Mobile-App Analysis and Instrumentation Techniques Reimagined with DECREE

Yixue Zhao
University of Southern California
yixue.zhao@usc.edu

Nenad Medvidovic
University of Southern California
neno@usc.edu

I. Meta Talk Abstract

With the emergence of smartphones, mobile devices have become the dominant computing platform over the past decade, resulting in millions of available mobile apps that have attracted large research attention. Current mobile computing research has focused extensively on three threads: *analysis techniques* that analyze the apps' implementation artifacts statically to extract information of interest (e.g., security vulnerabilities [1]); *instrumentation techniques* that improve targeted aspects (e.g., performance [2]) of an app by directly modifying the app's implementation; and *auxiliary techniques* that analyze external information associated with mobile apps to learn useful lessons (e.g., our recent work that assessed prefetching and caching opportunities [3]).

However, those techniques' constituent components are difficult to extract and reuse outside their original tools, their evaluation results are hard to reproduce, the tools themselves are hard to compare, and there is a lack of adoption of those techniques in practice. To identify the reasons behind this phenomenon, we contacted the authors of several techniques and studied existing work along with their implementation, and discovered several common themes: (1) there is no established communication channel between researchers and app developers, thus the techniques may not meet the exact needs in practice and may violate real-world assumptions; (2) research techniques often have steep learning curves, making them difficult to adopt in practice; (3) research techniques are often evaluated in limited settings, rendering any claims insufficiently convincing for app developers to adopt; and (4) existing techniques are usually designed as one-off solutions without proper modularity, making them hard to reproduce, reuse, or customize.

To address the above problem and facilitate open science in the mobile computing domain, we believe a new infrastructure is needed with standard baselines to guide future techniques. To that end, we propose *DECREE*, an approach aiming to transform how research in the mobile arena is conducted in order to produce reusable, practical, and reproducible research techniques that are easier to adopt in practice.

DECREE is an infrastructure that provides a comprehensive baseline for *D*eveloping, *E*valuating, *C*omposing, *R*eusing, *E*volving, and *E*xploring research techniques in the mobile computing domain, with three major components:

① Motivated by the significant but often neglected reuse opportunities in existing work, we have designed a microservice-based *reference architecture* [4] for analysis and instrumentation techniques based on existing techniques and our own experience in the mobile domain. The reference architecture is intended to be comprehensive in scope but simple enough to adopt and tailor in order to guide the design of future techniques. We have shown examples of how existing techniques can be migrated to our reference architecture, to enable easy reuse and adoption that is hard to achieve with the current design of existing techniques [4].

② To facilitate reproducible and fair evaluation of analysis and instrumentation techniques, *DECREE* consists of a *testbed* with built-in automated test generation techniques, to rigorously evaluate and compare techniques in the same domain (e.g., taint analysis [1]) with standard baselines, such as a standard cloud-based testing environment, comprehensive test cases and evaluation metrics (e.g., performance, energy consumption), and representative benchmark applications.

③ To bridge the gap between researchers and app developers, *DECREE* provides a cloud-based *open repository* that contains *DECREE*-compatible techniques, along with the corresponding *testbed*, allowing both researchers and app developers to easily discover what they need, and enabling unbiased comparison and replication studies of *DECREE*-compatible techniques in an automatic manner.

In this talk, we will introduce *DECREE*'s high-level design including all three major components, highlight representative use cases with examples of existing techniques to show *DECREE*'s potential to fundamentally alter the landscape in the mobile app domain, and discuss research challenges and directions to motivate future research in this area.

References

[1] F. Pauck, E. Bodden, and H. Wehrheim, "Do android taint analysis tools keep their promises?" in *Proceedings of the 26th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*, November 2018.

[2] Y. Zhao, M. S. Laser, Y. Lyu, and N. Medvidovic, "Leveraging program analysis to reduce user-perceived latency in mobile applications," in *Proceedings of the International Conference on Software Engineering (ICSE)*, May 2018.

[3] Y. Zhao, P. Wat, M. S. Laser, and N. Medvidović, "Empirically assessing opportunities for prefetching and caching in mobile apps," in *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*. ACM, 2018, pp. 554–564.

[4] Y. Zhao and N. Medvidovic, "A microservice architecture for online mobile app optimization," *arXiv preprint arXiv:1902.08879*, 2019.