Journal of Software Engineering Research and Development, 2025, 13:13, doi: 10.5753/jserd.2025.5689

Industrial Practices of Requirements Engineering for ML-Enabled Systems in Brazil: An Extended Analysis

Antonio Pedro Santos Alves [**Pontifical Catholic University of Rio de Janeiro (PUC-Rio)** | ***apsalves@inf.puc-rio.br***]
Marcos Kalinowski [**Pontifical Catholic University of Rio de Janeiro (PUC-Rio)** | ***kalinowski@inf.puc-rio.br***]
Daniel Mendez [**Blekinge Institute of Technology (BTH)** | ***daniel.mendez@bth.se***]
Hugo Villamizar [**Pontifical Catholic University of Rio de Janeiro (PUC-Rio)** | ***hvillamizar@inf.puc-rio.br***]
Tatiana Escovedo [**Pontifical Catholic University of Rio de Janeiro (PUC-Rio)** | ***tatiana@inf.puc-rio.br***]
Hélio Lopes [**Pontifical Catholic University of Rio de Janeiro (PUC-Rio)** | ***lopes@inf.puc-rio.br***]

Abstract [Context] In Brazil, 41% of companies use machine learning (ML) to some extent. However, several challenges have been reported when engineering ML-enabled systems, including unrealistic customer expectations and vagueness in ML problem specifications. Literature suggests that Requirements Engineering (RE) practices and tools may help to alleviate these issues, yet there is insufficient understanding of RE's practical application and its perception among practitioners. **[Goal]** This study aims to investigate the application of RE in developing ML-enabled systems in Brazil, creating an overview of current practices, perceptions, and problems in the Brazilian industry. **[Method]** To this end, we extracted and analyzed data from an international survey focused on ML-enabled systems, concentrating specifically on responses from practitioners based in Brazil. We analyzed the cluster of RE-related answers gathered from 72 practitioners involved in data-driven projects. We conducted quantitative statistical analyses on contemporary practices using bootstrapping with confidence intervals and qualitative studies on the reported problems involving open and axial coding procedures. **[Results]** Our findings highlight distinct RE implementation aspects in Brazil's ML projects. For instance, (i) RE-related tasks are predominantly conducted by data scientists; (ii) the most common techniques for eliciting requirements are interviews and workshop meetings; (iii) there is a prevalence of interactive notebooks in requirements documentation; (iv) practitioners report problems that include a poor understanding of the problem to solve and the business domain, low customer engagement, and difficulties managing stakeholders expectations. Our analysis suggests that development methodology plays a role in these challenges. Agile methods appear to facilitate the management of customer expectations compared to traditional approaches; however, they also appear to introduce greater difficulties in problem understanding and customer involvement. **[Conclusion]** These results provide an understanding of RE-related practices and challenges in the Brazilian ML industry, helping to guide research and initiatives toward improving the maturity of RE for ML-enabled system projects.

Keywords: *Requirements Engineering, Machine Learning, Survey, Brazil*

1 Introduction

Machine Learning (ML) has increasingly gained prominence in the global industry. Systems where ML components are integral parts of larger systems are known as ML-enabled systems. Their behavior is based on explicitly defined rules and data used by the ML component to make predictions (Sharma and Garg, 2021).

Transitioning from traditional software to ML-enabled systems poses various challenges from the viewpoint of Software Engineering (SE) (Martínez-Fernández et al., 2022). Some examples include covering additional quality properties such as fairness and explainability, dealing with a high degree of iterative experimentation, and mismatched assumptions in customers and multidisciplinary teams (Lewis et al., 2021a; Nahar et al., 2022). Such challenges typically demand extra effort to successfully develop ML-enabled systems and may contribute to the statistic that 87% of ML projects never reach production (Gartner, 2020).

Due to the communication and collaboration-intensive nature, as well as inherent interaction with most other development processes, the literature suggests that Requirements Engineering (RE) can help address several challenges when engineering ML-enabled systems (Ahmad et al., 2021; Villamizar et al., 2021; Vogelsang and Borg, 2019). However, establishing effective RE practices in ML projects may be difficult mainly due to (i) the lack of practitioners involved in formal RE activities (Alves et al., 2023), and (ii) the absence of tailored techniques and tools for data-driven projects since research on this intersection focuses mainly on using ML techniques to support RE activities rather than exploring how RE can improve the development of ML-enabled systems (Dalpiaz and Niu, 2020). Therefore, it is not surprising that recent studies emphasize that practitioners find RE the most difficult activity of ML projects (Alves et al., 2023; Ishikawa and Yoshioka, 2019; Kuwajima et al., 2020; Nahar et al., 2023).

To strengthen empirical evidence on current Brazilian industrial RE practices, perceptions, and challenges when developing ML-enabled systems, we extracted and analyzed data from an international survey focused on current practices and challenges for ML-enabled systems (Kalinowski et al., 2025). We concentrated specifically on the RE-related survey questions and on responses provided by 72 practitioners involved in ML projects that were based in Brazil. We conducted quantitative and qualitative analyses to objectively provide information on (i) what role is typically in charge of requirements; (ii) how requirements are typically elicited and documented; (iii) which non-functional requirements typically play a major role; (iv) which RE activities are perceived as most difficult; and (v) what RE-related challenges ML practitioners face. We described these results in our previous study presented at SBES 2024 (Alves et al., 2024).

This paper extends and refines our previous study (Alves et al., 2024), providing much more detail on the survey results. We provide a deeper understanding of the projects by characterizing them regarding company size, management frameworks, agility, application domains, programming languages, ML purpose, and ML algorithms. We also provide additional information, such as the ML lifecycle effort distribution. Furthermore, we refine our investigation of RE-related challenges by unfolding a new research question to examine how project agility influences these challenges. Finally, we provide a more comprehensive discussion of the implications of our findings, strengthening the connections between empirical insights and opportunities for improving RE practices for ML-enabled systems.

The remainder of this paper is organized as follows. Section II provides the background and related work. Section III describes the research method. Section IV presents the results. Sections V and VI discuss the results and threats to validity. Finally, Section VII presents our concluding remarks.

2 Background and Related Work

Machine Learning (ML) is a subfield of artificial intelligence that involves the study of algorithms and statistical models that allow software systems to learn and make predictions based on data (Jordan and Mitchell, 2015). By recognizing patterns in the data on which they are trained, ML algorithms are developed to improve automatically over time on unseen data (Mitchell, 1997). Consequently, the development of ML-enabled systems differs significantly from conventional software systems due to several key factors.

