该文提出一种基于需求分析的项目级复用技术 PR-REQ。该方法首先分析历史开源项目,给出了开源项目的领域信息提取算法,代码的功能操作序列提取算法以及数据模型信息的提取算法;然后给出了针对待开发项目需求文档的领域信息提取算法,用例的功能操作序列提取算法以及数据模型信息的提取算法;最后构建需求文档与历史项目的相似性度量方法,从而找到最相似的项目进行项目级复用。为了验证该方法的有效性和准确性,从 Github 上下载了 8 个类别的开源项目进行实验,实验结果表明该方法对项目级复用具有较好的实用价值。

关键词:项目级复用;需求分析;开源项目;领域相似分析;功能相似性分析;数据模型相似性分析

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2022)02-0094-07

doi:10.3969/j.issn.1673-629X.2022.02.015

Research on Project-level Reuse Based on Software Requirement Specification

BA Yuan-xiu, ZHAO Feng-yu, LIU Ya

(School of Optical-electrical & Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Most software reuse technologies are focus on coding level reuse. With the increasing number of open source projects, it is quite valuable to realize project-level reuse based on the requirement document analysis of the projects to be developed. When a developer acquires a project's software requirements, it is often necessary to analyze them, build a solution, and then design and implement it. If a similar historical project can be found for reuse according to the software requirements of the project, the design and implementation time of the project can be greatly saved. Therefore, we present a project-level reuse technology PR-REQ based on requirement analysis. This method firstly analyzes the history of open source projects, and gives the extraction algorithms of the domain information, the functional operation sequence of the code and of data model for the projects. Then, the algorithm of extracting domain information of requirement document, functional operation sequence of use case and data model information are given. Finally, the similarity measure method of requirement document and history project is constructed to find the most similar projects for project-level reuse. In order to verify the validity and accuracy of this method, eight categories of open source projects were downloaded from Github. The experimental results show that this method has good practical value for project-level reuse.

Key words: project-level reuse; requirement analysis; open source projects; domain similarity analysis; functional similarity analysis; data model similarity analysis

1 PR-REQ 项目复用方法

软件项目复用对减少开发工作量和成本、提高软件项目质量具有重要影响^[1]。在软件开发中,通过复用技术可以充分利用已有的开发成果,从而提高软件

开发的效率、降低开发成本^[2]。同时,通过复用高质量的已有的开发产品,避免了重新开发过程中可能引入的错误,从而提高软件的质量。

Gharehyazie 等人^[3]利用代码克隆检测工具

收稿日期:2021-03-06

修回日期:2021-07-06

基金项目:国家密码管理局“十三五”密码发展基金理论课题(MMJ20180202)

作者简介:巴元秀(1996-),女,硕士研究生,研究方向为软件项目复用;赵逢禹,博士,教授,CCF 会员(15341M),研究方向为计算机软件与软件系统安全、软件工程与软件质量控制、软件可靠性;刘 亚,博士,副教授,CCF 会员(A3546M),研究方向为信息安全、密码学等。

Deckard 研究项目内部以及跨项目的克隆,研究发现代码克隆非常普遍,从同一项目的几行代码到跨项目的代码片段之间都检测到不同程度的相似。Zhang Yun 等人^[4]研究 GitHub 上的项目复用,发现含有相似的 README 文件的软件项目存在相似性。由此可以看出不论是代码级别还是项目级别,都存在着大量的复用。

目前最常用的软件复用技术主要是代码的复用。软件开发人员进行项目开发的过程中发现,开发同类型的软件应用时会实现相似或相同的功能,相应的代码实现也是相似的。代码克隆技术便是代码复用方法中最原始且最常用的技术,一般表示为开发人员通过代码搜索技术从开源项目中找到自己所需要的源代码并复制到自己的项目中。文献^[5]提出一种代码级别的复用技术,该技术利用提取工具将源代码的内容生成代码摘要,将用户输入的文本与代码摘要进行搜索查询,最后获取相似的代码进行复用。

