Course: Requirements Engineering

Emergent Non-Functional Requirements in Machine Learning

Author: André Gomes

Instructor: **António Lucas Soares**

Date Last Edited: June 26, 2022

1 Introduction

Machine Learning (ML), a sub-field of Artificial Intelligence (AI), with its fields, such as Deep Learning (DL), it's a field of study that uses computing power as a way to turn empirical data, into usable models that can make useful predictions. [2] In recent years, there's been an adoption of ML systems to solve various problems, such as image classification and sentiment analysis, and during the development of these systems, there's a need for a Requirements Engineering (RE) process to elicit, model and document the requirements.

This essay will focus on the Non-Functional Requirements (NFR) of these systems, and put aside the Functional Requirements (FR), like the accuracy, functionality and specific goals of each system, to elaborate on existing NFR used both for source code systems and ML systems, highlight the NFR that have a different meaning in the ML context and discover new requirements that only make sense when applied to ML.

2 Development

ML programming differs substantially from source code programming, in the way that, for an average project, the programmer develops code to perform a task, using the available resources, to accomplish a goal. In the case of ML, the programmer uses large amounts of data to feed a computing model and make predictions, with each wrong prediction, the model can update itself based on hyperparameters and the data provided.

More succinctly, ML programming is based on three criteria: the model, the data and the tuning of parameters. And in the case of RE, requirements can be elicited for demands over the model, over the data or the system [3]. Due to the learning and black-box nature of ML models, you can train two models on the same data and obtain different final models, which means that settings requirements for the models, or even the system, can sometimes be more important than setting requirements for the data.

The first NFR specific to ML to be analysed is **Transparency**. Models receive data and output predictions, and a programmer can choose which data it feeds to the model and can evaluate the predictions, but is never sure on how the model adapted itself on how to learn to provide more accurate predictions. For example, for a system that receives images of skin moles and classifies them as benign or malignant, it is hard to know based on what criteria does the system bases itself upon to provide accurate results, which could be the colour, size, texture, shape or even a combination of these. Because of this, a new field was created to develop methods to explain and interpret machine learning models [4] called Explainable Artificial Intelligence (XAI). Similar to this NFR is **Traceability**, to know the steps of reasoning of the model and **Justifiability**.

Fairness is also a NFR specific to ML. The main goal of a model is to discriminate the data, find patterns and categorize it to make classifications and predictions, but a model does not possess the morality that humans have, and as such, may focus on some sensible features that should not be used for classification, such as race and gender. This criterion depends on the type of system where it is applied. If there's a model that decides if a defendant is guilty or not, it cannot base its verdict on the person's race or gender, but these qualities are important in the majority of the medical scenarios [1]. Similar to this NFR is **Bias** and **Ethics**.

To verify that a model is providing correct predictions, the results it outputs are validated on a subset of data to check the hit rate of correct predictions. This is mostly done during the initial development of a model. Once the model is deployed somewhere and is actively used, it is constantly taking in new training data to learn from it, and at this point, it needs something to evaluate its **Testability**. It is hard to know if a model, once it is faced with real-world updated data, will provide the same kind of performance and expected predictions, or will learn some new features that didn't exist in previous data. To keep track of the state of a model, it needs to be able to be tested in an orderly fashion, as to evaluate if it is behaving as expected in different conditions. Similar to this NFR is **Retrainability**, given the model's ability to adapt to new data.

3 Conclusion

These are just some of the NFR that have emerged in the latest years due to the higher production of more systems based on ML techniques. For now, there isn't a lot of work on how these NFR are dealt with in an industrial ML context and how to identify, scope and measure them [3], but because of the increased usage and adoption of said systems, these can become important matters in the RE area.

For future work, it could be useful if a tool is created to profile and evaluate these aspects of ML systems or models, with the usage of quantitative methods when possible and the creation of requirement models to evaluate the NFR.

References

[1] Markus Borg. *Requirements Engineering for Machine Learning: Perspectives from Data Scientists*. en-US. Sept. 2019. URL: https://mrksbrg.com/aire19-vogelsang/ (visited on 06/26/2022).

[2] T.W. Edgar and D.O. Manz. *Research Methods for Cyber Security*. Journal Abbreviation: Research Methods for Cyber Security Pages: 404 Publication Title: Research Methods for Cyber Security. Apr. 2017.

[3] K. M. Habibullah and Jennifer Horkoff. "Non-functional Requirements for Machine Learning: Understanding Current Use and Challenges in Industry". In: *2021 IEEE 29th International Requirements Engineering Conference (RE)* (2021). DOI: 10.1109/RE51729.2021.00009.

[4] Pantelis Linardatos, Vasilis Papastefanopoulos, and S. Kotsiantis. "Explainable AI: A Review of Machine Learning Interpretability Methods". In: *Entropy* (2021). DOI: 10.3390/e23010018.