Benchmarks for ML4Code

Romain Robbes*[1]

[1]LaBri – LaBRI, Universite de Bordeaux, Bordeaux INP, CNRS, UMR 5800, Talence, France – France

Résumé

We present two benchmarks and datasets that are designed to help the ML4Code community progress on goals that we think are important.

GLUE Code (Global and Local Understanding Evaluation of Code) is geared towards the development of models that use a global context beyond a code snippet. Indeed, one of our studies shows that 60% of method calls are project-specific, and 40% come from a distant context. GLUE Code is based on the JEMMA dataset of source code projects, a dataset of 50,000 projects (derived from 50K-C) that include significant post-processing to add source code representations, call graphs, and static analysis tool data. GLUE Code includes tasks that require a model to go beyond the current code snippet and include larger context (file, package, callers/callees). GLUE Code users can use JEMMA to assemble the context they need to solve the GLUE Code tasks. In this way, GLUE Code and JEMMA allow users to experiment with a variety of source code contexts.

Find more details on JEMMA at: https://arxiv.org/abs/2212.09132

RunBugRun is a large-scale, executable, and multi-lingual dataset to incentivize Automated Program Repair models to leverage runtime information in their design. RunBugRun is derived from CodeNet; it includes 450,000 (carefully curated) executable bug/fix pairs that can be validated via running tests. Generated patches can be compiled and executed. RunBugRun includes bug/fix pairs in C, C++, Python, Java, JavaScript, Go, Ruby, and PHP. The bug/fix pairs are also labeled with respect to the kind of changes they include. Initial results on two baselines show both that there is room for future improvement, and the potential of transfer learning from common to uncommon languages.

Find more details on RunBugRun at: https://github.com/giganticode/run_bug_run

Based on joint work of:
Anjan Karmakar, Julian Prenner, Miltiadis Allamanis

*Intervenant