近年来,随着开源生态的完善,人们更多地关注项目级别的复用研究。Xu 等人^[6]研究跨项目复用,利用历史开源软件项目的描述文档和源代码,将其与待开发项目的功能进行相似性度量并推荐相似的软件项目。Thung 等人^[7]利用相似的项目会共享相似的第三方库这一原理,提取开源项目使用的库,利用对库的相似性匹配相似的项目实现项目级别的复用。Nguyen 等人^[8]提出生态系统概念,由历史开源项目、库以及项目之间的相互依赖关系组成,将开发人员对待开发项目的创建、修改等行为与生态系统分别构建成图,利用图的相似性算法选取相似的项目进行复用。

文献^[6~8]在项目级复用研究中仍有不足之处。文献^[6]主要研究了项目开发过程中跨项目的代码复用技术,是编码阶段的复用方法。文献^[7]是从项目使用的库方面考虑复用,文献^[8]是根据历史开源项目、库以及待开发项目三者之间的依赖关系考虑复用。以上的研究都还没有实现从待开发项目的需求分析考虑项目级别的复用。

文献^[9]提出一种基于需求规范文档进行代码的功能特征复用的方法,该方法首先从需求规范文档中提取出功能特征关键词,然后将提取出的关键词与代码库中代码的功能特征关键词进行相似性匹配,最后根据相似的功能特征进行代码复用。文献^[9]从需求规范文档的功能特征出发,搜索可复用的代码,仍然属于项目局部功能的代码复用。

在软件工程实践中,当开发人员获取项目的软件需求后,通常需要根据需求文档中的问题领域、用例描述^[10]、数据模型 E-R 图搜索开源项目库中的相似软件项目。如果能从需求文档的问题领域、用例描述以

及数据模型方面找到相似的历史项目进行复用,可以大大节省项目设计与实施时间。而用自动化的方法找到相似的历史项目,是一件复杂的工作。因此本文提出了一种基于软件需求规范的的项目级复用方法 PR-REQ (Project Reuse based on Requirements Specification)。该方法首先分析历史开源项目,构建算法提取历史项目核心信息,包括项目的领域信息、代码的功能操作序列以及数据模型信息。然后针对待开发软件项目的需求文档,研究提取问题领域、用例的功能操作序列以及数据模型等信息的方法。最后分别构建领域相似性度量、功能操作序列相似性度量以及数据模型相似性度量算法,加权计算得到最终的相似性度量值并按由大到小排列。从 GitHub 上下载了 8 类 Java web 项目构造实验,对文中提出的方法进行实证。

2 PR-REQ 项目复用方法

项目级别的复用是对历史项目在软件架构、功能实现、数据模型、设计与编码等多方面的复用,也是软件复用中最高级别的复用。为了实现项目级别的复用,图 1 给出了该方法的操作流程图,分为 3 步,它们分别是历史开源项目信息的提取、待开发项目的需求文档分析以及项目相似性度量计算。

(1) 历史开源项目信息的提取。

为了复用历史项目,需要对开源历史项目进行数据分析与特征提取。在每个项目中,大部分都有源代码和描述文档。源代码包含了项目的功能信息和数据模型信息。描述文档通常给出了项目的功能介绍、用法以及如何安装或部署。

