Department of Computer Science,
University of Saskatchewan

July 22, 2020

The Guest Editors
Journal of Systems and Software

Re: Submitting an extended version of our earlier work

Dear Guest Editors,

We would like to submit an extended version of our earlier work on ranking some existing good clone detection tools based on their capability to predict co-change candidates during software evolution. Our earlier work titled "**Evaluating Performance of Clone Detection Tools in Detecting Cloned Cochange Candidates**" was published in IEEE 14th International Workshop on Software Clones (IWSC), 2020. We evaluated the performance of six good clone detection tools based on detecting cloned co-change candidates in that work. Our goal was to practically verify whether a tool that is good in detecting cloned fragments from software systems is also good in detecting cloned co-change candidates or not. The key points of our earlier published paper are:

- ➢ Used six open-source software systems as subject systems.
- ➢ Evaluated the performance of six clone detection tools.
- ➢ We also investigated and reported two possible reasons for the difference in the performance of clone detection tools while we are using them to predict co-change fragments.

We have made the following improvements on our earlier work for submitting to the Journal of Systems and Software (Special Issue on Software Clones).

- ✓ **Adding more clone detectors in the study:** In the earlier implementation, we have applied six clone detection tools, five of them (Deckard, ConQAT, iClones, NiCad, and SimCAD) have been reported as very good tools for detecting Type-1, 2, 3 clones. We added one text similarity-based tool (Simian) as the sixth tool in the earlier implementation. We added six additional implementations of clone detection techniques by using three additional tools to extend the work. New tools added are CloneWorks, CCFinder, and Duplo. CloneWorks have been reported as a fast and flexible clone detector for large-scale near-miss clone detection experiments. CloneWorks provides options to modify its configuration files, affecting the source code processing mechanism while detecting the clones. The configuration files are important to target specific types (1, 2, or 3) of clone using this clone detection tool. We have applied four types of different configuration files to detect Type-3 pattern, Type-3 Token, Type-2 Blind, and Type-1 clones by using the CloneWorks tool. Our variation of the configuration files provided four additional sets of detected clone results. CCFinder is known as a multilinguistic token-based code clone detection technique for large scale source code. The inclusion of CCFinder enriched the variation of detected clone fragments in the extended study. To make more comparison of the performance of type-1 clones in detecting co-change candidates, we added Duplo in our study.

- ✓ **Adding more subject systems:** In the earlier version of the study, we added six software systems four of them are written in Java and two of them are written in C programming language. To increase the generalizability of the study we have added two more software systems written in C. Therefore, we now have four software systems written in each of the programming languages C and Java. The subject systems have also increased the diversity in size and application domain compared to our earlier study.

- ✓ **Additional two research questions:** Our earlier investigation evaluated only two research questions to find the comparison scenario of clone detectors in detecting co-change candidates and the reasons behind these performance variations of those tools. We have added two more research questions in the extended study to determine the relation of the types of clone fragments and source code processing techniques of the clone detection as the additional reasons for these difference in performance.

- ✓ **Adding statistical significance test to verify the obtained results:** We performed The Wilcoxon Signed-Rank Test to verify the hypothesis that the F1 Scores of a tool which has obtained a higher rank in our extended study are significantly different (better) than the F1 Scores of the tools which have got lower ranks. Here, F1 Scores of each tool contains eight values obtained in all the eight subject systems. Results of the significance test and its relevant analysis show that both the configurations (Pattern and Token) of CloneWorks clone detection tool for detecting type-3 clones are performing significantly better compared to more than 72% other clone detectors used in this study. Deckard and CCFinder are also better compared to more than 55% of the other tools. The remaining tools are not showing satisfactory performance. Our extended paper contains the details of the significance test results.

- ✓ **Updating different sections in the paper to incorporate our new findings:** We have updated different sections such as Abstract, Introduction, Methodology, Experimental Result, Discussion, Conclusion and Future Works by adding our new findings from our extended research.

We believe that our extended version contains at least 60% more content than our earlier work. We have also updated the textual presentation in the new paper. Using Grammarly's (https://www.grammarly.com/) premium subscription, we found no textual similarity (plagiarism) of our paper to the published paper in IWSC 2020.

Sincerely,
Md Nadim
Manishankar Mondal, and
Chanchal K. Roy