



LEARNING TO RANK AND CLASSIFICATION OF BUG REPORTS USING SVM AND FEATURE EVALUATION

¹S.Rajeswari, ²S. Sharavanan, ³R.Vijai and ⁴RM. Balajee

¹PG Scholar, Department of CSE,

²Professor & Head, Department of CSE,

^{3,4} Assit Professor, Department of CSE, Annapoorana Engineering College, Salem.

Email: rajeswariselvaraj89@gmail.com

Submitted: May 27, 2017

Accepted: June 15, 2017

Published: Sep 1, 2017

Abstract- *When a new bug report is received, developers usually need to reproduce the bug and perform code reviews to find the cause, a process that can be tedious and time consuming. A tool for ranking all the source files with respect to how likely they are to contain the cause of the bug would enable developers to narrow down their search and improve productivity. This project introduces an adaptive ranking approach that leverages project knowledge through functional decomposition of source code, API descriptions of library components, the bug-fixing history, the code change history, and the file dependency graph. Given a bug report, the ranking score of each source file is computed as a weighted combination of an array of features, where the weights are trained automatically on previously solved bug reports using a learning-to-rank technique. I applied SVM (Support Virtual Machine) to classify the bug reports to identify, which category the bug belongs to. It helps to fix the critical defects early. The ranking system evaluated on six large scale open source Java projects, using the before-fix version of the project for every bug report. The experimental results show that the learning-to-rank approach outperforms three recent state-of-the-art methods. In particular, proposed method makes correct recommendations within the top 10 ranked source files for over 70 percent of the bug reports in the Eclipse Platform and Tomcat projects.*

Index terms: Learning to rank, SVM, Preprocessing, CF(collaborative Filtering)