Towards Quality Assurance of Software Product Lines with Adversarial Configurations*

*This work has already been accepted, published and presented at SPLC'19 in Paris in September 2019.

Paul TEMPLE
Faculty of Computer Science,
NaDI, PReCISE, University of Namur
Namur, Belgium
paul.temple@unamur.be

Mathieu ACHER
Univ Rennes, IRISA, Inria, CNRS
Rennes, France
mathieu.acher@irisa.fr

Gilles PERROUIN
Faculty of Computer Science,
NaDI, PReCISE, University of Namur
Namur, Belgium
gilles.perrouin@unamur.be

Battista BIGGIO
University of Cagliari
Cagliari, Italy
battista.biggio@diee.unica.it

Jean-Marc JEZEQUEL
Univ Rennes, IRISA, Inria, CNRS
Rennes, France
jean-marc.jezequel@irisa.fr

Fabio ROLI
University of Cagliari
Cagliari, Italy
fabio.roli@diee.unica.it

Software Product Line Engineering aims at delivering massively customized products within shortened development cycles [1] [2] by systematically reusing software assets realizing the functionality of one or more features, loosely defined as units of variability. Users can specify products matching their needs by selecting/deselecting the features and provide additional values for their attributes. Based on such configurations, the corresponding products can be obtained as a result of the product derivation phase. A long-standing issue for developers and product managers is to gain confidence that all possible products are functionally viable, *e.g.*, all products compile and run. This is a hard problem, since modern software product lines (SPLs) can involve thousands of features and practitioners cannot test all possible configurations and corresponding products due to combinatorial explosion. This problem is even more exacerbated when dealing with qualities aspects of products (performance, costs, *etc.*) usually requiring their derivation. A promising approach is to sample a number of configurations and predict the quantitative or qualitative properties of the remaining configurations using Machine Learning (ML) techniques [3]–[7]. Inference capabilities of predictive models (classifiers) can prevent further derivations while classifying configurations that have not been seen before. This way, configurations that do not match specific properties can be automatically discarded permanently [7], [8]. However, we need to trust the ML classifier [9], [10] to avoid costly misclassifications. Our overall goal is to study how advML techniques can be used to assess quality assurance of ML classifiers employed in SPL activities. We evaluate this new approach on an industrial video generator (called MOTIV [26]) capable of generating 10^{314} video variants. Our work makes the following contributions: (1) an adversarial attack generator, based on evasions attacks and dedicated to SPLs; (2) an assessment of its effectiveness and a comparison against a random strategy, showing that up to 100% of the attacks are valid with respect to the variability model and fooling the prediction over videos leading to a 5% accuracy loss;(3) a qualitative discussion the practical impact of advML in the quality assurance workflow of SPLs among other aspects; (4) a public repository gathering our implementation and empirical results: https://github.com/templep/SPLC_2019.

Gilles Perrouin is an FNRS Research Associate. This research was partly supported by EOS Verilearn project grant no. O05518F-RG03. This research was also funded by the ANR-17-CE25-0010-01 VaryVary project.

References

[1] P. Clements and L. M. Northrop, "Software Product Lines : Practices and Patterns." Addison-Wesley Professional, Boston, USA, 2001.

[2] K. Pohl, G. Bückle, and F. J. van der Linden, "Software Product Line Engineering: Foundations, Principles and Techniques." Springer-Verlag, 2005.

[3] J. Guo, K. Czarnecki, S. Apel, N. Siegmund, and A. Wasowski, "Variability-aware performance prediction: A statistical learning approach", ASE'13, 2013

[4] A. Sarkar, J. Guo, N. Siegmund, S. Apel, and K. Czarnecki, "Cost-Efficient Sampling for Performance Prediction of Configurable Systems", ASE'15, 2015

[5] N. Siegmund, A. Grebhahn, C. Kästner, and S. Apel, "Performance-Influence Models for Highly Configurable Systems", ESEC/FSE'15, 2015

[6] 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

[7] P. Temple, M. Acher, J.-M. Jézéquel, and O. Barais, "Learning Contextual-Variability Models", IEEE Software 34, 6, 64-70, 2017

[8] P. Temple, J. A. Galindo Duarte, M. Acher, and J.-M. Jézéquel, "Using Machine Learning to Infer Constraints for Product Lines", SPLC'16, Beijing, China, 2016

[9] M. Barreno, B. Nelson, R. Sears, A. D Joseph, and J Doug Tygar, "Can machine learning be secure?", ACM CCS'06, ACM, NewYork, NY, USA, 16-25, 2006

[10] B. Nelson, M. Barreno, F. Jack Chi, A. D Joseph, B. IP Rubinstein, U. Saini, C. A Sutton, J. Doug Tygar, and K. Xia, "Exploiting Machine Learning to Subvert Your Spam Filter", LEET, 8, 1-9, 2008