*论文题目：Java API 文档缺陷自动化检测方法

*作者：周宇，古睿航，

*单位：南京航空航天大学

联系方式：zhouyu@nuaa.edu.cn, ruihang_gu@nuaa.edu.cn

*文章发表信息：(论文集，请提供论文集名称，出版社，出版时间；杂志，请提供杂志名称，年，卷，期，页码)
Proceedings of the 39th International Conference on Software Engineering
Pages 27-37, Buenos Aires, Argentina — May 20 - 28, 2017, IEEE Press Piscataway, NJ, USA ©2017, ISBN: 978-1-5386-3868-2

*原文链接地址：

http://dl.acm.org/citation.cfm?id=3097373

*正文：

应用程序编程接口(Application Programming Interface)，即 API，一般通过一些开放的函数接口提供功能支持，其底层功能的具体实现对开发人员透明，使开发人员更关注于业务逻辑，进而提高开发的效率，是软件复用的一种重要方式。API 文档是开发人员了解、使用 API 的重要参考，从而可以根据文档正确地使用相应的 API。如 JDK（Java Development Kit，Java 软件开发工具包）中就提供了相应的编程接口文档。在 Java 语言标准中，这样的 API 文档通常是以 Javadoc 文档规范、以 Annotation 注解的形式，标注在源代码中，能够通过一些文档提取工具如 Javadoc、Doxygen 等将注解提取出来，并生成对应的接口文档。开发者在进行相关接口调用时，通过接口文档，可以快速清楚地获取诸如接口功能、接口参数标准及其约束条件等信息，达到快速、准确使用 API 的目的。

开发者通过 API 文档了解接口的约束条件，从而正确使用该接口。高质量的接口文档，应该清楚地描述出其接口被调用时需要满足的相关约束条件，其中主要是对参数的约束、以及违反约束而抛出的异常。但由于人工撰写文档可能存在错漏、文档和代码更新进度不一致等原因，API 文档的描述和代码功能存在不一致的情况。模糊甚至错误的 API 文档，会造成软件开发者理解困难甚至理解错误。例如，在 javax.swing.JTabbedPane 类的 addTab(String title,Component component)方法中，如果传入参数 component 是一个 Window 类的实例，该方法会抛出 IllegalArgumentException 异常。但是在 addTab()方法对应的 Javadoc 文档中对此并没有详细说明。JTabbedPane 类中其他几个实现类似功能的方法也存在这种文档描述不一致的状况。在 java.awt.JobAttributes 类中的 setPageRanges(int[][] pageRanges)方法中，如果传入参数 pageRanges 为 null，则会抛出 IllegalArgumentException 异常，而该方法文档中也没有对此情况进行说明。在 java.awt.event.InputEvent 类中，其方法 getMaskForButton(int button)对应文档中表示，如果参数 button 小于 0 或者大于某个数，会抛出 IllegalArgumentException 异常。但是从代码中分析，该异常在 button 等于 0 时也会抛出，此时我们发现该文档对参数约束进行的描述是错误的。开发者使用 API 进行软件开发的时候，如果参照上述模糊甚至错误的文档，在没有其他资料辅助的情况下，很容易引入 bug，在程序运行中抛出异常，造成程序崩溃等。

