STEVENS INSTITUTE OF TECHNOLOGY 1870

Ph.D. DISSERTATION DEFENSE

Candidate: Yutong Zhao
Degree: Doctor of Philosophy
School/Department: School of Systems Engineering and Enterprises

Date: Wednesday, March 15th, 2023
Time/Location: 10:00 a.m. / https://stevens.zoom.us/my/luxiao66
Title: A Data-driven and Architecture-centric Approach for Identifying and Analyzing Real-life Software Performance Issues

Chairperson: Dr. Lu Xiao, School of Systems Engineering and Enterprises

Committee Members: Dr. Onur Asan, School of Systems Engineering and Enterprises
Dr. Mehmet Gunes, School of Systems Engineering and Enterprises
Dr. Ying Wang, School of Systems Engineering and Enterprises
Dr. Hui Wang, Department of Computer Science, School of Engineering & Sciences

ABSTRACT

Software performance is a critical quality attribute that determines the success of a software system. And performance issues have been recognized to have their roots in poor architectural design decisions, which are often made before system implementation. Despite the advances in both research directions in performance engineering and software architecture, there remains gaps between the two. First, there lacks a large-scale benchmark dataset of real-life performance issues to support related research regarding how real-life performance issues are caused and resolved, specifically in the architecture and design level. Second, there is limited knowledge regarding whether and how the architectural connections among software design elements contribute to performance issues. This dissertation makes three contributions towards to closing these gaps.

First, it contributes a performance issue detection framework which combines heuristic linguistic patterns and cutting-edge machine/deep learning techniques. We derived 80 project-agnostic linguistic patterns that practitioners frequently used to describe performance issues by manually analyzing 980 real-life performance issue reports. These linguistic patterns serve to construct effective learning features to automatically tag unlabeled performance issue reports in the software issue tracking systems. This framework has shown significantly improved accuracy and transferability across different issue tracking platforms, compared to cutting-edge baseline approaches, such as BERT.

Second, it contributes comprehensive and in-depth empirical knowledge regarding of how real-life performance issues are caused, resolved, and tested. We investigated 570 issues from 13 open-source projects across different problem domains. We summarized eight recurring root causes of performance issue with corresponding resolutions. A non-trivial portion (27%) of these performance issues require design-level optimization. We also revealed four typical patterns of design-level performance optimization that practitioners should be aware of when looking for performance optimization solutions. From the engineering perspective, we found that only 15% of these issues involve test code revision, highlighting the

potential weakness of inadequate performance verification in regression testing. From the economic perspective, we found that design-level performance optimization often requires more investment, but can lead to long-term benefits, such as software maintainability. This study provides valuable insights for practitioners to understand and treating performance issues.

Third, it contributes an architecture modeling approach, named *Butterfly Space* modeling, to bridge the gap between software architecture and performance analysis during software maintenance and evolution. The rationale of *Butterfly Space* modeling is that the performance of one method impacts and is impacted by all the other methods that have architectural connection with it. Case studies of three open-source Java projects showed that when a method contains performance problems, 33% to 60% of the methods that connects to it, directly or indirectly, may also contain optimization opportunities. Compared to dynamic profiling, *Butterfly Space* modeling could be more efficient and cost-effective to identify performance optimizations opportunities based on the connections among methods. We believe that Butterfly Space modeling has great potential for investigating the architectural roots of performance issues.