a. 描述文档领域信息的提取。

分析描述文档中的项目功能介绍,利用自然语言的方法处理文本主题分析提取项目的领域信息。

b. 代码中功能信息的提取。

采用静态代码分析技术解析历史项目的代码,提取项目的各功能操作,构建项目的功能操作序列。

c. 数据模型信息的提取。

从数据库配置文件或源代码中提取所使用的数据库的表名、列名等信息,构建数据模型。

(2) 待开发项目的需求文档分析。

a. 领域信息的提取。

在软件复用时,属于同一个领域的项目被复用的可能性更高。这里通过自然语言处理方法,对需求文档进行分析,提取待开发项目的领域信息。

b. 功能操作信息的提取。

为了从项目的功能方面进行相似性度量分析,需提取需求文档中的用例,根据用例中的活动和参与活动的对象,构建待开发项目各功能的功能操作序列。

c. 数据模型信息的提取。

在数据模型 E-R 图中提取与待开发项目有关的实体信息。

(3) 项目相似性度量。

a. 领域相似性度量。

为了能够根据需求文档中的领域信息在历史开源项目中寻找领域方面相似的项目,本文从开源历史项目的描述文档和待开发项目的需求文档中各提取若干个主题,利用主题的相似性计算领域的相似度。 $a = (a_1, a_2, \dots, a_n)$ 定义为从待开发项目的需求文档描述中提取的 n 维主题向量, $b = (b_1, b_2, \dots, b_n)$ 定义为从开源历史项目的描述文档中提取的 n 维主题向量,领域相似度计算如公式(1)所示。

$$S_{\text{领域}}(a, b) = \frac{\sum_{i=1}^n (a_i \cdot b_i)}{\sqrt{\sum_{i=1}^n (a_i)^2} \times \sqrt{\sum_{i=1}^n (b_i)^2}} \quad (1)$$

b. 功能操作相似性度量。

为了从功能操作方面比较待开发项目的需求文档和历史开源项目之间的相似性,本文构建了功能操作序列的相似性度量方法。待开发项目的需求文档中的功能操作序列定义为 $W = \{W_1, W_2, \dots, W_m\}$, 其中 $W_i (i = 1, 2, \dots, m)$ 代表一组功能操作序列的集合。历史开源项目中提取的功能操作序列定义为 $T = \{T_1, T_2, \dots, T_n\}$, 其中 $T_j (j = 1, 2, \dots, n)$ 代表一组功能操作序列的集合。序列 W 中序列的个数记为 m , 序列 T 中序列的个数记为 n , 采用余弦相似度^[11] 计算 W 中任一个序列 W_i 与 T 中任一个序列 T_j 的相似度值记为 $S(W_i, T_j)$, 最后根据公式(2)计算最终的相似度值。

$$S_{\text{功能}}(W, T) =$$

$$\frac{\sum_{i=1}^m \text{Max}(S(W_i \cdot T_1), S(W_i \cdot T_2), \dots, S(W_i \cdot T_n))}{\text{Max}(n, m)} \quad (2)$$

c. 数据模型相似性度量。

为了更准确地进行项目复用,本文对数据模型也进行分析,构建数据模型相似性分析。历史开源项目中的数据模型定义为 $A = \{A_1, A_2, \dots, A_n\}$, 其中 $A_j (j = 1, 2, \dots, n)$ 代表由表名、列名信息组成的一组数据模型集合,集合 A 中的个数记为 n 。待开发项目的需求文档中提取出的数据模型定义为 $C = \{C_1, C_2, \dots, C_m\}$, 其中 $C_i (i = 1, 2, \dots, m)$ 代表由需求文档中表名、列名信息组成的一组数据模型集合,集合 C 中的个数记为 m 。采用余弦相似度^[11] 计算集合 A 中任一组数据模型 A_j 与集合 C 中任一组数据模型 C_i 的相似度值记为 $S(C_i, A_j)$, 最后根据公式(3)计算集合 C 与 A 的相似度值。

$$S_{\text{数据}}(C, A) =$$

$$\frac{\sum_{i=1}^m \text{Max}(S(C_i, A_1), S(C_i, A_2), \dots, S(C_i, A_n))}{\text{Max}(n, m)} \quad (3)$$

d. 计算候选项目的最终相似得分。

在分别计算上述三种相似度值后,最后采用公式(4)进行加权计算得到最终的相似性度量分值。

$$\begin{cases} S = \alpha * S_{\text{领域}} + \beta * S_{\text{功能}} + \gamma * S_{\text{数据}} \\ \alpha + \beta + \gamma = 1 \end{cases} \quad (4)$$

