Tackle 'em bugs! Managing the issue overflow in large-scale software engineering

Borg, Markus

2014

[Link to publication](#)

Total number of authors:
1

Tackle 'em bugs! Managing the issue overflow in software engineering

Markus Borg

Software maintenance is one of the most expensive phases in software engineering, as shown by studies from the 1970s until today. Some practitioners consider operational and maintenance costs to require as much as 85-90% of their project budgets [3]. Thus, making software maintenance more efficient has potential to considerably reduce overall development costs. Issue management is a central part of software maintenance, revolving around issue reports collected in a central repository. Large organizations receive 10,000s of issue reports yearly, resulting in challenging prioritization decisions and work allocation.

Our work aims to support issue management by providing automated decision support [2]. A fundamental approach in our work is to make use of the rich information in issue repositories, i.e., to let previous work guide future decisions in issue management. Issue reports typically contain both text describing the experienced problem and other pieces of (often nominal or ordinal) information such as the version of the faulty software, the severity of the issue, and details on the execution environment. Also, issue reports are not independent, but are parts of complex networks of related problems [1].

Our contributions address several parts of issue management. First, we have used Information Retrieval (IR) techniques to detect duplicated issue reports. Second, we have applied machine learning to train classifiers (ensemble learning under stacked generalization) to automate assignment of issue reports to appropriate development teams, and to prioritize among issues. Third, we have replicated a study showing that (k-means) clustering issue reports based on their textual content can lead to clusters of issues with significantly different average resolution times. Fourth, we are currently evaluating support for change impact analysis by a recommendation system combining IR and network analysis.

References

[1] Borg, M., Pfahl, D., and Runeson, P. Analyzing Networks of Issue Reports (2013) In *Proc. of the Eur. Conf. on SW Maint. and Reeng.*, pp. 79-88.

[2] Borg, M. and Runeson, P. Changes, Evolution and Bugs: Recommendation Systems for Issue Management, (2014) *Recommendation Systems for Software Engineering*, Springer, pp. 477-509.

[3] Erlikh, L. Leveraging Legacy System Dollars for E-Business, . (2000) *IT Professional*, 2(3), pp. 17-23.

Markus Borg, Dept. of Computer Science, Lund University, markus.borg@cs.lth.se