There is a high level of experimentation and uncertain outcomes when developing ML-enabled systems (Aho et al., 2020), and a multidisciplinary team is essential, comprising domain experts, software developers, data science, and engineering professionals (Nahar et al., 2022). Data scientists, who typically take the rein when developing ML projects (Kim et al., 2017), experiment with various data, algorithms, and models to determine the most effective approach for achieving their objectives, which means that setting up goals and requirements at the beginning of the process would demand an estimate of different metrics (*e.g.*, accuracy) in advance (Ishikawa and Yoshioka, 2019).

Uncertainty and experimentation are expected for this scenario, as ML projects often begin as small Proof-of-Concept (PoC) initiatives, and 87% of them never reach production (Gartner, 2020). The complexity of transitioning from laboratory-level models to production architectures brings several challenges (Lewis et al., 2021b; Zimelewicz et al., 2024). Although ML-enabled systems are hugely popular and in demand, multiple ML projects that have overcome the first barrier of reaching production have failed in recent years, leading to severe repercussions for the organizations involved and the society at large (Beede et al., 2020; Fry, 2018). The reason for this is often the same: systems that incorporate ML components tend to put stakeholder needs in the background and oversimplify important scenarios and trade-offs. This leads to a problem that the RE discipline can tackle.

RE constitutes approaches to understanding the problem space and specifies requirements that all stakeholders agree upon (Damian, 2007). As such, it concentrates on understanding the actual problem, what is needed towards a system result, and how to resolve potential conflicts, and it is thus characterized by the involvement of interdisciplinary stakeholders and often results in uncertainty (Wagner et al., 2019). The large degree of uncertainty in developing ML-enabled systems introduces new challenges and heavily affects RE (Challa et al., 2020; Martínez-Fernández et al., 2022). Agile methods aim at better coping with uncertainties and changes throughout the project, and it is known that RE is being treated differently in agile contexts (Wagner et al., 2018), raising additional interest in understanding how agility affects the RE-related challenges of developing ML-enabled systems.

To overcome such difficulties, some studies have proposed new methods or adapted existing ones to handle requirements on such systems (Ishikawa and Matsuno, 2020; Villamizar et al., 2024). However, gathering empirical evidence from the industry is essential to identify real-world challenges, perceptions, and current practices accurately. For instance, several studies have surveyed practitioners and found that unpredictability makes it difficult to define any criteria or requirements regarding the output of ML components (Alves et al., 2023; Correia et al., 2021; Vogelsang and Borg, 2019). This introduces a challenge in collaboration with stakeholders, who may perceive what ML is capable of wrongly (Giray, 2021).

We advocate that insights from practitioners can guide the development of new RE techniques for ML, thereby increasing the likelihood of designing and developing ML-enabled systems that meet customer needs and potentially avoid costly problems later on. To complement the already discussed research, we present additional empirical evidence from Brazil on the current practices, perceptions, and challenges regarding RE for ML-enabled systems obtained from our previous study, an international survey (Alves et al., 2023). We understand that bridging the gap between theory and practice is essential for RE maturity in such systems.

3 Research Method

In this section, we present the methodology adopted in our study. We begin by outlining the overall goal and the research questions that guided our investigation. We then describe the survey design, followed by the data collection and analysis procedures.

3.1 Goal and Research Questions

This paper aims to characterize the current practices, perceptions, and challenges regarding RE for ML-enabled system projects in the Brazilian industry. From this goal, we established the following research questions:

- **RQ1. What are the contemporary practices adopted in Brazil regarding RE for ML-enabled systems?** This question aims to reveal how practitioners are currently approaching RE for ML in Brazilian companies by identifying trends, main methods, and the extent to which the degree of alignment with established industry practices. We refined *RQ1* into the following questions:
  - RQ1.1: Who is addressing the requirements of Brazilian ML-enabled system projects?
  - RQ1.2: How are requirements typically elicited in Brazilian ML-enabled system projects?
  - RQ1.3: How are requirements typically documented in Brazilian ML-enabled system projects?
  - RQ1.4: Which NFRs do typically play a major role in Brazilian ML-enabled system projects?
  - RQ1.5: Which activities are considered to be most difficult when defining requirements for Brazilian ML-enabled system projects?
- **RQ2. What are the main RE-related challenges faced by practitioners working on ML-enabled system projects in Brazil?** Identifying these challenges in Brazil reflects the current maturity of these systems in the country. At the same time, it also informs the development of strategies to mitigate difficulties, helping to steer future research on the topic in a problem-driven manner. We divided this research question into two other questions. One that provides a wider view, and another that considers the development agility, as follows:
  - RQ2.1: What are the overall main RE-related challenges?
  - RQ2.2: How do the RE-related challenges manifest in agile and traditional project contexts?

  For this research question, we applied open and axial coding procedures to allow the problems to emerge from the open-text responses provided by the practitioners.

3.2 Survey Design

We extracted and analyzed data from our previous study, which presented an international survey (Alves et al., 2023; Kalinowski et al., 2025) that was conducted based on best practices of survey research (Wagner et al., 2020), carefully conducting the steps below:

- **Step 1. Initial Survey Design**. We conducted a literature review on RE for ML (Villamizar et al., 2021) and combined our findings with previous results on RE problems (Fernández et al., 2017) and the RE status quo (Wagner et al., 2019) to provide the theoretical foundations for questions and answer options. Therefrom, the initial survey was drafted by software engineering and machine learning researchers of PUC-Rio (Brazil) with experience in R&D projects involving ML-enabled systems.
- **Step 2. Survey Design Review**. The survey was reviewed and adjusted based on online discussions and annotated feedback from software engineering and machine learning researchers of BTH (Sweden). Thereafter, the survey was also reviewed by the other co-authors.
- **Step 3. Pilot Face Validity Evaluation**. This evaluation involves a lightweight review by randomly chosen respondents. It was conducted with 18 Ph.D. students taking a Survey Research Methods course. They were asked to provide feedback on the clarity of the questions and to record their response time. This phase resulted in minor adjustments related to usability aspects and unclear wording. The answers were discarded before launching the survey.
- **Step 4. Pilot Content Validity Evaluation**. This evaluation involves subject experts from the target population. Therefore, we selected five experienced data scientists developing ML-enabled systems, asked them to answer the survey, and gathered their feedback. The participants had no difficulties answering the survey, which took an average of 20 minutes. After this step, the survey was considered ready to be launched.

