

LLM TEST GENERATION FOR JAVA LIBRARIES IN LOW CONTEXT SETTINGS

The Impact of Javadoc on LLM Test Generation Without Source Code

IPSHIT RAYCHAUDHURI (I.Raychaudhuri@student.tudelft.nl)

RESPONSIBLE LECTURER: SEBASTIAN PROKSCH

SUPERVISOR: CATHRINE PAULSEN



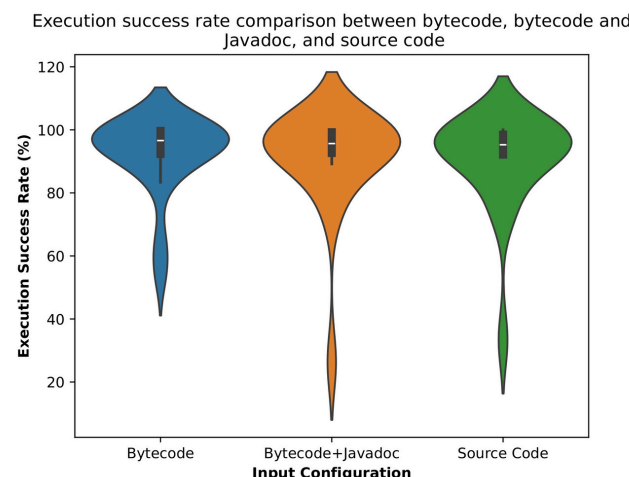
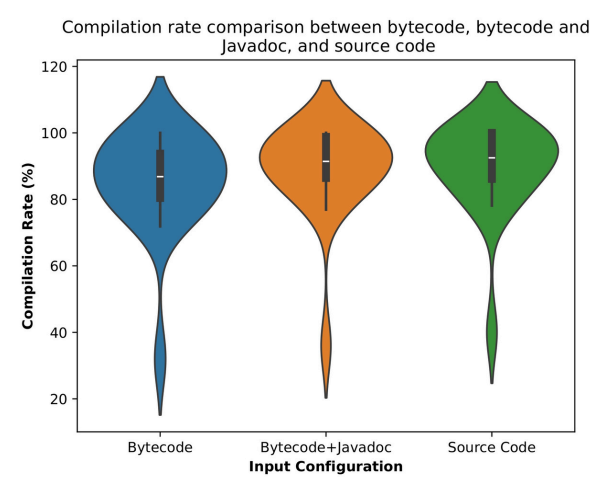
1 - Introduction

Modern software has become highly reliant on independently evolving dependencies. We need to keep these dependencies up-to-date, however there's no guarantees of compatibility afterwards. Testing can help! LLMs show promise in automatic test generation, however, current approaches assume access to full source code and additional context. So, unclear how they perform in low context settings where source code is not available.

It is also uncertain how much context is required to generate "good" tests in the absence of source code. Specifically, does the addition of Javadoc help to offset source code absence in LLM-based test generation?

4 - Results

RQ1



- Compilation rates increase with more context, Javadoc lowers the distance to source code.
- Execution success rates stay mostly similar.

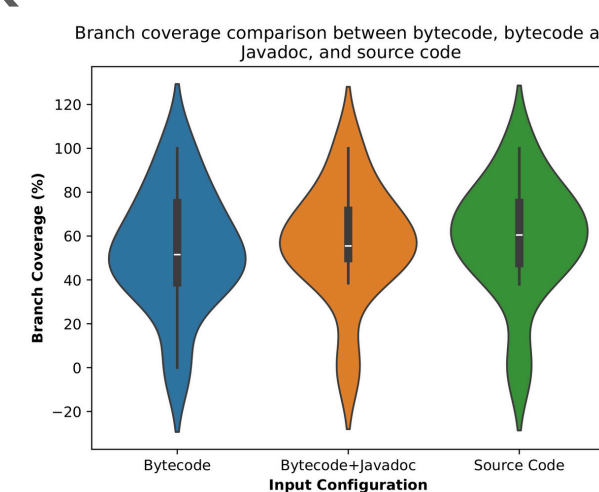
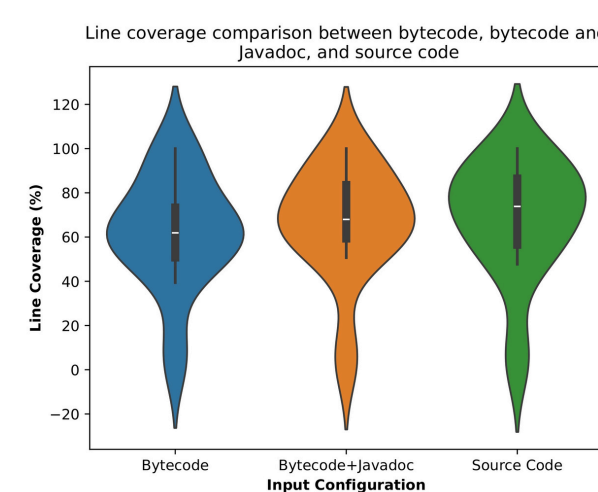
2 - Research Questions

"To what extent does Javadoc mitigate the impact of the lack of source code context in LLM-based test generation for Java libraries?"