其中, α 代表领域信息的权重, β 代表功能操作序列的权重, γ 代表数据模型的权重。

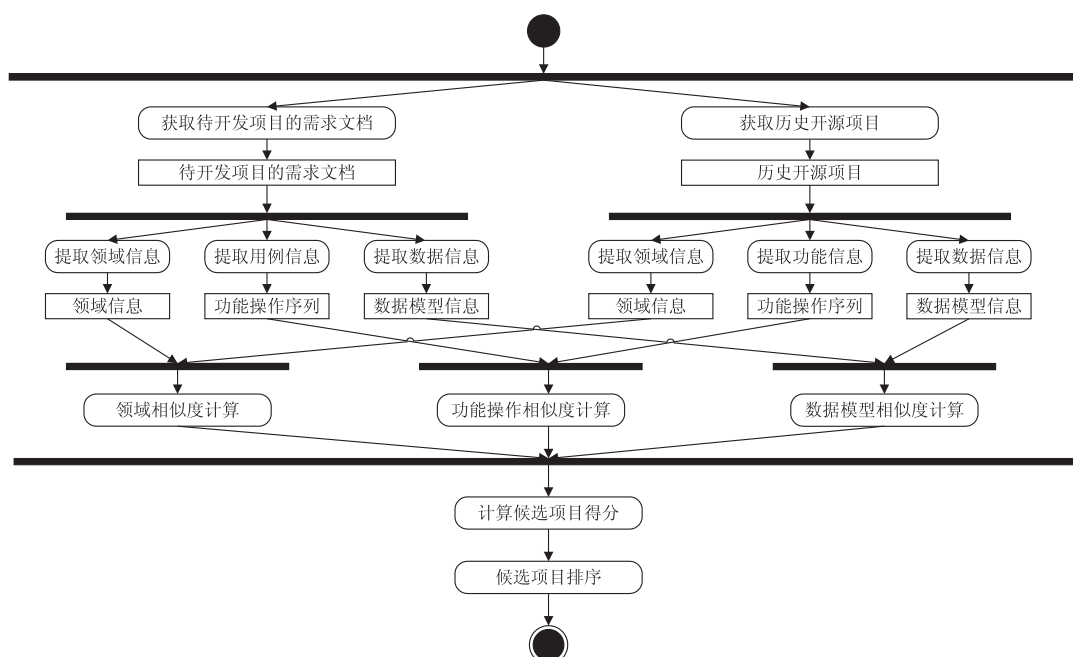


图 1 PR-REQ 方法流程图

3 PR-REQ 方法的关键技术

PR-REQ 项目复用方法中的关键技术主要是从历史开源项目与待开发项目的需求文档中提取相关特征,然后计算这些特征的相似度并基于特征相似性推荐可复用的历史项目。其中的关键技术主要有历史开源项目信息的提取和待开发项目需求文档信息的提取。

3.1 历史开源项目信息提取

大多数开源项目中都有描述文档和源代码。为了分析历史开源项目,本文对项目中包含的描述文档和源代码进行数据分析和特征提取。

(1) 领域信息的提取

描述文档是描述软件项目信息的文档,它一般包含软件的基本功能、简要的使用说明、代码目录结构说明等信息。文中使用斯坦福主题模型工具提供的基于概率模型的 LDA 主题模型算法^[12]对描述文档中的项目功能介绍部分进行领域信息的主题提取。

在对领域描述文本进行数据提取等相关工作之前,一般要进行文本的预处理。文本的预处理也就是将需要进行分析的文本通过一定的方式转换成方便处理的结构化的数据形式,可以提高文本处理的准确性。需求规范文档大多由中文文本构成,因此文中主要使用的预处理方式是中文文本的预处理。结合对于中文文本的特征进行分析,在研究的过程中,文本预处理方式主要包括语料库清理、分词、词性标注和停止词过滤。