The final survey started with a consent form describing the purpose of the study and stating that it was conducted anonymously. The remainder was divided into 15 demographic questions (D1 to D15) and three specific parts with 17 substantive questions (Q1 to Q17): seven on the ML life cycle and problems, five on requirements, and five on deployment and monitoring. This paper focuses on the demographics, the Problem Understanding and Requirements stage of the ML life cycle, and specific questions regarding requirements. Excerpts of the substantive questions related to this paper are shown in Table 1. The survey was implemented using the Unipark Enterprise Feedback Suite [1].

3.3 Data Collection

Our target population concerns professionals involved in building ML-enabled systems, including different activities, such as management, design, and development. Therefore, it includes practitioners in positions such as project leaders, requirements engineers, data scientists, and developers. We used convenience sampling, sending the survey link to

[1] https://www.unipark.com/en/survey-software/

Table 1. Research questions and survey questions

RQ	Survey No.	Description	Type
-	...	...	...
RQ1.1	Q8	Who is actively addressing the requirements of ML-enabled system projects in your organization?	Closed (MC)
RQ1.2	Q9	How were requirements typically elicited in the ML-enabled system projects you participated in?	Closed (MC)
RQ1.3	Q10	How were requirements typically documented in the ML-enabled system projects you participated in?	Closed (MC)
RQ1.4	Q11	Which Non-Functional Requirements (NFRs) typically play a major role in terms of criticality in the ML-enabled system projects you participated in?	Closed (MC)
RQ1.5	Q12	Based on your experience, what activities do you consider most difficult when defining requirements for ML-enabled systems?	Closed (MC)
-	...	...	...
RQ2.1	Q4	According to your personal experience, please outline the main problems or difficulties (up to three) faced during the Problem Understanding and Requirements ML life cycle stage.	Open
RQ2.2	D11	Considering the ML-enabled system projects in which you participated, how agile do you rate your development?	Closed (SC)
-	...	...	...
-	-	MC = Multiple Choice SC = Single Choice	-

professionals active in our partner companies, and also distributed it openly on social media.

In this paper, we excluded participants who indicated in the survey that they had no experience with ML-enabled system projects and those working in countries other than Brazil. Data collection was open from January 2022 to April 2022. We received responses from 276 professionals; 188 completed all four sections of the survey, and of these, **72** were working in Brazil, constituting our total sample. The average time to complete the survey was 20 minutes. We conservatively considered only the 72 fully completed survey responses from professionals working in Brazil.

3.4 Data Analysis Procedures

For data analysis purposes, given that all questions were optional, the number of responses varies across the survey questions. Therefore, we explicitly indicate the number of responses when analyzing each question.

Research questions *RQ1.1 - RQ1.5* concern closed questions, so we decided to use inferential statistics to analyze them. Our population has an unknown theoretical distribution (*i.e.*, the distribution of ML-enabled system professionals is unknown). In such cases, resampling methods - like bootstrapping - have been reported to be more reliable and accurate than inference statistics from samples (Lunneborg, 2001; Wagner et al., 2020). Hence, we use bootstrapping to calculate confidence intervals for our results, similar as done in Wagner et al. (2019). In short, bootstrapping involves repeatedly taking samples with replacement and then calculating the statistics based on these samples. For each question, we take the sample of n responses for that question and bootstrap S resample (with replacements) of the same size n. We assume n as the total valid answers of each question (Efron and Tibshirani, 1993), and we set 1000 for S, which is a value that is reported to allow meaningful statistics (Lei and Smith, 2003). Figure 1 summarizes the adopted bootstrapping method.

For research question *RQ2*, which seeks to identify the main challenges faced by practitioners involved in engineering ML-enabled systems related to problem understanding and requirements, the corresponding survey question is designed to be open text. We conducted a qualitative analysis using open and axial coding procedures from grounded theory (Stol et al., 2016) to allow the challenges to emerge from the open-text responses reflecting the experience of the practitioners. The primary author performed the qualitative coding procedures and subsequently reviewed them with the secondary author. Additionally, three researchers from academic and industry partners reviewed the resulting codes independently.

The questionnaire, the collected data, and the quantitative and qualitative data analysis artifacts, including Python scripts for the bootstrapping statistics, charts, and peer-reviewed qualitative coding spreadsheets, are available in our open science repository (Alves et al., 2025).

4 Results

This section describes the context and results of our study. We begin by characterizing the study population and the types of projects represented. Next, we explore how participants face the requirements-related ML life cycle stage when compared to others. Finally, we go into detail to answer our research questions regarding current RE practices adopted in ML-enabled system development and challenges faced in this context.

4.1 Study Population

We focus specifically on the data obtained from Brazil as part of our previous study, which provided a larger international survey on ML-enabled systems engineering (Kalinowski et al., 2025). The study population consisted of 72 practitioners involved in data-driven projects across various industries in Brazil. These respondents held various roles, backgrounds, and professional experiences. This diverse group provides a comprehensive view of the current practices, perceptions, and challenges related to RE for ML within the Brazilian context.

Figure 2 provides insights into the characteristics of the Brazilian participants involved in the survey. Regarding

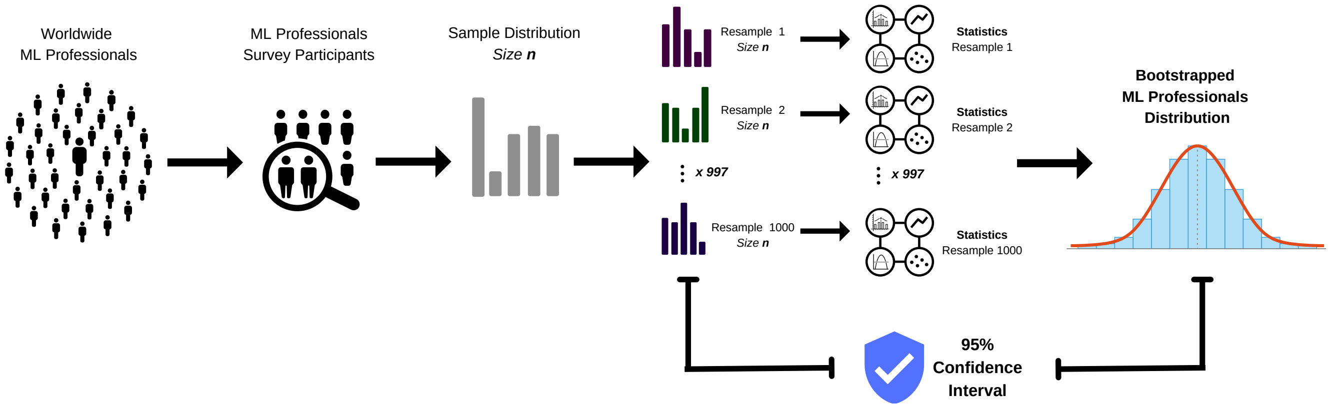


Figure 1. Bootstrapping technique

company size, the majority of participants (58.3%) are employed by companies with over 2000 employees and only 11.2% of them are employed by small companies as presented Figure 2 (a). In Figure 2 (b), we present participants' main roles. Data Scientists, Business Analysts, and Project Leads/Project Managers are the most common roles represented. Notably, the least assessed positions were Tester and Requirements Engineer, with one professional each. In the 'Others' field, some isolated positions were mentioned, including ML engineer, Data Analyst, and C-level positions, but they were less representative.

