

A3: Context-Aware GUI Testing for Mobile Applications

1. Introduction

In recent years, testing Graphical User Interfaces (GUIs) for mobile applications has faced increasing challenges due to the diversity and complexity of mobile app interactions. Traditional automated testing tools often lack the adaptability to handle context-specific user interactions, dynamic content, and the wide variety of screen sizes and resolutions in mobile devices. In this context, the application of Large Language Models (LLMs) has emerged as a promising approach for enhancing GUI testing by making it more context-aware and adaptable to real-world scenarios.

LLMs, trained on extensive datasets of language and user interactions, offer the potential to understand user intent and app behavior within different contexts. This ability allows LLMs to simulate realistic user interactions and predict potential edge cases that are difficult to capture with rule-based automation alone. For example, LLMs can generate test cases that respond to contextual cues, such as location, app permissions, or user settings, and adapt these tests as app states change dynamically.

The main idea behind using LLMs for context-aware GUI testing include:

- Contextual Interaction Simulation: LLMs can simulate user interactions that adapt to specific contexts, such as user profiles, permissions, and app state, offering insights into real-life usage patterns.
- Dynamic Test Case Generation: By generating test cases that are flexible and responsive to changes in app behavior and layout, LLMs can improve coverage and reveal defects that would typically require extensive manual testing.

The objective of this research is to investigate the potential of LLMs to support context-aware GUI testing in mobile applications, with the goal of enhancing test coverage, improving the accuracy of test results, and reducing the time and resources required for manual testing. By exploring the integration of LLMs into mobile testing frameworks, this study aims to offer insights into how AI-driven testing can support the development of more resilient and user-friendly mobile applications.

2. Research Questions

RQ1: How effectively can LLMs generate working context-aware test cases for mobile GUI testing?

RQ2: What coverage can be obtained by generating test cases with LLMs?

3. Background

Provide a description of the background based on the provided papers. You are free to re-organize the section into sub-sections, and to include additional references.

[1] Zhe Liu, Chunyang Chen, Junjie Wang, Mengzhuo Chen, Boyu Wu, Xing Che, Dandan Wang, "Make LLM a testing expert: Bringing human-like interaction to mobile GUI testing via functionality-aware decisions" in Proceedings of International Conference on Software Engineering 2024, ACM

[2] Junjie Wang; Yuchao Huang; Chunyang Chen; Zhe Liu; Song Wang; Qing Wang, "Software testing with large language models: Survey, landscape, and vision" in IEEE Transaction on Software Engineering, Volume 50 issue 4

[3] Liu, Zhe, et al. "Fill in the blank: Context-aware automated text input generation for mobile gui testing." 2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE). IEEE, 2023.

4. Methodology

Select an architecture and an approach, involving LLM agents, to solve this problem. Motivate each decision with concepts, theories and demonstrations discussed in the course, in the background material for this project, or both.

Select and prepare mobile applications and testing environments to ensure a representative and controlled setup for evaluating LLM-driven testing capabilities.

- **Application Selection:** Choose mobile apps that vary in complexity, interaction patterns, and user interface designs.
- **Testing Environment Setup:** Configure a testing setup. You can use Android Espresso or Appium.
- **Baseline Testing:** Conduct initial testing using traditional automated GUI testing tools to establish baseline metrics for coverage, defect detection, and test execution time.

Reference mobile applications:

<https://github.com/federicoiosue/Omni-Notes>

<https://github.com/ligi/PassAndroid>

<https://github.com/thunderbird/thunderbird-android>

Utilize LLMs to model and simulate user interactions that are contextually aware, capturing real-life usage patterns based on various dynamic conditions.

- **Context Identification:** Define contextual factors relevant to the mobile app, such as user profiles, state of the application, and so on.
- **LLM Input Generation:** Feed the LLM with natural language descriptions of these contexts and expected user interactions to generate test scenarios..

Verify the GUI page coverage reached by the test sequences (average number of interactions in a page over all the interactable object) with a manual inspection.

5. Results

5.1 Effectiveness in generating working test cases

Describe here the results related to Research Question 1.

5.2 Coverage of generated test cases

Describe here the results related to Research Question 2.

6. Conclusion

Provide in this section a reasoning about the results of your experimentation. Make sure to answer the following questions:

- What are the limitations of the approach you used?

- What further investigations or modifications to the methodology can be performed?

- Is the problem solved? Is it an advantage to perform this problem by using an LLM-based approach instead of performing it manually?