Proceeding Paper

Towards a Semi-Automated Data-Driven Requirements Prioritization Approach for Reducing Stakeholder Participation in SPL Development †

María Isabel Limaylla * , **Nelly Condori-Fernandez** and **Miguel R. Luaces**

Database Lab. Elviña, Fac. Informática, Universidade da Coruña, CITIC, 15071 A Coruña, Spain;
n.condori.fernandez@udc.es (N.C.-F.); miguel.luaces@udc.es (M.R.L.)
* Correspondence: maria.limaylla@udc.es
† Presented at the 4th XoveTIC Conference, A Coruña, Spain, 7–8 October 2021.

Abstract: Requirements prioritization (RP), part of Requirements engineering (RE), is an essential activity of Software Product-Lines (SPL) paradigm. Similar to standard systems, the identification and prioritization of the user needs are relevant to the software quality and challenging in SPL due to common requirements, increasing dependencies, and diversity of stakeholders involved. As prioritization process might become impractical when the number of derived products grows, recently there has been an exponential growth in the use of Artificial Intelligence (AI) techniques in different areas of RE. The present research aims to propose a semi-automatic multiple-criteria prioritization process for functional and non-functional requirements (FR/NFR) of software projects developed within the SPL paradigm for reducing stakeholder participation.

Keywords: multiple-criteria prioritization; Software Product Line; Artificial Intelligence (AI) techniques

1. Introduction

Requirements prioritization (RP) is an important activity of requirements management, however, this activity can become a complex process in a family of products projects, due to common requirements, increasing dependencies, and diversity of stakeholders involved. In most prioritization method, such as Hundred Dollar, MoSCoW and Numerical Assignment Technique (NAT), the participation of stakeholders are essential to provide the prioritization criteria based on their expertise [1]. In this respect, Hujainah et al. [2] suggest the exclusion of users from tasks that can be automated, and include them only in important tasks that generate value.

In the latest years, the application of AI techniques in several stages in Software Engineering has been increasing and will continue growing [3]. We argue that it is possible to take advantage of these techniques to exploit information and discover new criteria, to decrease the stakeholder's participation.

In this paper, we focus mainly on those activities that can be automated for identifying a set of prioritization criteria and generating a list of ranked requirements. We also analyzed the available datasets and discuss their main limitations. In the next section, the proposed process is shown in detail.

2. A Semi-Automated Data-Driven Requirements Prioritization Process

The proposed process consists of two phases, Criteria Identification Phase and Requirements Prioritization Phase. A summary of the proposal is shown in Figure 1.
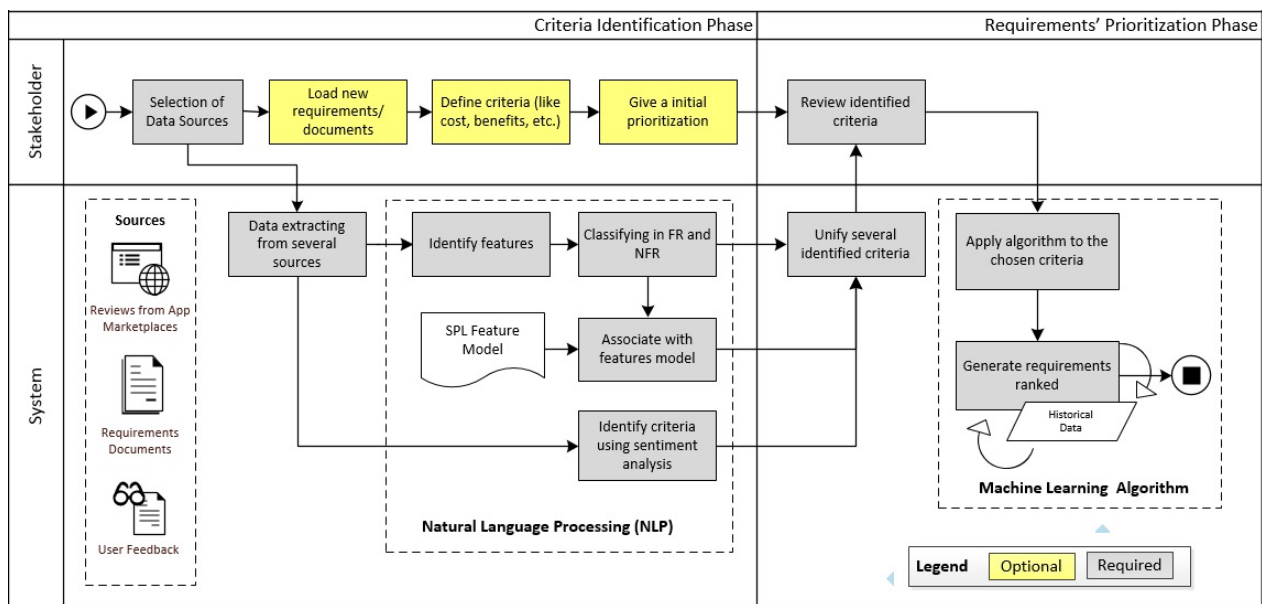
Figure 1. Data sources and AI techniques used for prioritizing requirements of SPL projects.