Regarding ML-enabled system experience, in Figure 2 (c), most participants reported having 1 to 2 years of working experience. Closely, another significant portion of respondents indicated a higher experience range of 3 to 6 years. This proportion emphasizes a balanced population of beginner and experienced professionals. It is noteworthy that regarding participants' educational background, 87.5% mentioned having a bachelor's degree in computer science, information systems, statistics, or electrical engineering. Moreover, 45.83% held master's degrees in computer science, data science, or electrical engineering. Lastly, 19.44% completed Ph.D. programs in computer science, physics, or computer engineering.

4.2 Project Characterization

When looking at our population, an important view is the characteristics of the ML-enabled system projects. The management framework that guides development, development agility, and the application domain are some characterization examples that help us to understand the context of ML-enabled systems development.

4.2.1 Project Management Framework

As shown in Figure 3, the most used project management frameworks rely on agile management, such as Scrum (**38.94%**) and Kanban (**28.32%**). It is also worth mentioning that 15.04% of the participants reported that no project management framework was used.

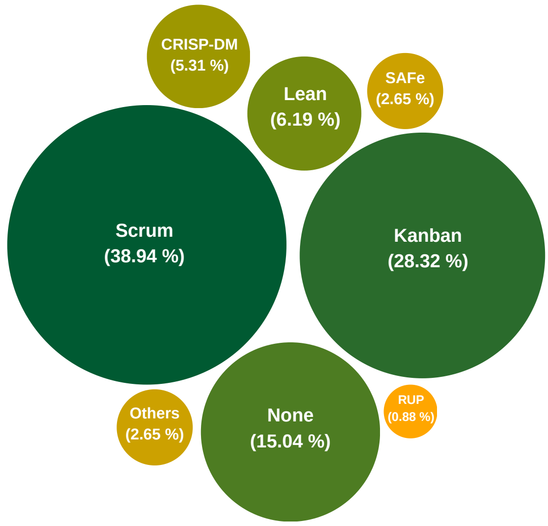
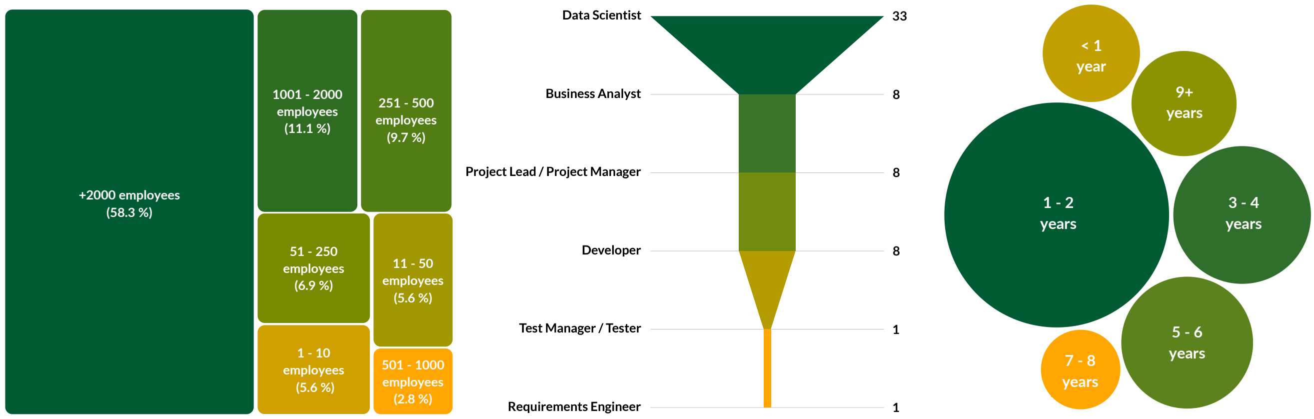


Figure 3. Project management framework

4.2.2 Agility

Regarding development agility (Figure 4), most participants consider their projects agile, with 12.5% of them considering 'Totally agile', and 27.8% 'Mostly agile'. A significant proportion feel that their projects are balanced between agile and traditional styles (34.7%). 13.8% of participants informed working under a traditional methodology (6.9% in 'Totally traditional' and 6.9% in 'Mostly traditional' methodologies). Only 11.1% answered "I don't know".



(a) Participants' Company Size (N = 72) (b) Participants' Main Role (N = 72) (c) Participants' ML Experience (N = 72)

