Symposium MATHEMATICS AND APPLICATION, Faculty of Mathematics, University of Belgrade, 2025

Improving ML-Based Static Profiling Using Method Names

Stefan Milenković

Faculty of Mathematics, University of Belgrade
Oracle, Belgrade
e-mail: stefan.milenkovic@matf.bg.ac.rs

Milan Čugurović

Faculty of Mathematics, University of Belgrade
Oracle, Belgrade
e-mail: milan.cugurovic@matf.bg.ac.rs

Milena Vujošević Janičić

Faculty of Mathematics, University of Belgrade
Oracle, Belgrade
e-mail: milena.vujosevic.janicic@matf.bg.ac.rs

Abstract. Profile-guided optimizations (PGO) can yield substantial performance improvements or reduce the binary size of generated programs. Despite these benefits, PGO is still not widely adopted because it relies on dynamic profiling, which places non-trivial demands on developers by requiring them to identify suitable workloads for profile data collection. To mitigate this cost, several static profiling techniques have been proposed [1], with recent approaches leveraging machine learning for more accurate predictions [2, 3]. These techniques typically estimate branch probabilities from feature sets that capture static branch information, such as control-flow structure, basic-block properties, and branch-instruction types.

In this work, we employ a gradient-boosted binary classifier to predict method hotness in the GraalVM Native Image compiler [4], with a focus on minimizing binary size. We further extend existing feature sets by incorporating method-name features, which aim to improve prediction accuracy by exploiting semantic information often reflected in method names. Using GloVe embeddings [5] to encode method names, we measure an 8% reduction in binary size with only a 2% runtime performance penalty compared to a baseline model without these features.

Keywords: Static Profilers; GraalVM Native Image; Binary Size Reduction; Machine Learning.

References

[1] **Y. Wu, J. R. Larus.** Static branch frequency and program profile analysis. In: *Proceedings of the 27th annual international symposium on Microarchitecture (MICRO 27)*, 1994, 1—11

[2] **M. Čugurović, M. Vujošević Janičić, V. Jovanović, Thomas Würthinger.** GraalSP: Polyglot, efficient, and robust machine learning-based static profiler. In: *Journal of Systems and Software*, 2024, 213, 112058.

[3] **A. A. Moreira, G. Ottoni, F. M. Q. Pereira.** VESPA: Static Profiling for Binary Optimization. In: *Proc. ACM Program. Lang.*, 2021, 5 (OOPSLA), 144.

[4] **C. Wimmer, C. Stancu, P. Hofer, V. Jovanović, P. Wögerer, P. B. Kessler, O. Pliss, T. Würthinger.** Initialize once, start fast: application initialization at build time. In: *Association for Computing Machinery*, 2019, 3 (OOPSLA), 184.

[5] **J. Pennington, R. Socher, C. D. Manning** GloVe: Global Vectors for Word Representation. In: *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, 1532–1543.