针对这个问题，南京航空航天大学、密德萨斯大学和苏黎世大学的研究人员提出了一种自动化的解决方案用来检测 API 文档描述的缺陷。该方法处理过程如图 1 所示：

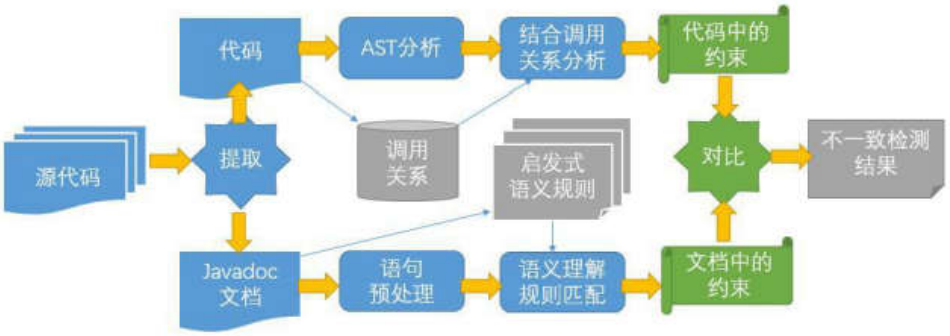

图 1：不一致检测方法

方法可分为如下 4 个步骤：

1) 从源代码中分别提取 API 接口的执行代码及其对应的 API 文档；

2) 对实现 API 功能代码部分进行静态分析，包括 AST 分析、调用关系库建立、引入调用关系进行解析等，然后对约束条件进行提取，生成逻辑形式；

3) 对 API 文档进行自然语言的分析，理解相关条目的描述语义，句法分析、词性分析

和依存语法分析等，提取约束条件，并生成逻辑形式；

4) 对二者中提取出的约束条件进行逻辑验证，最终检测出不一致问题。

其中，从代码中抽取约束条件，利用静态分析的方法，对 API 方法体代码构造抽象语法树，进行解析和并获取其控制流条件，从而达到提取其中的约束条件的目的。具体的，可将约束条件看成两部分组成：造成的代码错误和触发错误的条件。通过制定规则搜索目标语法树结构定位代码错误，回溯过程中对错误的触发条件进行收集，从而提取约束条件。由于代码可能法正在调用方法，所以引入调用关系的分析，以便完整地提取 API 约束。本文所研究的 Java API 文档，具体指的是在 Java 源代码中符合 Javadoc 规范、以注解形式存在的、对 API 相应的功能以及其约束条件进行描述的文档。对于文档对约束条件的描述，本文关注文档中的相关条目的描述语句。尽管 Javadoc 具有一定的结构形式，其中对于约束的描述主要还是以自然语言作为表达方式。因此，本文采用一种启发式的方法，引入自然语言处理机制，对文档进行语义理解，从中提取约束条件信息。

由于分别从文档和代码中提取出的约束条件在形式上可能存在一定的变化，如表达式顺序、否定的逻辑关系等，直接将文档约束转变为对应的代码表达式来进行比对可能产生误判断。对此，本方法提出，分别将从文档和代码中提取出的约束条件转换成统一的中间形式，再对其逻辑进行对比分析。现有的文档形式化方面的研究表明，一阶逻辑适合于对自然语言的分析和再现，本方法使用一阶逻辑（First-Order-Logic）对分别从文档和代码中提取的约束进行展现，并使用可满足性模型理论对其进行验证。为验证方法的有效性和可行性，我们针对 JDK1.8 的 API 文档做了相应的实验，该方法检测出了 1158 个文档描述缺陷，准确率达到 81.6%，召回率达到 82.0%。

本方法的详细描述论文发表在 ICSE 2017 上，英文标题为“Analyzing APIs Documentation and Code to Detect Directive Defects”，相比较于自然语言属性的文档而言，API 的程序代码经过了大量的测试或者复杂的演化更新，而文档在这一过程中往往被忽略，因此我们在论文中我们做了一个假设，当代码和文档不一致出现的时候，认为是文档存在缺陷（defects）。论文的作者包括南京航空航天大学周宇副教授，硕士生古睿航和黄志球教授，密德萨斯大学大学陈韬略博士，苏黎世大学 Sebastiano Panichella 博士和 Harald Gall 教授。

作者简介：周宇（1981-），男，博士，副教授，江苏徐州人，主要研究方向为软件演化、软件体系结构、软件自适应技术等;古睿航（1991-），男，硕士研究生，四川内江人，主要研究方向为软件演化、数据挖掘等。

说明：*为必填项。