Figure 2. Practitioners' demographics: company size, roles, and ML experience

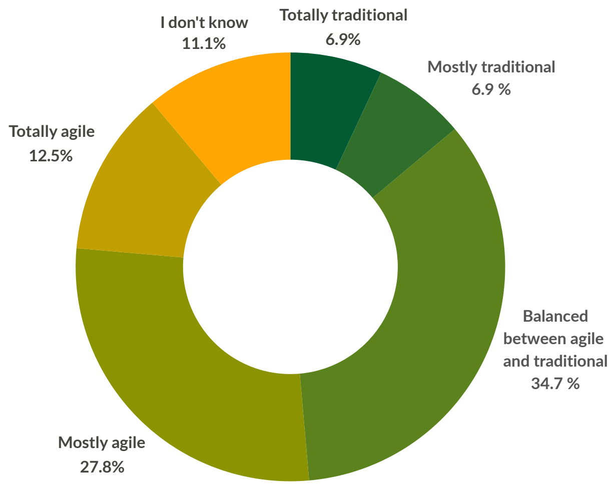


Figure 4. Development agility

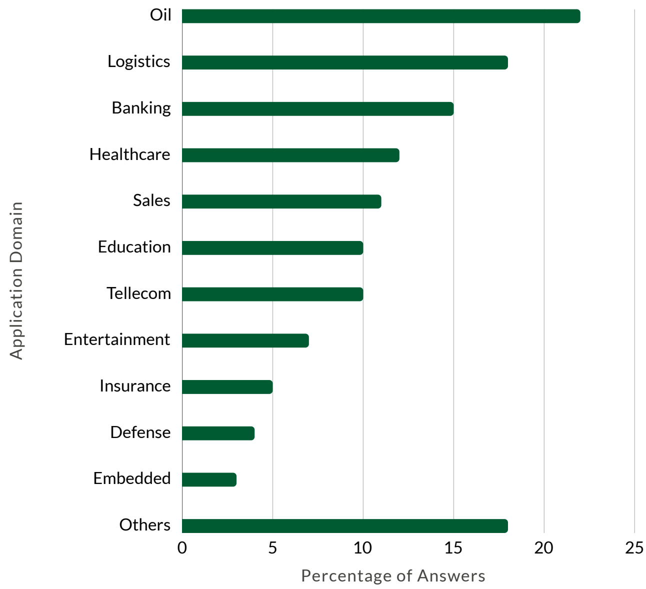


Figure 5. Project application domain

4.2.3 Application Domain

ML-enabled systems are being developed in Brazil for several different application domains (Figure 5). Within the responses, the predominant domain is Oil, with 16.30% of the projects within this domain, followed by Logistics (13.33%), Banking (11.11%), Healthcare (8.89%), Sales (8.15%), Education (7.41%), Telecom (7.41%), Entertainment (5.19%), Insurance (3.70%), Defense (2.96%), and Embedded systems (2.22%). Furthermore, several additional domains were informed on the 'Others' field, such as Human Resources, Meteorology, Law, Compliance, and Agriculture, summing up 18.33% of the answers.

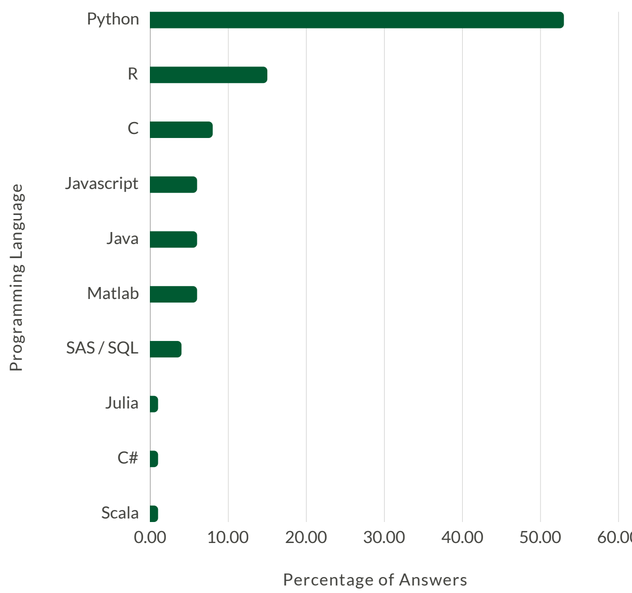


Figure 6. Programming language

4.2.4 Programming Language

The most used programming language in Brazilian ML-enabled systems projects is Python (52.8%). Another significant technology is R and C, which are respectively used in 15.2% and 8% of projects. Javascript (6.4%), Java (5.6%), Matlab (5.6%), and SQL (4%) have been less frequently used in the context of ML-enabled system projects. Furthermore, a very small number of projects were reported using Julia, C#, and Scala, each with 0.8% adoption.

4.2.5 Main Purpose

The main purposes of ML-enabled system projects in Brazil (Figure 7) concern Classification and Prediction, with 32.5% and 31.8% of the answers, respectively. Participants could specify the usage in an open-text format. Examples cited for classification projects included classifying churned users, hand gestures, and equipment anomaly detection. In terms of prediction, examples included profit and cost prediction, oil production estimation, and banking fraud prediction.

Clustering also appeared as an important ML usage purpose, representing 21.7% of projects. The examples provided included clustering personas to improve telemarketing services, geographic store segmentation, and separating users given their behaviors (*e.g.*, recommendation systems). Finally, the purpose of Association appears in 7% of the projects. For this purpose, the main reported examples involved cause-and-effect association. Computer vision topics, such as denoising and pose estimation, were discussed in the Others' field but were less representative.

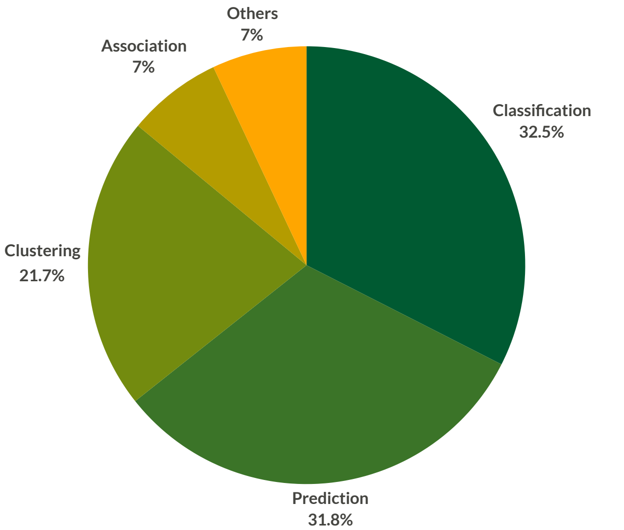


Figure 7. ML main purpose

4.2.6 ML-enabled system Main Algorithms

The most common algorithms in use for Brazilian ML-enabled system projects (Figure 8) are Neural Networks (e.g., CNN, LSTM) with 13.4 %, followed by Ensembles (e.g., XG-Boost, Random Forests) with 12.71 %, Decision Trees with 12.03 %, and KMeans with 12.03 % of the answers. Another significant proportion uses Logistic Regression (10.31 %), SVM (9.28 %), KNN (8.59 %), and Linear Regression (7.22 %). The least used algorithms are Naive Bayes (3.44 %), DB-SCAN (2.75 %), Apriori (1.72 %), Gaussian Mixture (1.37 %), and Bayesian Networks (1.03 %). Some options provided in the 'Others' field are just other examples of one of the previous options, such as LGBM and Catboosting, that could be informed in Ensembles or GNN and RNN, which are also types of neural networks. Other informed options were Time Series algorithms (ARIMA), dimensionality reduction techniques (PCA), and statistical decision processes (Markov Decision Process).