Sub-Questions:

- **RQ1:** What impact does the level of input context have on the compilation rate and execution success rate?
- **RQ2:** What impact does the level of input context have on the degree to which tests exercise library behavior, measured in terms of line and branch coverage?
- **RQ3:** What impact does the level of input context have on the fault detection capabilities, measured in terms of mutation coverage?

RQ2



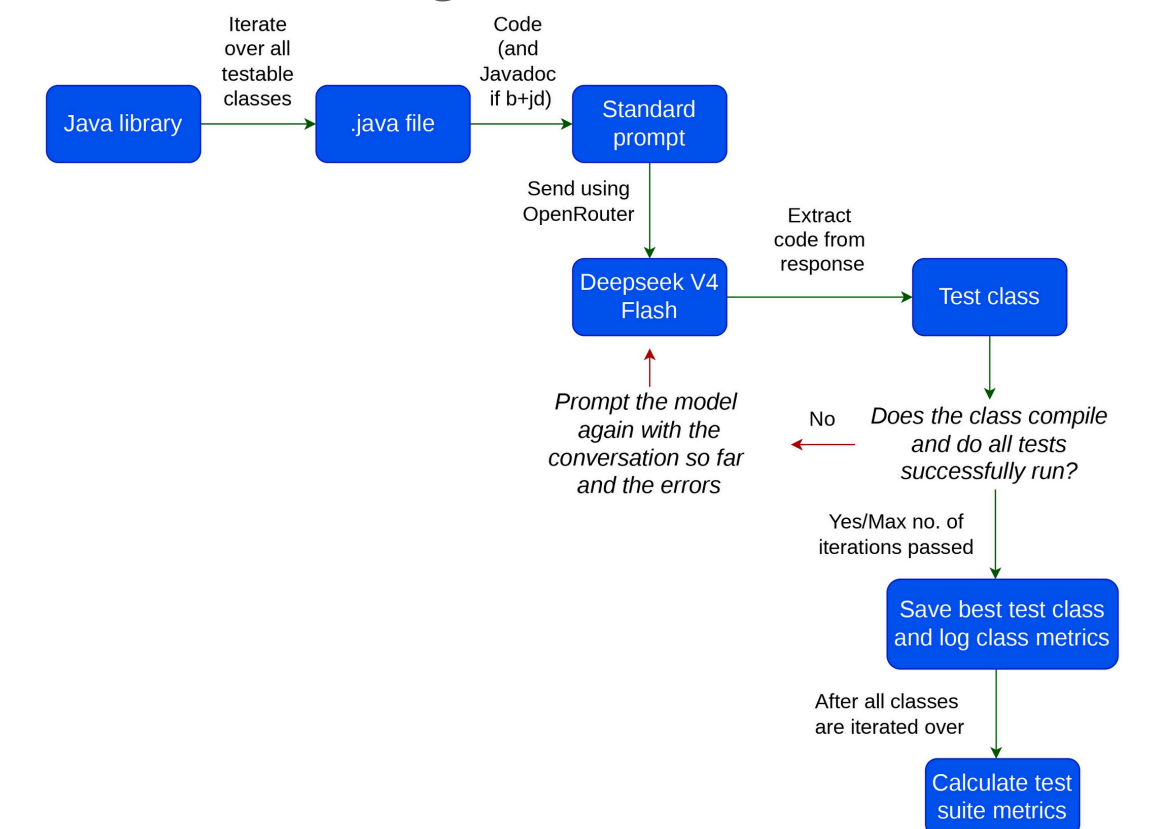
Line and branch coverage increase with more context, with Javadoc helping to reduce the gap to source code.

5 - Conclusions

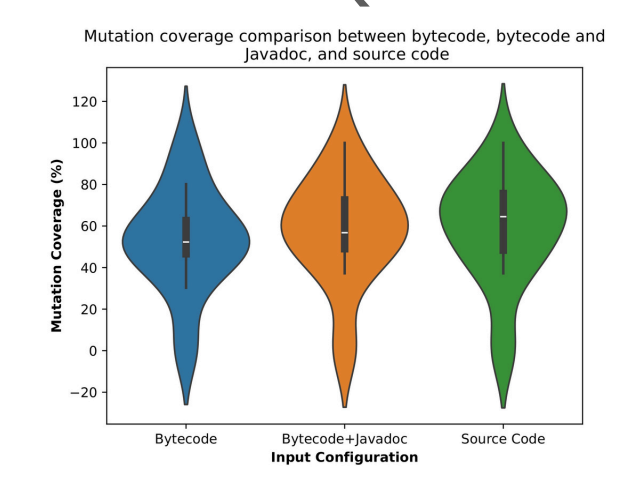
- A bytecode and Javadoc configuration is viable for practical use.
- While it does not fully match the performance of source code, it meaningfully reduces the distance to it from just bytecode.

3 - Methodology

Experiments carried out on a set of 20 libraries, where each class had a Javadoc file. The pipeline below was used to generate the test suites.



RQ3



Mutation coverage again increases with context level, with Javadoc again helping to bridge the gap.

6 - Threats to Validity & Future Work

- The size of the dataset used.
- Scope was limited to "well-maintained" libraries.
- Deepseek may have been trained on the libraries used.