① 语料库清理。

语料库的清理主要是清理和删除一些非法语言文字等不良数据。

② 分词、词性标注。

词是构成语句的基本单元,分析语句前需要先分词,将文本中词切分出来作为特征值,是自然语言处理比较重要的一步。中文分词就是将句子中汉字序列切分成词集合。相对于英文而言,中文分词要复杂得多。

③ 停止词过滤。

在普通文本中含有标点符号、介词、语气词等,这些词对理解文本没有实际意义,应从分词结果中去除,这些词称之为停用词。去停用词可以省存储空间,减少停用词对理解语句造成的噪音,降低文本维度,可以提高处理文本的效率和准确率。

文献[13]认为对一篇科技文献,提取主题数量在5到6个时分析效果最好。借鉴该研究的结论,对软件需求与历史项目文档的主题数量为5。

(2) 代码中功能操作序列的提取。

如何在软件项目代码中提取主要功能的操作序列

是本文的关键算法。对于不同的软件项目,由于其软件架构不同,其功能的操作序列提取方法也有差异,但都可以通过对配置文件与源代码的静态分析获得。文中以 Java web 项目为研究对象,给出特征提取算法。Java web 项目的功能操作信息主要体现在项目的页面文件上,因此文中针对 Java web 项目的页面文件提取功能操作序列。为了从页面文件中提取项目的功能操作,需要利用 JSOUP 解析器^[14]从应用程序的入口页面开始对其进行数据解析,提取对应页面的功能操作,构建功能操作序列。在提取功能操作时,在 web 页面中,操作功能主要分为静态和动态两类。`<input type="button" value="注册" onclick="window.location.href('register.jsp')"/>`就是一个静态页面的跳转,可以从中提取出“注册”功能;`<input type="submit" width="100" value="提交订单" name="submit" border="0" style="..."/>`是一个动态页面的跳转,在点击“提交订单”的同时会调用 saveOrder 方法,从后台数据库中调取数据,可以从中提取出“提交订单”功能。算法 1 给出了代码功能操作提取算法。

算法 1: 代码功能操作提取算法。

输入: Java web 项目的页面文件集合 $P = \{p_1, p_2, \dots, p_n\}$, 项目的配置文件 config。

输出: 项目的功能操作序列 $T = \{T_1, T_2, \dots, T_m\}$, 其中 $T_i (i = 1, 2, \dots, m)$ 为一组功能操作序列的集合。

处理:

1. 初始化队列 Queue, $G = \{V, E\}$, $V = \Phi$, $E = \Phi$;
2. 从 config 中找到项目的入口页面文件,不失一般性假设为 p_1 ;
3. 将 p_1 作为访问的第一个页面文件入队列 Queue;
4. while (Queue 非空)
 - {
 - CurrentPage = deQueue (Queue); // 队头元素出队;
 - $V = V \cup \{ \text{CurrentPage} \}$;
 - 利用 JSOUP 解析器提取 CurrentPage 页面中的功能操作信息 f_i 和与之对应的跳转页面 p_i , $p_i \in P$;
 - 构造 G 的有向边,加入到集合 E 中,记为 $E = E \cup \{ \langle \text{CurrentPage}, p_i \rangle, f_i \}$;
 - if (p_i not in V)
 - p_i 入队列 Queue;
 - }
5. 在图 G 中,从节点 p_1 开始按照深度优先算法遍历路径上的功能操作 f_i 构造功能操作序列 T_i ;
6. 输出功能操作序列 T , 结束算法。

(3) 数据模型的提取

算法以 Java web 项目的代码文档集合作为输入,采用一种轻量级的查询提取工具 SQL 提取器^[15],该

工具使用 AST 过程内的字符串解析,能够对代码中的数据模型进行静态分析,提取出带有元信息的 SQL 语句,通过对 SQL 语句进行分析,输出含有数据库的表名、列名信息的集合。文中给出了利用 SQL 提取器进行数据模型提取的过程,见算法 2。