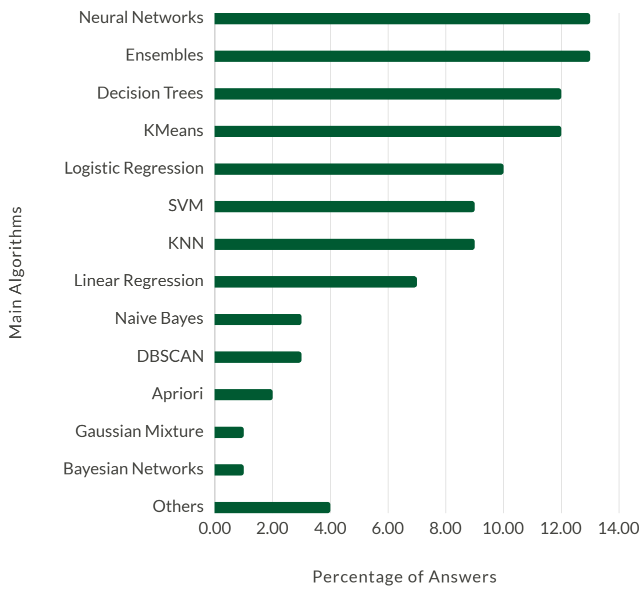


Figure 8. ML main algorithms

4.3 Problem Understanding and Requirements ML Life Cycle Stage

In the survey, based on the nine ML life cycle stages presented by Amershi et al. (2019) and the CRISP-DM industry-independent process model phases (Schröer et al., 2021), we abstracted seven generic life cycle stages (Kalinowski et al., 2023) and asked about their perceived relevance and difficulty. The answers presented in Figure 9 and 10 revealed that ML practitioners are extremely worried about requirements, given that the *Problem Understanding and Requirements* stage is clearly perceived as the most relevant and most complex life cycle stage.

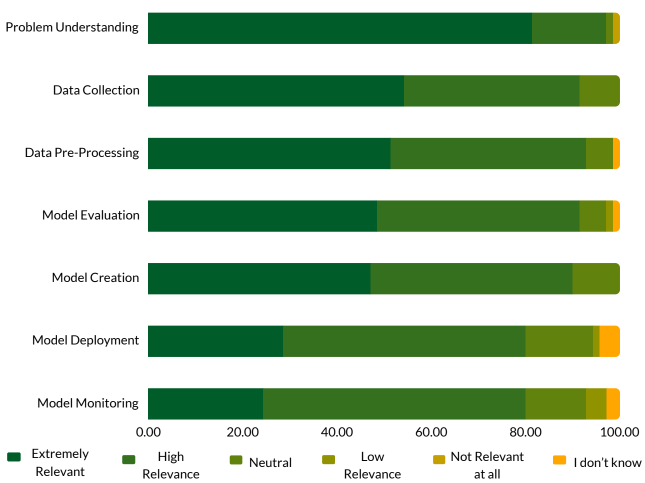


Figure 9. Perceived relevance of each ML life cycle stage

Figure 11 shows the median estimated effort of each ML life cycle stage. The *Problem Understanding and Requirements* stage is, again, the stage with more effort involved. Hence, it could be summarized as being perceived as the most relevant, complex, and effort-demanding ML life cycle phase.

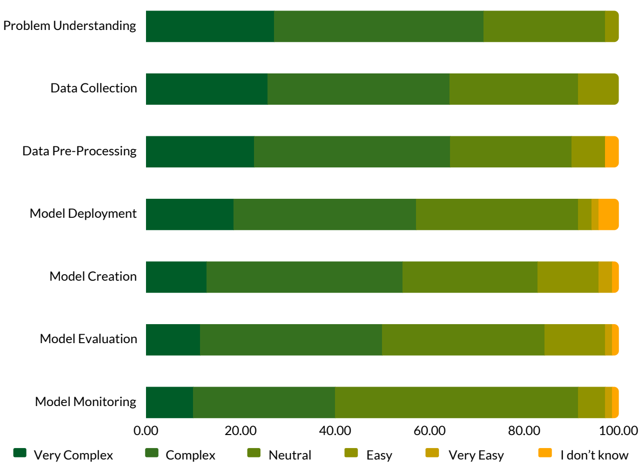


Figure 10. Perceived difficulty of each ML life cycle stage

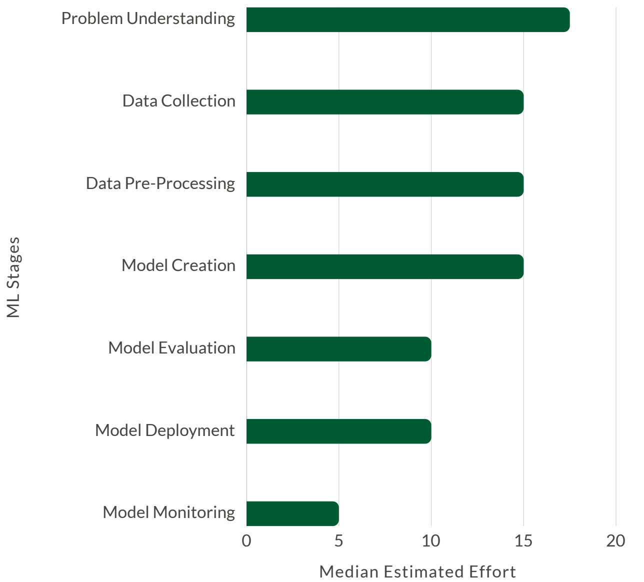


Figure 11. Estimated effort of each ML life cycle stage

4.4 Contemporary RE practices for ML-enabled Systems

In this subsection, we provide the results and analysis regarding contemporary RE practices for ML-enabled systems. In summary, we detail who is addressing requirements, how they are elicited and documented, the typical NFRs, and which activities are considered the most difficult ones.

4.4.1 [RQ1.1] Who is addressing the requirements of ML-enabled system projects?

The proportion of positions reported to address the requirements of ML-enabled system projects within the bootstrapped samples is shown in Figure 12 together with the 95% confidence interval. The *N* in each figure caption is the number of participants who answered this question. We report the proportion *P* of the participants that checked the corresponding answer and its 95% confidence interval in square brackets.

It is possible to observe that Data Scientists were most associated with requirements in ML-enabled systems with **P = 61.389 [60.955, 61.822]**, followed by Project Leaders (**P = 49.6 [49.219, 49.981]**), Business Analysts (**P = 28.339 [28.024, 28.653]**), and Developers (**P = 21.386 [21.061, 21.71]**). The less associated roles within requirements addressing were Solution Architects (**P = 11.563 [11.353, 11.773]**), Requirements Engineers (**P = 8.46 [8.281, 8.639]**), and Testers (**P = 1.481 [1.397, 1.566]**). Several isolated options were mentioned in the "Others" field (*e.g.*, Machine Learning Engineer, Data Lead, and Tech Lead), altogether summing up 14% and not significantly influencing the overall distribution (**P = 14.303 [14.032, 14.573]**).

