Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management
Kuala Lumpur, Malaysia, March 8-10, 2016

Investigating Software Detection Methods

Kiran Khatter
School of Engineering & Technology
Ansal University
Gurgaon, India
kirankhatter@gmail.com

Arvind Kalia
Department of Computer Science
Himachal Pradesh University
Shimla, India
arvkalia@gmail.com

***Abstract*— A defective module not only increases the development time and development cost but also increases the maintenance time and maintenance cost. According to the available literature survey, many systems failed due to schedule and time budget overruns. Therefore a software defect detection technique is needed to identify those software modules that are very likely to include defects and thereby improves the software quality by contributing in the efficient removal software defects. The main objective of the paper is to help software developers in identifying the software defects based on the various software metrics using various classification and machine learning techniques. In this paper, we are performing empirical classification comparison on 5 real world datasets.**

Keywords— ***Software defect detection, Clustering Techniques, Machine Learning Techniques, Software Metrics***

I. INTRODUCTION

Computer Software have been become an essential part in everyday life and have been used for different applications ranging from business and personal computing applications to real-time applications. The main objective of software development is to develop high quality software with the expected functionalities. In context of software engineering, software quality refers to the satisfaction of functional and non-functional requirements. A functional requirement refers to the operations performed by the software and its components whereas Non-functional requirement refers to the quality attributes such as reliability, maintainability, availability etc. of the system and the degree of satisfaction of non-functional requirements determines the success/failure of software systems [1].There are also various approaches to achieve these attributes with emphasis to improve the software quality by preventing the introduction of faults. These software defect detection approaches help in finding the defective modules in the software. Testing at the various level of software development process is the traditional way to identify the defects, but when the project size increases in terms of lines of codes(LOCs) and complexity, it becomes difficult and expensive to find the defects with the use of traditional testing techniques. If defects or errors are not identified in the initial phase, these may likely to creep in the later phases such as design and implementation phase of the software development process. Repair of such errors and defects increases the time and cost of software development process [2][3][4]. National Institute of Standards and Technology did study on various projects of U.S. and concluded that software errors cost U.S. economy about $59.5 billion yearly. So testing and debugging are persistent during the software development process. [5].Thus sooner we detect defected software components; it lowers down the cost& time to develop the software and hence improves the reliability of the software. Thus Defect prediction is imperative to accomplish the software quality.

In order to perform comparison of different software fault detection classifiers, we are using various performance metrics such as probability of detection, accuracy, precision, G-mean, F-measure. This paper also represents the comparison of various different classification models using ROC curves.

This paper is organized as follows: Section 2 explains the related work reported by various researchers for software fault detection. Section 3 describes the data sets used in the study to perform comparison. Section 4 specifies the different classifiers and a detailed comparison between various classifiers. Section 5 represents the experimental set up and the results interpretation followed by conclusion of the study.

978-1-4673-7762-1/16/$31.00 ©2016 IEEE