Machine Learning and Configurable Systems

Encadrants

Mathieu Acher (Contact)
Mail : mathieu.acher@irisa.fr, http://www.mathieuacher.com/

Jean-Marc Jézéquel (Contact)
Mail : jean-marc.jezequel@irisa.fr

Structure d'accueil

Ville : Rennes
Désignation de l'établissement : Laboratoire
Nom de l'établissement : Inria / IRISA
Équipe : DiverSE

Mots-clés :

- Artifical intelligence
- Software engineering

Description :

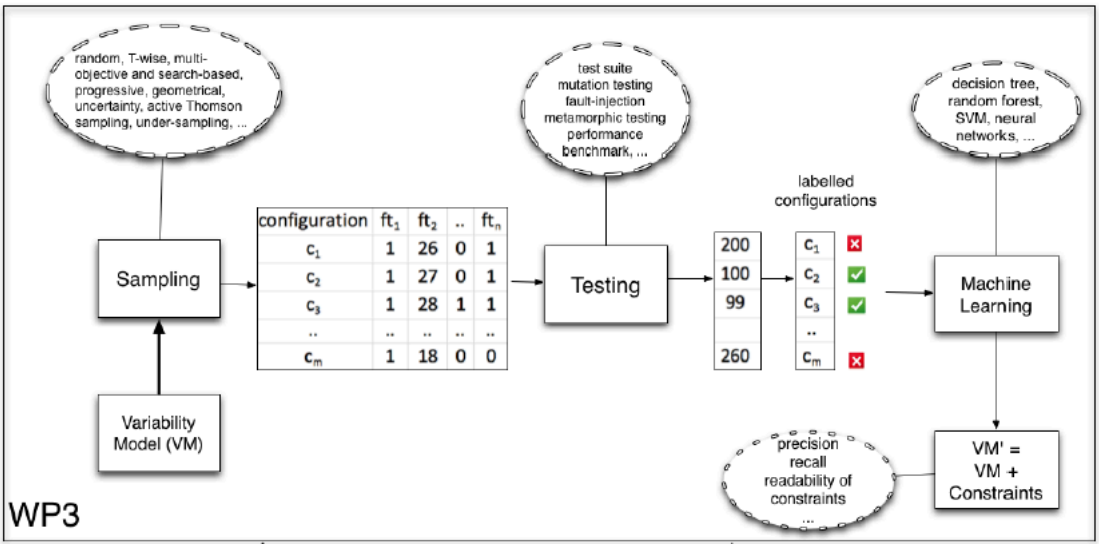
Most modern software systems are configurable and offer (numerous) configuration options to users. Web servers like Apache, operating systems like the Linux kernel, or a video encoder like x264: all can be highly configured at compile-time or at run-time for delivering the expected functionality, hopefully with an adequate performance (e.g., execution time).

Among the resulting many billions possible configurations, relating option and parameter values to desired performance is then a daunting task relying on a deep know how of the internals of the configurable system.

We propose machine learning-based process (see Figure) to narrow the space of possible configurations to a good approximation of those satisfying the wanted high level customer requirements. Based on an oracle (e.g. a runtime test) that tells us whether a given configuration meets the non-functional requirements (e.g. speed or memory footprint), we leverage machine learning to retrofit the acquired knowledge into a variability model of the system that can be used to automatically specialize the configurable system.

For instance, our process has the potential to specialize the Linux kernel such that all of its configurations boot in less than 5 seconds --- the kernel is still highly configurable for supporting the diversity of usages of Linux but

users will not spend their time in choosing a configuration that is too slow to boot. We can also specialize the Linux kernel with regards to its size, its security, its footprint – in fact any non-functional quality.



However several questions need to be addressed to scale up our process – and this is the goal of the internship (see Figure): Are some machine learning techniques (e.g., random forest, SVM) more effective ? Are some sampling techniques and heuristics more cost-effective ? Are there some « performance objectives » that are easier or harder to learn ? Large experiments as well as the development of innovative machine learning techniques are needed to answer these questions.

This project is part of VaryVary, a research project recently funded by the Agence National de la Recherche (ANR). We plan to involve the candidate into this project. After the internship, the plan is to continue the work with a PhD thesis.

Bibliographie :

[1] http://learningconstraints.github.io

[2] P. Temple, J. A. Galindo Duarte, M. Acher, and J.-M. Jézéquel, “Using

Machine Learning to Infer Constraints for Product Lines,” in Software

Product Line Conference (SPLC), Beijing, China

[3] F. Hutter, L. Xu, H. H. Hoos, and K. Leyton-Brown, “Algorithm

runtime prediction: Methods and evaluation,” Artificial Intelligence, vol.

206, pp. 79 – 111, 2014.

[4] N. Siegmund, M. RosenmüLler, C. KäStner, P. G. Giarrusso, S. Apel, and S. S. Kolesnikov, “Scalable prediction of non-functional properties in software product lines: Footprint and memory consumption,” Inf. Softw. Technol., 2013.

[5] P. Valov, J. Guo, and K. Czarnecki, “Empirical comparison of regression methods for variability-aware performance prediction,” in SPLC’15.

[6] A. Sayyad, T. Menzies, and H. Ammar, “On the value of user preferences in search-based software engineering: A case study in software product lines,” in Software Engineering (ICSE), 2013 35th International Conference on, May 2013, pp. 492–501.

[7] C. Henard, M. Papadakis, M. Harman, and Y. L. Traon, “Combining multi-objective search and constraint solving for configuring large software product lines,” in 37th IEEE/ACM International Conference on Software Engineering, ICSE 2015, Florence, Italy, May 16-24, 2015, Volume 1, 2015, pp. 517–528.

[8] Y. Xiong, A. Hubaux, S. She, and K. Czarnecki, “Generating range fixes for software configuration,” in 34th International Conference on Software Engineering, 06/2012 2012.

[9] Paul Temple, Mathieu Acher, Jean-Marc A Jézéquel, Léo A Noel-Baron, and José A Galindo. Learning-Based Performance Specialization of Configurable Systems (2017). https://hal.archives-ouvertes.fr/hal-01467299