算法 2:数据模型的提取。

输入:项目的代码文档集合 $B = \{b_1, b_2, \dots, b_n\}$ 。

输出:表名、列名信息组成的数据模型集合 $A = \{ \langle \text{TableName}_i, (\text{colname}_{i1}, \dots) \rangle \}$ 。

处理:

1. 初始化集合 A ;
2. 将 b_1 作为访问的第一个代码文档;
3. for each (CurrentFile in B)

{

利用 SQL 提取器提取出 CurrentFile 中 SQL 语句的表名和 Insert 语句中的列名,记为 $\langle \text{TableName}_i, (\text{colname}_{i1}, \dots) \rangle$;

$A = A \cup \{ \langle \text{TableName}_i, (\text{colname}_{i1}, \dots) \rangle \}$;

}

3.2 待开发项目需求文档信息提取

对于待开发项目,需要在需求文档中提取领域信息、功能操作信息、数据模型。其中领域信息的提取方法与历史开源项目中提取领域信息的方法一致。数据模型的提取是对需求文档的数据模型 E-R 图中的实体关系进行分析。

功能信息的提取是对需求文档中的用例进行特征提取,提取用例的活动,构建待开发项目的功能操作序列。限于篇幅,文中主要介绍对需求文档中的功能信息进行提取的方法。

用例由用例名称、描述、角色、编号、前置条件以及主事件流等组成,文中给出了用例功能操作序列的提取算法,该算法使用需求文档中的用例集合 $R = \{r_1, r_2, \dots, r_n\}$ 作为输入,提取出与系统进行交互的用例活动,构建功能操作序列 $W = \{W_1, W_2, \dots, W_m\}$ 作为输出,见算法 3。

算法 3:用例功能特征的操作序列的提取。

输入:需求文档中的用例集合 $R = \{r_1, r_2, \dots, r_n\}$ 。

输出:功能操作序列 $W = \{W_1, W_2, \dots, W_m\}$, 其中 $W_i (i = 1, 2, \dots, m)$ 为一组功能操作序列的集合。

处理:

1. 初始化集合 W ;
2. 将 r_1 作为访问的第一个用例;
3. for each (CurrentUse in R)

{

根据 CurrentUse 中的事件流的交互过程,提取出参与用例活动的操作和对应的活动对象,构建功能操作序列集合记为 W_i ;

$W = W \cup W_i$;

}

4. 输出操作序列 W , 结束算法。

4 实验研究

在软件项目级别的复用研究领域中,目前尚没有找到包含待开发项目的需求文档、历史项目的源代码数据集,以及它们之间相似度的度量参考标准。因此为了验证文中提出的 PR-REQ 方法的准确性,文中构造了一个实验,采用人工方法和 PR-REQ 方法分别对需求文档进行分析,将 PR-REQ 方法找出的 Top N 的相似软件项目与人工找出的 Top N 的相似软件项目进行一致性和包含性比较。

4.1 数据集

文中从 GitHub 上下载了 958 个 Java Web 项目,这些项目包含学校管理类、企业类、竞赛类、网购类、游戏类等八类项目,在这些项目中同时具有需求描述和项目代码的项目数共 68 个。开源软件项目数及其类别见表 1。

表 1 开源软件项目数及其类别

项目类别	项目总数	同时具有需求描述和项目代码的项目数
学校管理类	253	18
企业类	145	14
竞赛类	62	7
网购类	74	12
游戏类	86	3
视频类	134	6
院类	120	4
银行类	84	4
总数	958	68

