

TEST CASE PRIORITIZATION & SELECTION

Aicha Moussaid, Ricardo Chávez Tapia, Saad Waseem, S M Zahid Hasan

Problem Statement

- Regression testing is an essential activity to assure that software code changes do not adversely affect existing functionalities. CI increases the frequency of running software builds making tests execution **time-consuming and resource intensive**.
- TPS techniques deal with the costly execution time of regression testing by selecting and prioritizing test cases that are (1) sufficient to test new changes while accounting for their side effects and (2) able to detect faults as early as possible.

Approach

- Reinforcement learning agent that takes in tests individually and learns to prioritize them adequately.
- Verdict-based reward function which provides a positive reward to the agent in case of failure detection and vice versa

Research Objectives

Devise an RL approach to prioritize test cases while they are being executed to collect information for the scheduling of the pending ones, in order to reduce regression time and resource consumption.

Challenges

- Features and Datasets that fit the CI context.
- Lack of a united evaluation benchmark for dynamic TPS.
- Outperform state-of-the-art TPS model (RETECS) and domain-specific heuristic methods

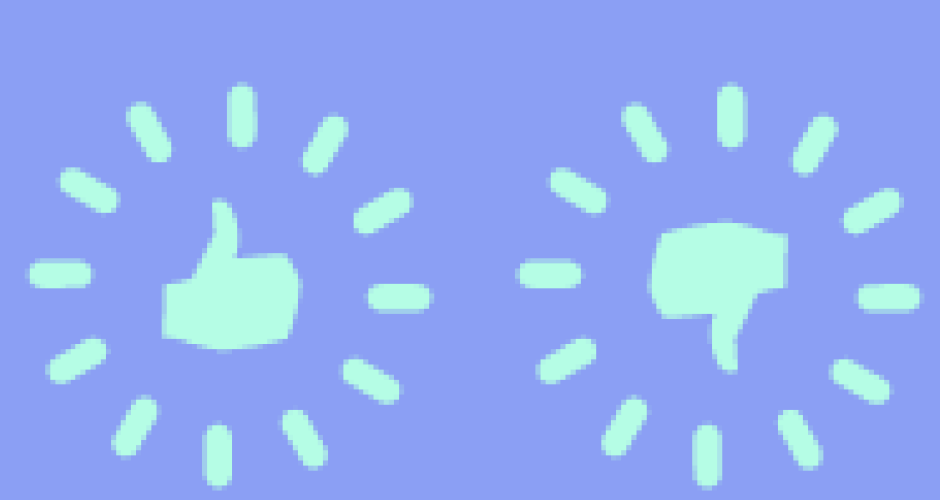
Test suite execution history



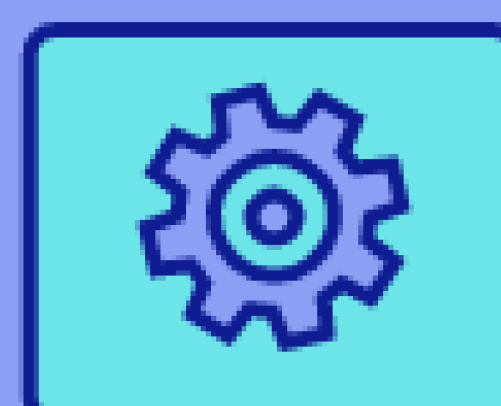
Agent (sklearn / MLP)



Reward function



CI environment



Results

The graphs displayed here compare our results against two sets of models: a random selection approach and RETECS (left), and two different heuristic approaches (right).

