Grant Proposal

Jincheng He
Department of Computer Science
University of Southern California
jinchenh@usc.edu

I. Introduction

Developing software with the source code open to the public is a common way of developing software. To improve the quality of open source software, we conduct this research. This research investigates how the different purposes of commits impact the quality of software open source software (OSS). By identifying these impacts, we will establish a new set of guidelines for code and development process, thus achieving better quality.

II. Previous work

Previous researchers have revealed when, where, how and what the developers contribute to the projects and how they impact software quality. However, there is little work on purpose-focused categorizations and how different categories impact software quality.

III. Our Plan

A. Stage One

Previous researchers didn't study the correlation between the change type and the code as well as their impact on the software quality. In this stage, we start with establishing categorization of commit changes in open source software repositories. We evaluate the quality of those changes by obtaining quality metrics from static analysis tools. To assess the correlation between the quality and the categories, we plan to train a machine learning model, in addition to standard mathematical correlation analyses.

B. Stage Two

In this stage, although we have categorized the commits changes, we need further insight on the distinguishing different categories. This is because it has been shown that high-level categories have overlaps with each other. In this stage, we will be working on removing the ambiguity of the categories by analyzing the code changes within the commits rather than the commit messages and manual categorizing. Once this is done. We will investigate the correlation between the categories and changes in code to reveal whether they have close connection and how those changes impact software quality.

C. Stage Three

In the final stage, with the refined categories, we will start construct guidelines for developers on how they can better contribute to open source software when they make different type of changes. In addition, we will create an index to indicate how different code patterns impact the software quality. We will conclude this research by completing and validating these two aspects.

IV. Feasibility

We show the feasibility of this project by the following three aspects:

- Data: Open-source software and version control systems provide sufficient meta-data for analysis.
- Tool: Existing tools, such PMD, SonarQube, FindBugs and CAST provide various quality metrics.
- Techniques: The machine learning and natural language methods required in this research already exist.

With all above provided, we believe this plan will succeed in 3 5 years. The midterm milestone is a reasonable high prediction accuracy from the machine learning model. The final milestone is the completion of the new systematic coding standard and development guidelines.

V. Benefits

Once the plan work out, we will be able to provide guidelines on how open-source software developers contribute to projects to attain better quality. In addition, results in the second stage will allow us to provide more reliable coding standards and will improve the overall code quality. Improved quality will either help to reduce the cost or improve the service quality of the vendors.

VI. Intellectual Advancement

The first goal of this research is to practically improve the quality of open-software by providing guidelines for developers about how to better contribute to projects and to improve code quality. We believe this will be an intellectual improvement in software engineering since it changes the way of how people think, code and develop software.