Aiding the integration of automatically generated tests into pre-existing manually written test suites

Bryn Loftness
Colorado Mesa University

Research Advisor:
Dr. Venera Arnaoudova

Graduate Student Advisor:
Devjeet Roy
The Problem

Test Suite Reduction through the merging of redundant tests

Identifying optimal approach for selection of automatically generated tests for integration with similar manually written tests
Automatic

Manual
Our Approach

• Manual process is arduous and complex, even with test case scenarios and identifier renaming
• The systematic nature of our approach simulates a manual experts process of by-hand clustering:
  • Analyze jUnit manually written tests with Evosuite automatically written tests (in Java)
  • Identify key features of similarity
  • isolate and relate test cases based on the key features
  • Apply NLP-based information retrieval on key features to identify model(s) that best reduce false positive rates from the cluster list
Evaluation and Current Results

• We create an oracle based on results from manual clustering, we then evaluate model performance based on this oracle

• Clustering identifies groups of two or more test cases that are similar, our oracle simplifies these clusters into one-to-one matches

• Current approach narrows down potential one to one matches by over 50%
Identifying information that could help us categorize and find similar test cases...

Sequencing of words/keywords, similar words and keywords. And...
Digging a little deeper into sequencing by analyzing the Abstract Syntax Tree of the test case...

...Sequencing of words/keywords, similar words and keywords, similar methods being called, similar assert statements.
Our Progress So Far...

• Identified key features of test case and test case scenario to use as fruitful data inputs for similarity models (scenario words, assert statements, invoked methods)

• Series of models identifying test cases with these similar key features have reduced the potential cluster group to 50% of the possible cluster group, with plenty of room for more growth through further experimentation
Conclusions

Impacts of this Research:
Auto-identified clusters through a light-weight prototype can help programmers to select which test cases to keep, integrate, or discard.

Next Steps:
Continue working towards reducing False Positive rates by finding/tuning models to identify key features of similarity (similar wording, similar methods being tested, similar sequences of actions, etc) and continue creating the ‘series of experts’ prototype.
Acknowledgements

This material is based upon work supported by the National Science Foundation REU Program under Grant No. 1757632
Thank You!!

...Questions?