A Artifact Appendix

A.1 Abstract

This artifact includes a binary release of our proposed disassembler, DeepDi, and a sample script to show how we can use it to obtain instruction and function boundaries. This binary release is mainly to demonstrate the performance and efficiency of DeepDi, so for now it can only be used on Windows where CUDA is available.

A.2 Artifact check-list (meta-information)

- **Run-time environment:** Windows 10 x64, CUDA 11.1, cuDNN v8.2.4, MSVC runtime 2019, and Python 3.8.
- **Hardware:** NVIDIA GPU with compute capability of 5.2 or above.
- **Metrics:** Execution time and validation accuracy.
- **Output:** The evaluation script outputs precision, recall, and execution time.
- **Experiments:** An evaluation script is provided.
- **How much disk space required (approximately)?:** 300 MB.
- **How much time is needed to prepare workflow (approximately)?:** An hour to download and install all the dependencies.
- **How much time is needed to complete experiments (approximately)?:** A minute.
- **Publicly available?:** Yes, on GitHub.

A.3 Description

A.3.1 How to access

The full artifact is available on GitHub at the following URL: https://github.com/DeepBitsTechnology/DeepDi/tree/74f0af0d4cdf33fc5de6f55d5f4ec5142de68c18.

A.3.2 Hardware dependencies

A machine with NVIDIA GPU with compute capability of 5.2 or above is required. Here is a list of NVIDIA GPUs and their compute capabilities: https://developer.nvidia.com/cuda-gpus.

A.3.3 Software dependencies

- Windows 10 x64
- Python 3.8
- CUDA 11.1
- cuDNN v8.2.4 for CUDA 11.4
- MSVC runtime 2019
  - https://aka.ms/vs/16/release/vc_redist.x64.exe
- pyelftools
  - pip install pyelftools
- NumPy

A.3.4 Data sets

This artifact contains the benchmark and real-world binaries and the corresponding ground truth for instruction- and function-level evaluation, but feel free to try any x86 or x64 binaries. Please change DATA_PATH in eval.py to your real data path.

You can download the benchmark binaries at https://drive.google.com/file/d/1UfS4YsbKWw6Xlp7NXf4tTHDN7gzDRY7p/view?usp=sharing
and the real-world binaries at https://drive.google.com/file/d/1x3N_0FAmSuT56D-KHRPvSaMz8Xl3gQFj/view?usp=sharing.

A.4 Installation

This binary release requires no installation. Detailed instructions on using and running the tool are included in the README file.

A.5 Experiment workflow

The included evaluation script gets the code section of the included sample file and feeds it into DeepDi. DeepDi then outputs where instructions and functions are.

A.6 Evaluation and expected results

The evaluation script will measure the precision and recall of instruction and function prediction, and the execution time of DeepDi.

A.7 Experiment customization

Though only one binary is included for demonstration purposes, DeepDi can be easily extended to evaluate arbitrary x86 and x64 binaries.