Review 4

1/31/12

CS6V81.502

Seungtack Baek

**PRMiner: Automatically Extracting Implicit Programming**

**Rules and Detecting Violations in Large Software Code**

Implicit programming rules can be easily violated since they are hard to be documented by programmers. Also when they are violated it is “likely” to introduce defects to the software. According to Li and Zhou, using PR Miner (Programming Rule Miner), we can extract such programming rules, thus find the violations.

PR miner extracts closed programming rules by looking at the patterns of element within the project. To put it simply, if an element A appears frequently with B, PR miner might conclude that there is a programming rule that where A appears, B should appear as well. Also, it calculates the confidence, which is proportional to the ratio of the times A appears with B to the times A appears regardless of B. Now, then can sort the rule by their confidence, thus examine the ones with highest confidence first. Through this method, they were able to find about 60 violations, of which 16 were confirmed as bugs.

To me, 16 unfound bugs seem very significant. However, as the authors noted on this paper, this method is suited for ordinary PC machine. I wish that the authors would talk about what company wants to do, if they really wish to find hard-to-find bugs. Also, from my experience, everyone has their own coding style. Say if one person does 90% of the work, and the other does the remaining 10%, this might lead to false positive. But, I think this is inherent to the “internal” program rule finder.

Other usage of this could be that it can be used to detect wrong usage of framework. Whereas the rules for using framework are oftentimes documented and adhering to those rules are desirable (either architectural or performance reason), sometimes developer (junior developers, especially) can misuse the framework. This PR miner method can find the violation and alert the developer.

**Question:**

1. To be honest, I did not find any of the rules in the paper to be “hard-to-documented,” especially the rules states that “one function should follow the other” Detecting the violation is great, but is not clear documentation the first job?