Paper Review 《Automated Localization for Unreproducible Builds》

Paper Info

Zhilei Ren, He Jiang, Jifeng Xuan and Zijiang Yang

ICSE 2018

Main Contribution

This paper is aimed at the localization for unreproducible builds. The approach in this work is fancy. They propose an automated framework called RepLoc to localize the problematic files for unreproducible builds. The contributions of this work include:

- Feature a query augmentation component
- Design a heuristic rule-based filtering component that narrows the search scope
- Integrate these two components with a weight file ranking module

The core insight of this work is to combine the static information and the dynamic aspect of the build process. They reduce the problem of localization to the information retrieval. This view is brand-new and this work is a good exploration in this topic.

Main Work

The approach is totally based on the paradigm of the traditional information retrieval. In the heuristic rule-based filtering component, each file is being checked to find out whether it contains the non-deterministic elements, such as the usage of random number or system date-time. The authors suppose that these elements are the cause root of the unreproducible builds.

They also notice that some files might not be used in the actual run. Hence, the dynamic information of the build process should be utilized. In the file ranking module, the similarity between the specific file and the augmentation query is combined with the score of non-deterministic degree obtained in the heuristic rule-based filtering component.

At last, all of the files are sorted according to the scores. The authors discuss the influence of the weight in the ranking module and claim that the performance is not sensitive to the weight.

Future Work

The fault localization is reduced to information retrieval and text mining in this work. Although this approach is fancy, the performance of the tool is not striking enough. Because of the limit of mining methods, the accuracy of localization is not high enough.

The problem can be perfectly solved by some other approaches. For example, the system calls can be monitored and we can find out the inconsistencies based on the information of actual run. In this work, they select several patterns to model the randomness of the code, including the usage of random number and system date-time. This method can not cover the unreproducible cases completely.