4.2 实验步骤

基于 PR-REQ 方法开发了一个项目复用推荐程序,该推荐程序首先从数据库中读取历史开源项目和待开发项目的需求文档。然后分别提取开源项目 and 需求文档中的领域信息、功能操作序列以及数据模型信息并存储于数据库中。最后通过调用相似度量程序对需求文档和历史开源项目进行相似度量,利用相似度得分机制将候选项目按得分由大到小排列,并将候选项目的结果存储到数据库中。

4.3 验证 PR-REQ 方法的准确性

为了验证该方法的准确性,文中从百度文库中下载了 5 个学校管理类需求文档和 5 个企业类需求文档,对其从①到⑩进行编号。为了便于实验中对相似项目进行排序,将表 1 中同时具有需求描述和项目代码的 68 个项目进行从 1 到 68 编号。其次,分别采用人工方法和 PR-REQ 方法对这 10 个需求文档进行分析,从 68 个项目中找出相似的软件项目。人工方法是由作者共同对每个需求文档进行分析,从领域、功能操

作以及数据模型三个方面进行数据分析,找出与之相似程度最高的3个项目,对其进行相似度排序并记录3个项目对应的编号。PR-REQ方法则是利用开发好的项目复用推荐程序,对这10个需求文档进行特征提取与分析,同样找出与之相似程度最高的3个项目,对其进行相似度排序并记录3个项目对应的编号。最后,分别对人工选取的项目和PR-REQ方法选取的项目进行一致率和包含率分析,验证文中提出的PR-REQ方法的准确性。

表2 人工方法和PR-REQ方法选取的相似项目的编号

需求文档编号	人工方法选取的相似项目排序	PR-REQ方法选取的相似项目排序
①	3,7,9	3,9,7
②	12,2,3	12,2,3
③	18,14,6	18,14,5
④	7,9,10	9,7,10
⑤	8,9,1	8,9,1
⑥	20,28,30	20,29,21
⑦	19,27,32	19,32,27
⑧	22,24,19	20,19,24
⑨	29,21,26	29,24,26
⑩	30,19,31	30,19,31

表3 一致率和包含率分析

	%		
评价指标	Top 1	Top 2	Top 3
一致率	80	40	30
包含率	80	80	83.3

由表3可以看出,针对这10个需求文档,人工分析与PR-REQ分析推荐的第一个历史项目有80%相同,在推荐一个最相似的项目时,一致率与包含率意义相同。Top 2一致率为40%,Top 3一致率为30%,可以看出,随着推荐的项目越多,一致率会有所下降,这一结果也符合预期;而对于包含率,Top 1、Top 2与Top 3都不低于80%,也就是说利用PR-REQ方法找出的1到3个相似项目与人工找出的1到3个相似项目虽然无法在相似程度上保持一致,但包含率在80%以上,这说明文中提出的PR-REQ方法具有较高的准确性,对于项目复用具有实用价值。

5 结束语

软件项目需求分析是开发人员开发一个新项目时的第一阶段,目前尚没有一种项目复用方法可以根据待开发项目的需求文档进行项目级复用。针对这一问题,文中提出了一种基于软件需求分析的项目级复用方法PR-REQ,该方法从项目的主题领域、功能操作信息以及数据模型三个方面对待开发项目的需求文档和开源历史项目进行特征提取和数据分析,最后构建相

定义1:一致率:PR-REQ系统找出的项目与人工找出的项目一致的比率。

定义2:包含率:PR-REQ系统找出的项目包含在人工找出的项目中的比率。

表2是针对每个需求文档,利用人工方法和PR-REQ方法找出的相似软件项目对应的编号。然后对表2中的编号进行一致率和包含率分析,分析结果见表3。

似性度量方法进行相似项目推荐。基于文中提出的方法,随机选取了企业类和学校管理类的需求文档各5个来构造实验验证其准确性。

实验结果表明,文中提出的基于需求分析的项目级复用方法在寻找相似的软件项目方面具有较高的准确性,该方法对于开发人员在项目开发的初期阶段具有重要作用。但是本实验只取了68个历史项目文件,针对两类共10个项目需求文档进行了分析,实验样本数量仍然偏少,为了证实该方法的实用性,还需下载更多的项目进行分析。另外对于非Java web项目,还需基于历史项目的代码进一步构建算法提取功能操作序列与项目的数据模型。