2.1. Criteria Identification Phase

In this first phase, multiple prioritization criteria are identified with the minimum stakeholder participation. This phase starts with the data sources selection carried out by the analyst, and, optionally, loading new requirements and criteria. Then, the data is automatically collected by extracting and analyzing data from several sources, like reviews from App Marketplaces and requirements' formal documents. After collection, Natural Language Processing (NLP) techniques can be used to identify features (features are distinctive characteristics or properties of a family of systems) and associating them with existing features in feature models. Feature models are diagrams in SPL projects that show features in a hierarchical structure and conceptual relationships among features [4]. These features can be previously prioritized and new prioritization criteria can be obtained when associating the new features with the existing ones. Moreover, thanks to the use of sentiment analysis, we aim to identify sentiment and deontic in user reviews, which can provide another type of prioritization criteria. A (supervised or non-supervised) classification algorithm is used to perform the classification in FR/NFR. This classification can be used as other criteria, due to the importance of some NFRs like security or performance, considered crucial to the quality of systems. All these criteria can be obtained automatically, without the participation of stakeholders.

2.2. Requirements Prioritization Phase

In the Second Phase, a requirements prioritization is performed based on criteria previously identified. All these criteria require to be unified and summarized in order to provide more understandable information. At this point, stakeholders can review the prioritization criteria, by confirming those that are relevant for the project. Once the criteria are selected, the prioritization is performed automatically by means of a machine learning algorithm. Algorithms such as Machine-Learned ranking, classification algorithms like Decision Tree or Random Forest, and even Deep Learning algorithms in combination with others algorithms can be used in this process. Finally, the output of this process is a list of ranked requirements. This will be saved as historical data for future use.

2.3. Datasets

Datasets are an essential component of any machine learning model. PROMISE [5] is a dataset used in most of the research for FR/NFR classification. This dataset has 625 requirement sentences, with 255 identified as functional and 370 as non-functional

requirements. The NFR is labeled with the following types: Availability, Legal, Look and feel, Maintainability, Operational, Performance, Scalability, Security, Usability, Fault tolerance, and Portability. However, it presents unbalanced data in the categories of NFR. The unbalanced data can affect the precision and recall metrics of several classification algorithms, and generate a biased model. There are several ways to address this problem. Down-sampling in the majority classes is one technique, but it could lose valuable data. Synthetic data generation (Up-sampling) is another technique, that using some algorithms to create data that follow the tend of the minority classes. Balanced ensemble learning refers to the use of multiple learning machines and combines their outputs to obtain a better prediction.

For requirements prioritization methods based on supervised algorithms, RALIC [6] is a dataset used for some research. RALIC dataset contains several data about ratings and recommendations of requirements by stakeholders. This dataset is used in traditional methods and in machine learning methods for predicting the value of a rating from stakeholders.

Both datasets are in the English language. These datasets are good references but more datasets, especially in Spanish, are needed. This implies collecting historical requirements and carrying out their labeling, get balanced and standardized data and ensure enough quantity for training, testing and validation.

3. Conclusions

In this article, we presented a data-driven requirements prioritization process that can be used in SPL projects. The proposed prioritization process aims to reduce mainly the stakeholder participation through the identification of additional criteria to avoid some risks like disagreement between stakeholders and lack of time. We rely on AI techniques, like NLP and Machine Learning algorithms, to optimize mainly the criteria identification by exploiting information from different data sources. We also review two datasets that are used for FR/NFR classification and for requirements prioritization. As a result of this review, some of their limitations (e.g., imbalanced datasets), and the necessity of new datasets were identified.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available PROMISE dataset [7] and RALIC dataset [6] were analyzed in this study.

References

1. Hudaib, A.; Masadeh, R.; Qasem, M.H.; Alzaqebah, A. Requirements Prioritization Techniques Comparison. *Mod. Appl. Sci.* **2018**, *12*, 62. [CrossRef]
2. Hujainah, F.; Bakar, R.B.A.; Abdulgabber, M.A.; Zamli, K.Z. Software Requirements Prioritisation: A Systematic Literature Review on Significance, Stakeholders, Techniques and Challenges. *IEEE Access* **2018**, *6*, 71497–71523. [CrossRef]
3. Barenkamp, M.; Rebstadt, J.; Thomas, O. Applications of AI in classical software engineering. *AI Perspect.* **2020**, *2*, 1–15. [CrossRef]
4. Lee, K.; Kang, K.C.; Lee, J. Concepts and guidelines of feature modeling for product line software engineering. In *Lecture Notes in Computer Science, Proceedings of the International Conference on Software Reuse, Austin, TX, USA, 15–19 April 2002*; Springer: Berlin/Heidelberg, Germany, 2002; pp. 62–77.
5. Sayyad Shirabad, J.; Menzies, T. *The PROMISE Repository of Software Engineering Databases*; School of Information Technology and Engineering, University of Ottawa: Ottawa, ON, Canada, 2005.
6. Lim, S.L.; Finkelstein, A. StakeRare: Using social networks and collaborative filtering for large-scale requirements elicitation. *IEEE Trans. Softw. Eng.* **2012**, *38*, 707–735. [CrossRef]
7. Cleland-Huang, J.; Mazrouee S.; Huang L.; Port, D. *nfr [Data Set]*; Zenodo: Geneva, Switzerland, 2007. [CrossRef]