Leveraging HPC Resources to Improve the Experimental Design of Software Analytics

Chakkrit (Kla) Tantithamthavorn[1,3], Shane McIntosh[2], Ahmed E. Hassan[1], Kenichi Matsumoto[3]

[1] Queen's University, Canada.
[2] McGill University, Canada.
[3] Nara Institute of Science and Technology, Japan.
chakkrit@cs.queensu.ca, shane.mcintosh@mcgill.ca, ahmed@cs.queensu.ca, matumoto@is.naist.jp

Abstract. Software quality analytics is a statistical or machine learning classifier that is trained to identify defect-prone software modules. The goal of software quality analytics is to help software engineers prioritize their software testing effort on the most risky modules and understand past pitfalls that lead to defective code. While the adoption of software quality analytics enables software organizations to distill actionable insights, there are still many barriers to broad and successful adoption of such analytics systems. Indeed, even if software organizations can access such invaluable software artifacts and toolkits for data analytics, researchers and practitioners often have little knowledge to properly develop analytics systems. Thus, the correctness of the predictions and the insights that are derived from analytics systems is one of the most important challenges of data science in software engineering.

In this work, we conduct a series of empirical investigation to better understand the impact of experimental components (i.e., class mislabelling, parameter optimization of classification techniques, and model validation techniques) on the performance and interpretation of software quality analytics. To accelerate the large amount of compute-intensive experiment, we leverage the High-Performance-Computing (HPC) resources of Centre for Advanced Computing (CAC) from Queen's University, Canada. Through case studies of systems that span both proprietary and open-source domains, we demonstrate that (1) class mislabelling does not impact the precision of software quality analytics; (2) automated parameter optimization for classification techniques substantially improve the performance and stability of software quality analytics; and (3) the out-of-sample bootstrap validation technique produces a good balance between bias and variance of performance estimates. Our results lead us to conclude that the experimental components of analytics modelling impact the predictions and associated insights that are derived from software quality analytics. Empirical investigations on the impact of overlooked experimental components are needed to derive practical guidelines for analytics modelling.

Keywords: Software Engineering, Software Quality Analytics, Defect Prediction Models, High Performance Computing

Acknowledgement

This work would not have been possible without High Performance Computing (HPC) systems provided by the Compute Canada and the Centre for Advanced Computing at Queen's University. This work was supported by the Grant-in-Aid for JSPS Fellows (No. 16J03360), and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[2017a] Tantithamthavorn, C., McIntosh, S., Hassan, A.E. and Matsumoto, K.: "The Impact of Automated Parameter Optimization on Defect Prediction Models." Under review.

[2017b] Tantithamthavorn, C., McIntosh, S., Hassan, A.E. and Matsumoto, K.: "An Empirical Comparison of Model Validation Techniques for Defect Prediction Model." IEEE Transactions on Software Engineering (TSE). 1, 1–18 (2017).

[2017c] Hassan, S and Tantithamthavorn, C and Bezemer, CP and Hassan, A E. "Studying the Dialogue Between Users and Developers of Free Apps in the Google Play Store." Empirical Software Engineering. (2017).

[2016a] Tantithamthavorn, C., McIntosh, S., Hassan, A.E. and Matsumoto, K.: "Comments on "Researcher Bias: The Use of Machine Learning in Software Defect Prediction"." IEEE Transactions on Software Engineering (TSE). 11, 1092–1094 (2016).

[2016b] Tantithamthavorn, C., McIntosh, S., Hassan, A.E. and Matsumoto, K.: "Automated Parameter Optimization of Classification Techniques for Defect Prediction Models." In Proceedings of the International Conference on Software Engineering (ICSE). 321–332 (2016).

[2016c] Tantithamthavorn, C.: "Towards a Better Understanding of the Impact of Experimental Components on Defect Prediction Modelling." In Proceedings of the International Conference on Software Engineering (ICSE). 867–870 (2016).

[2016d] Jiarpakdee, J., Tantithamthavorn, C., Ihara, A., and Matsumoto, K.: "A study of redundant metrics in defect prediction datasets." In Proceedings of the International Symposium on Software Reliability Engineering Workshops (ISSREW). 51–52) (2016).

[2016e] Tantithamthavorn, C.: "ScottKnottESD: The Scott-Knott Effect Size Difference (ESD) Test." https://cran.r-project.org/web/packages/ScottKnottESD/. 2016.

[2015] Tantithamthavorn, C., McIntosh, S., Hassan, A.E., Ihara, A. and Matsumoto, K.: "The impact of mislabelling on the performance and interpretation of defect prediction models." In Proceedings of the International Conference on Software Engineering (ICSE). 812–823 (2015).