Practical Program Repair via Preference-based Ensemble Strategy

Authors: Wenkang Zhong, Chuanyi Li, Kui Liu, Tongtong Xu, + 4 | Authors Info & Claims

ICSE '24: Proceedings of the 46th IEEE/ACM International Conference on Software Engineering
Article No.: 5, Pages 1 - 13 https://doi.org/10.1145/3597503.3623310

Published: 06 February 2024 Publication History

Abstract

To date, over 40 Automated Program Repair (APR) tools have been designed with varying bug-fixing strategies, which have been demonstrated to have complementary performance in terms of being effective for different bug classes. Intuitively, it should be feasible to improve the overall bug-fixing performance of APR via assembling existing tools. Unfortunately, simply invoking all available APR tools for a given bug can result in unacceptable costs on APR execution as well as on patch validation (via expensive testing). Therefore, while assembling existing tools is appealing, it requires an efficient strategy to reconcile the need to fix more bugs and the requirements for practicality. In light of this problem, we propose a **P**reference-based **E**nsemble **P**rogram **R**epair framework (**P-EPR**), which seeks to effectively rank APR tools for repairing different bugs. **P-EPR** is the first non-learning-based APR ensemble method that is novel in its exploitation of repair patterns as a major source of knowledge for ranking APR tools and

References

[1] Aldeida Aleti and Matias Martinez. 2021. E-APR: Mapping the effectiveness of automated program repair techniques. *Empir. Softw. Eng.* 26, 5 (2021), 99.

Digital Library | Google Scholar

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7--9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1409.0473

Google Scholar

Show all references

Cited By

View all

Huang K, Xu Z, Yang S, Sun H, Li X, Yan Z and Zhang Y. (2024). Evolving Paradigms in Automated Program Repair: Taxonomy, Challenges, and Opportunities. ACM Computing Surveys. 10.1145/3696450. **57**:2. (1-43). Online publication date: 10-Oct-2024.

https://dl.acm.org/doi/10.1145/3696450

Xin Q, Wu H, Tang J, Liu X, Reiss S and Xuan J. (2024). Detecting, Creating, Repairing, and Understanding Indivisible Multi-Hunk Bugs. Proceedings of the ACM on Software Engineering. 10.1145/3660828. **1**:FSE. (2747-2770). Online publication date: 12-Jul-2024.

Software and its engineering

Software creation and management

Software verification and validation

Software defect analysis

Software testing and debugging

Recommendations

Practical program repair via bytecode mutation

ISSTA 2019: Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis

Automated Program Repair (APR) is one of the most recent advances in automated debugging, and can directly fix buggy programs with minimal human intervention. Although various advanced APR techniques (including search-based or…

Read More

Posterior probability based ensemble strategy using optimizing decision directed acyclic graph for multi-class classification

Ensemble strategy is important to develop a decomposition and ensemble method for multi-class classification problems. Most existing ensemble strategies use predetermined and heuristic decision rules. In this work, we build up the decision…

Read More

Model and Program Repair via SAT Solving

Special Issue on MEMCODE 2015 and Regular Papers (Diamonds)

We consider the *subtractive model repair problem*: given a finite Kripke structure M and a CTL formula η, determine if M contains a substructure M' that satisfies η. Thus, M can be "repaired" to satisfy *eta* by deleting some transitions and states.…

Read More