SE Software Engineering

HUMBOLDT-UNIVERSITÄT ZU BERLIN

Bachelor/Master Thesis Topic

Deduplication of Bugs through Input Space Analysis and Failure Circumstance Learning

Motivation and Background

In software systems, multiple bug reports often correspond to similar or identical underlying issues, leading to wasted effort in redundant debugging and patching processes. This thesis explores a novel approach to automated bug deduplication by leveraging the input space of a system and identifying failure circumstances using diagnosis tools such as **Avicenna** [1].

The central idea is to characterize the input conditions that trigger specific failures. By employing Avicenna, the system can analyze initial failure-inducing inputs and derive an abstract representation of the input space associated with each bug. These representations capture the essential properties of the inputs that lead to failure, effectively serving as a "*fingerprint*" for each bug. The deduplication process then involves comparing these fingerprints to detect whether different bug reports share the same failure input space.

Goals

The goal of this thesis is to develop a methodology for automating bug deduplication by analyzing the input spaces that lead to failures in software systems. By leveraging tools like Avicenna, we can systematically identify the failure circumstances associated with a bug, represented as abstract properties of the input space. These representations serve as unique fingerprints that encapsulate the root causes of failures. The deduplication process will compare these fingerprints to detect whether multiple bug reports stem from the same or similar underlying issues, thus reducing redundancy in bug tracking and resolution.

Description of the Task

The specific tasks are:

- *Failure Circumstance Learning***:** Leveraging diagnostic tools [1] to automatically generate and refine explanations for failure-inducing inputs.
- *Bug Similarity Analysis:* Defining and implementing metrics or algorithms to compare input space fingerprints and identify duplicates.
- *Evaluation and Validation:* Applying the approach to real-world software systems to measure its effectiveness in reducing duplicate bug reports.

Research Type

Theoretical Aspects:	****☆
Industrial Relevance:	****☆
Implementation	*****

Prerequisite

The student should be enrolled in the bachelor of computer science program, and has completed the required course modules to start a bachelor thesis (or similar).

Skills required

Programming skills in Python, understanding of, or willingness to learn, the software engineering methods needed for the project.

Contacts

Martin Eberlein, M.Sc.
Mail: martin.eberlein@hu-berlin.de
Software Engineering Group,
Institut für Informatik,
Humboldt-Universität zu Berlin

References

[1] Martin Eberlein, Marius Smytzek, Dominic Steinhöfel, Lars Grunske, and Andreas Zeller. 2023. Semantic Debugging. In Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2023).