参考文献:

- [1] HUANG Jianglin. The analysis of project factors for software development effort and quality: the impact and estimation [D]. Hong Kong: City University of Hong Kong, 2016.
- [2] 冯厚伟, 杜鹏宙, 刘 勇. 软件复用技术及其在软件开发中的应用[J]. 电子技术与软件工程, 2019(6): 51.
- [3] GHAREHYAZIE M, RAY B, FILKOV V. Some from here,

- some from there; cross-project code reuse in GitHub [C]//2017 IEEE/ACM 14th international conference on mining software repositories (MSR). Buenos Aires: IEEE, 2017: 291–301.
- [4] ZHANG Y, LO D, KOCHHAR P S, et al. Detecting similar repositories on GitHub [C]//2017 IEEE 24th international conference on software analysis, evolution and reengineering (SANER). Klagenfurt: IEEE, 2017: 13–23.
- [5] ISLAM M, IQBAL R. SoCeR: a new source code recommendation technique for code reuse [C]//2020 IEEE 44th annual computers, software, and applications conference (COMP-SAC). Madrid, Spain: IEEE, 2020: 1552–1557.
- [6] XU W, SUN X, HU J, et al. REPERSP: recommending personalized software projects on GitHub [C]//2017 IEEE international conference on software maintenance and evolution (ICSME). Shanghai: IEEE, 2017: 648–652.
- [7] THUNG F, LO D, LAWALL J. Automated library recommendation [C]//2013 20th working conference on reverse engineering (WCRE). Koblenz: IEEE, 2013: 182–191.
- [8] NGUYEN P T, DI ROCCO J, RUBEI R. CrossSim: exploiting mutual relationships to detect similar OSS projects [C]//2018 44th Euromicro conference on software engineering and advanced applications (SEAA). Prague: [s. n.], 2018: 388–395.
- [9] CHEEMA S M, ADNAN M, BAQIR A, et al. A recommendation system for functional features to aid requirements reuse [C]//2020 3rd international conference on computing, mathematics and engineering technologies (iCoMET). Sukkur, Pakistan: [s. n.], 2020: 1–4.
- [10] TIWARI S, RATHORE S, SAGAR S, et al. Identifying use case elements from textual specification: a preliminary study [C]//2020 IEEE 28th international requirements engineering conference (RE). Zurich, Switzerland: IEEE, 2020: 410–411.
- [11] SOYUSIAWATY D, ZAKARIA Y. Book data content similarity detector with cosine similarity (case study on digilib. uad. ac. id) [C]//2018 12th international conference on telecommunication systems, services, and applications (TSSA). Yogyakarta, Indonesia: [s. n.], 2018: 1–6.
- [12] LIU X, ZHANG Z, LI B, et al. Keywords extraction method for technological demands of small and medium-sized enterprises based on LDA [C]//2019 Chinese automation congress (CAC). Hangzhou, China: [s. n.], 2019: 2855–2860.
- [13] 王婷婷, 韩 满, 王 宇. LDA 模型的优化及其主题数量选择研究——以科技文献为例 [J]. 数据分析与知识发现, 2018, 2(1): 29–40.
- [14] WANG Jie, YANG Shuo, WANG Yuezhi, et al. The crawling and analysis of agricultural products big data based on Jsoup [C]//2015 12th international conference on fuzzy systems and knowledge discovery (FSKD). Zhangjiajie: IEEE, 2015: 1197–1202.
- [15] NAGY C, CLEVE A. A static code smell detector for SQL queries embedded in java code [C]//2017 IEEE 17th international working conference on source code analysis and manipulation (SCAM). Shanghai: IEEE, 2017: 147–152.