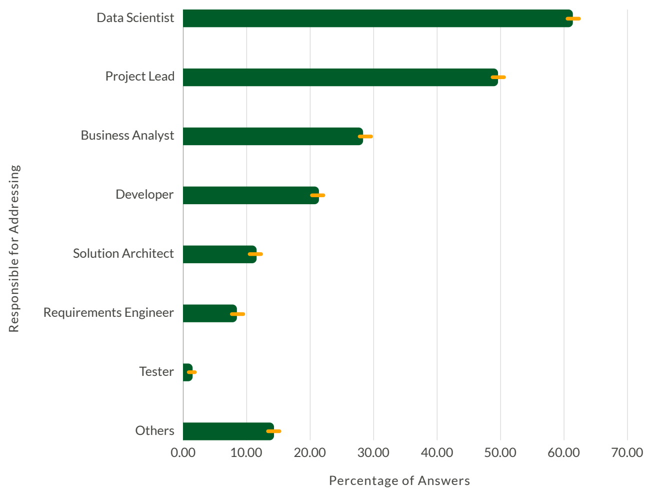


Figure 12. Roles addressing requirements of ML-enabled systems (N = 70)

4.4.2 [RQ1.2] How are requirements typically elicited in ML-enabled system projects?

As presented in Figure 13, practitioners reported interviews as the most commonly used technique (**P = 69.399 [69.062, 69.735]**), followed (or complemented) by workshops (**P = 47.296 [46.958, 47.634]**), prototyping (**P = 41.638 [41.292, 41.983]**), and scenarios (**P = 40.221 [39.841, 40.6]**). The least used elicitation technique was observation, with **P = 35.896 [35.535, 36.257]**. In the "Others" field, the Objective and Key Results (OKRs) system and informal meetings were mentioned, but with a much lower proportion (**P = 8.357 [8.156, 8.558]**).

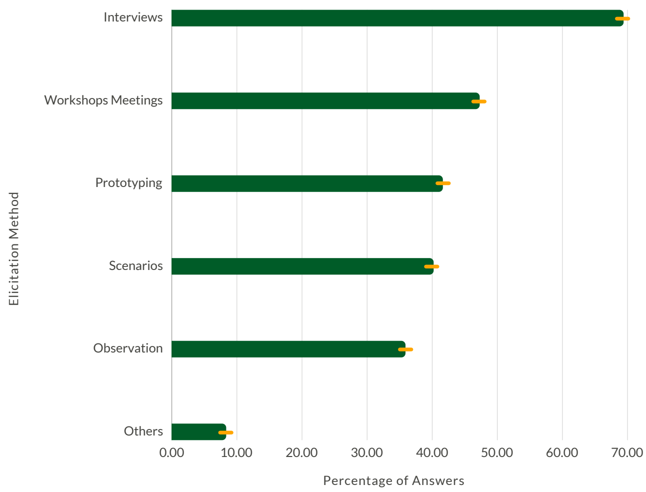


Figure 13. Requirements elicitation techniques of ML-enabled systems (N = 72)

4.4.3 [RQ1.3] How are requirements typically documented in the ML-enabled system projects?

Figure 14 shows Notebooks as the most frequently used documentation format with **P = 46.504 [46.129, 46.879]**, followed by User Stories (**P = 30.715 [30.374, 31.057]**), Vision Documents (**P = 21.304 [21.008, 21.6]**), Prototypes (**P = 21.182 [20.895, 21.468]**), Requirements Lists (**P = 19.713 [19.431, 19.994]**), and Data Models (**P = 19.669 [19.352, 19.986]**). Surprisingly, almost 17% mentioned that requirements are not documented at all with **P = 16.917 [16.632, 17.201]**. Some isolated options were mentioned in the "Others" field (*e.g.*, Notion, Github, and Confluence) with **P = 12.668 [12.429, 12.906]**.

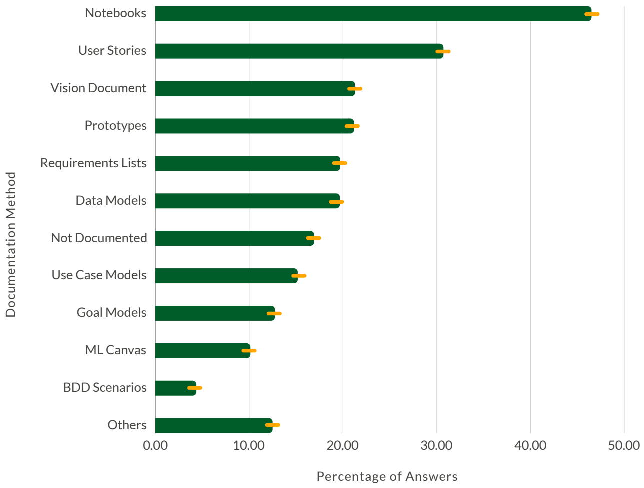


Figure 14. Requirements documentation of ML-enabled systems (N = 71)

4.4.4 [RQ1.4] Which Non-Functional Requirements (NFRs) do typically play a major role in terms of criticality in the ML-enabled system projects?

Regarding NFRs (Figure 15), practitioners show a significant concern with some ML-related NFRs, such as data quality (**P = 69.103 [68.75, 69.456]**), model explainability (**P = 37.825 [37.464, 38.187]**), and model reliability (**P = 36.721 [36.341, 37.101]**). Some NFRs regarding the whole system were also considered important, such as system performance (**P = 35.2 [34.874, 35.526]**), system maintainability (**P = 25.441 [25.122, 25.76]**), and system usability (**P = 25.175 [24.828, 25.521]**). A significant number of participants informed that NFRs were not considered within their ML-enabled system projects (**P = 12.623 [12.376, 12.869]**).

4.4.5 [RQ1.5] Which activities are considered most difficult when defining requirements for ML-enabled systems?

