Requirements-driven Reuse Recommendation

Muhammad Abbas
muhammad.abbas@ri.se
RISE Research Institutes of Sweden
Västerås, Sweden

Mehrdad Saadatmand
mehrdad.saadatmand@ri.se
RISE Research Institutes of Sweden
Västerås, Sweden

Eduard Paul Enoiu
eduard.paul.enoiu@mdh.se
Mälardalen University
Västerås, Sweden

ABSTRACT

This tutorial explores requirements-based reuse recommendation for product line assets in the context of clone-and-own product lines.

CCS CONCEPTS

• **Software and its engineering** → **Software product lines**.

KEYWORDS

SPL adoption, Software Reuse, Similarity

ACM Reference Format:
Muhammad Abbas, Mehrdad Saadatmand, and Eduard Paul Enoiu. 2021. Requirements-driven Reuse Recommendation. In *25th ACM International Systems and Software Product Line Conference - Volume A (SPLC '21), September 6–11, 2021, Leicester, United Kingdom.* ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3461001.3472729

1 MOTIVATION

Products are often developed as different variants to address varying customer needs. Quick delivery of such complex products is only made possible with an effective engineering process, such as software product line engineering (SPLE). However, systematic SPLE has a high upfront cost and requires drastic changes to the processes within a company. Therefore, companies often try to reduce the cost of SPLE by adopting it incrementally with a less systematic process for software reuse, such as clone-and-own. In such a setting, in product derivation, engineers tend to select and reuse individual requirements from the product line without taking into account the dependencies (called Free Selection [5]). Generally, clone-and-own reuse and free selection are not recommended in SPLE, but it has the benefit of being quick and less expensive. However, companies might face several challenges in product derivation and evolution with this way of working [2].

Problem. With time, when companies develop many derived products, engineers find it hard to know if a "new" requirement is already implemented [2, 4]. At this stage, engineers only have access to the agreed-upon customer requirements.

SPLC '21, September 6–11, 2021, Leicester, United Kingdom

ACM ISBN 978-1-4503-8469-8/21/09…$15.00
https://doi.org/10.1145/3461001.3472729

2 PLAN

In this *half-day (2.5 hours)* tutorial, we first present the motivation and rationale of unsystematic reuse within the SPLE. We then present why *Free Selection* becomes an important task in such a process. Therefore, this tutorial also explores the possibility of using Pre-Trained language models to support free selection. Finally, the tutorial provides a hands-on session on language models to leverage requirements similarity to support SPL assets reuse. The tutorial is intended for both industry professionals and researchers working in the area of SPLE.

2.1 Outline for Hands-on Session

We will start with exploring if requirements similarity can be used as a proxy for retrieving relevant software that can be used to realize the new requirement [1, 3]. This part of the tutorial also explores the association between requirements and software similarity.

Requirements similarity can be computed using different approaches, such as string level metrics, and on the vector representation of the requirements. We explore the use of a diverse set of metrics to compute similarity among requirements. In particular, we use similarity computed via string-level metrics, information retrieval, and machine learning.

Reuse Recommendation. In this part of the tutorial, we will put the pieces together to build a reuse recommender system in Python. The idea is to develop a recommender system that uses existing cases (requirements and their links to source code) for case-based reuse recommendation. Finally, we will share our experiences in applying and evaluating the VARA [3] reuse recommender in the railway domain.

Required Software. Python 3, Jupyter Notebook and R Studio.

REFERENCES

[1] Muhammad Abbas, Alessio Ferrari, Anas Shatnawi, Eduard Paul Enoiu, and Mehrdad Saadatmand. 2021. Is Requirements Similarity a Good Proxy for Software Similarity? An Empirical Investigation in Industry. In *Requirements Engineering: Foundation for Software Quality*, Fabiano Dalpiaz and Paola Spoletini (Eds.). Springer International Publishing, Cham, 3–18.

[2] Muhammad Abbas, Robbert Jongeling, Claes Lindskog, Eduard Paul Enoiu, Mehrdad Saadatmand, and Daniel Sundmark. 2020. Product Line Adoption in Industry: An Experience Report from the Railway Domain. In *Proceedings of the 24th ACM Conference on Systems and Software Product Line: Volume A*. ACM.

[3] Muhammad Abbas, Mehrdad Saadatmand, Eduard Enoiu, Daniel Sundamark, and Claes Lindskog. 2020. Automated Reuse Recommendation of Product Line Assets Based on Natural Language Requirements. In *Reuse in Emerging Software Engineering Practices*, Sihem Ben Sassi, Stéphane Ducasse, and Hafedh Mili (Eds.). Springer International Publishing, Cham, 173–189.

[4] Yael Dubinsky, Julia Rubin, Thorsten Berger, Slawomir Duszynski, Martin Becker, and Krzysztof Czarnecki. 2013. An Exploratory Study of Cloning in Industrial Software Product Lines. In *2013 17th European Conference on Software Maintenance and Reengineering*. 25–34. https://doi.org/10.1109/CSMR.2013.13

[5] Mike Mannion and Hermann Kaindl. 2008. Using parameters and discriminants for product line requirements. *Systems engineering* 11, 1 (2008), 61–80.