The answer options to this question were based on the literature regarding requirements (Wagner et al., 2019) and requirements for ML (Villamizar et al., 2021). Furthermore, we left the "Others" option to allow new activities to be added, but nothing new was informed. In this context, we show in Figure 16 that the respondents considered managing customer expectations is the most difficult task (**P = 71.554 [71.191, 71.916]**), followed by aligning requirements with

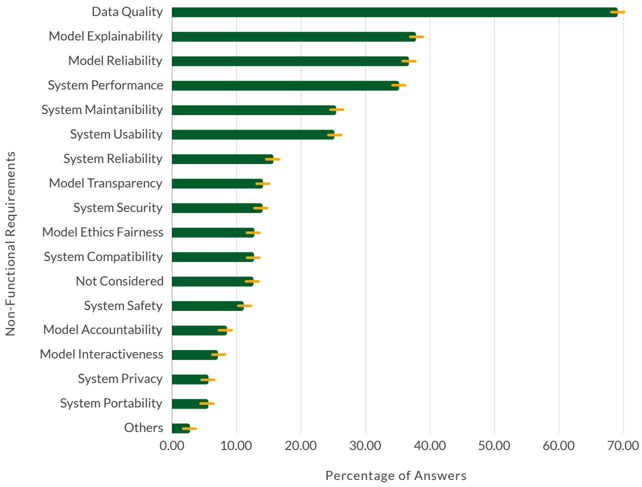


Figure 15. Critical non-functional requirements of ML-enabled systems (N = 71)

data (**P = 53.556 [53.197, 53.915]**), resolving conflicts (**P = 42.346 [41.987, 42.706]**), managing changing requirements (**P = 40.915 [40.574, 41.257]**), selecting metrics (**P = 32.079 [31.738, 32.42]**), and elicitation and analysis task (**P = 26.72 [26.418, 27.021]**).

Figure 16. Most difficult RE activities in ML-enabled systems (N = 71)

4.5 Main RE-related challenges in ML-enabled System Projects

In this subsection, we provide the results and analysis regarding RE-related challenges for ML-enabled systems. We detail the overall RE-related challenges and how these challenges vary in agile and traditional project contexts.

4.5.1 [RQ2.1] What are the overall main RE-related challenges?

Regarding the main concerns during each ML life cycle stage, we asked participants to inform up to three challenges related to each ML life cycle stage in an open-text answer. The main challenges related to the Problem Understanding and Requirements stage emerged from open coding applied to all of the 109 open-text answers provided for this stage.

We incorporated axial coding procedures to provide an easily understandable overview, relating the emerging codes to categories. We started with the sub-categories *Input*, *Method*, *Organization*, *People*, and *Tools*, as suggested for problems in previous work on defect causal analysis (Kalinowski et al., 2012). Based on the collected data, we merged the *Input* and *People* categories, as it was difficult to separate between the two, given the concise and short answers provided by the participants. We also renamed the *Tools* category into *Infrastructure* and identified the need to add a new category related to *Data*. It is noteworthy that these categories were identified considering the overall coding for the seven ML life cycle stages. At the same time, this paper focuses on the Problem Understanding and Requirements stage.

In Figure 17, we present an overview of the resulting codes' frequencies using a probabilistic cause-effect diagram (*e.g.*, fishbone diagram), which provides a comprehensive overview and it was introduced for causal analysis purposes in previous work (Kalinowski et al., 2010, 2011). The percentages are just frequencies of occurrence of the codes, then the sum of all code frequencies is 100%. It is important to notice that the highest frequencies within each category are organized closer to the middle.

It is possible to observe that most of the reported challenges are related to the *Input* category, followed by *Method* and *Organization*. The main challenges within the *Input* category report difficulties in understanding the problem, the business domain, and unclear goals and requirements. In the *Method* category, the prevailing reported challenges concern difficulties in managing expectations, experienced data science knowledge, and establishing effective communication. Finally, in the *Organization* category, the lack of customer or domain expert availability and engagement, and the lack of time dedicated to requirements-related activities were mentioned. Our summary focuses on the most frequently mentioned challenges, although less frequent ones may still be relevant in practice. For instance, computational constraints or lack of data quality and preprocessing can directly affect ML-related possibilities and requirements.

4.5.2 [RQ2.2] How do the RE-related challenges manifest in agile and traditional project contexts?

The previous research question *[RQ2.1]* provided an overview of the challenges practitioners face in ML-enabled systems. However, adopting an agile methodology affects the requirements engineering tasks when compared to traditional software development (Wagner et al., 2018). In order to better understand the reported RE-related challenges, we decided to look at the previous findings through different lenses, one for agile and one for traditional methodologies.

To analyze the RE-related challenges in agile teams, we filtered the responses from practitioners who informed us that their project followed a "Totally agile" or "Mostly agile" methodology (Question D11). On the other hand, for the RE-related challenges in traditional teams, we filtered the responses from practitioners who informed "Totally traditional" or "Mostly traditional". It is noteworthy that for this research question, we didn't consider the responses from participants who characterized their project agility as "Balanced between agile and traditional" or "I don't know".

Figure 18 shows the main problems reported within the context of agile projects. The *Input* category is the most challenging category for Brazilian practitioners. Within this category, understanding the problem is still the most representative challenge, followed by understanding the business domain and unclear goals. The most significant challenge within the *Organization* category still relies on low client and domain expert engagement or availability, followed by lack of time. Furthermore, in the *Method* category managing expectations remains as the most mentioned challenge. It is noteworthy that the two categories that did not appear within agile projects (infrastructure and data) were mentioned by only one participant of the overall sample (working in a traditional project context) and were also not representative of the overall scenario.

While the most frequently reported challenges within each category are basically the same, some subtle differences can be observed. Based on the frequencies, challenges related to problem understanding and customer engagement seem to be more frequent in agile projects, while challenges managing expectations are less frequent.

Figure 19 presents the challenges reported by practitioners working in a traditional project context. While the sample is much smaller (N=16), the results are consistent. In ML-enabled system projects, agile methods seem to improve the management of customer expectations compared to traditional approaches, but also make problem understanding and customer involvement more challenging.

5 Discussion

Our previous results reflected an international perspective regarding RE for ML-enabled systems (Alves et al., 2023). Given the importance of Brazil in this previous study and the growing interest of Brazilian companies in terms of ML, we bring a deeper and focused analysis of Brazil's practices, problems, and perceptions in this paper. We provide a comprehensive overview of Brazilian ML-enabled systems project characteristics.

We found that most ML-enabled system projects are conducted within agile contexts, with Scrum and Kanban being the most adopted management frameworks. Notably, CRISP-DM, which aims to support data-driven projects, such as ML projects, is one of the least used frameworks. We also observed that ML-enabled systems are being developed for a variety of application domains and for different purposes. Python is the main programming language in use to develop these solutions, and the most used ML algorithms are Neural Networks (e.g., RNN, CNN, LSTM), Ensembles (e.g., XGBoost, RandomForest, LightGBM), Decision Trees, and KMeans.

In terms of contemporary RE practices, we identified an intriguing distribution of roles that address requirements; in particular, we have data scientists taking the reins. It is less common to have a RE position in charge of requirements activities in companies (Wang et al., 2018). Instead, we have software engineers, business analysts, or project managers in charge of them (Herrmann, 2013). Unlike our previous findings (Alves et al., 2023), where we had project leaders mainly responsible for handling requirements, in Brazil, we

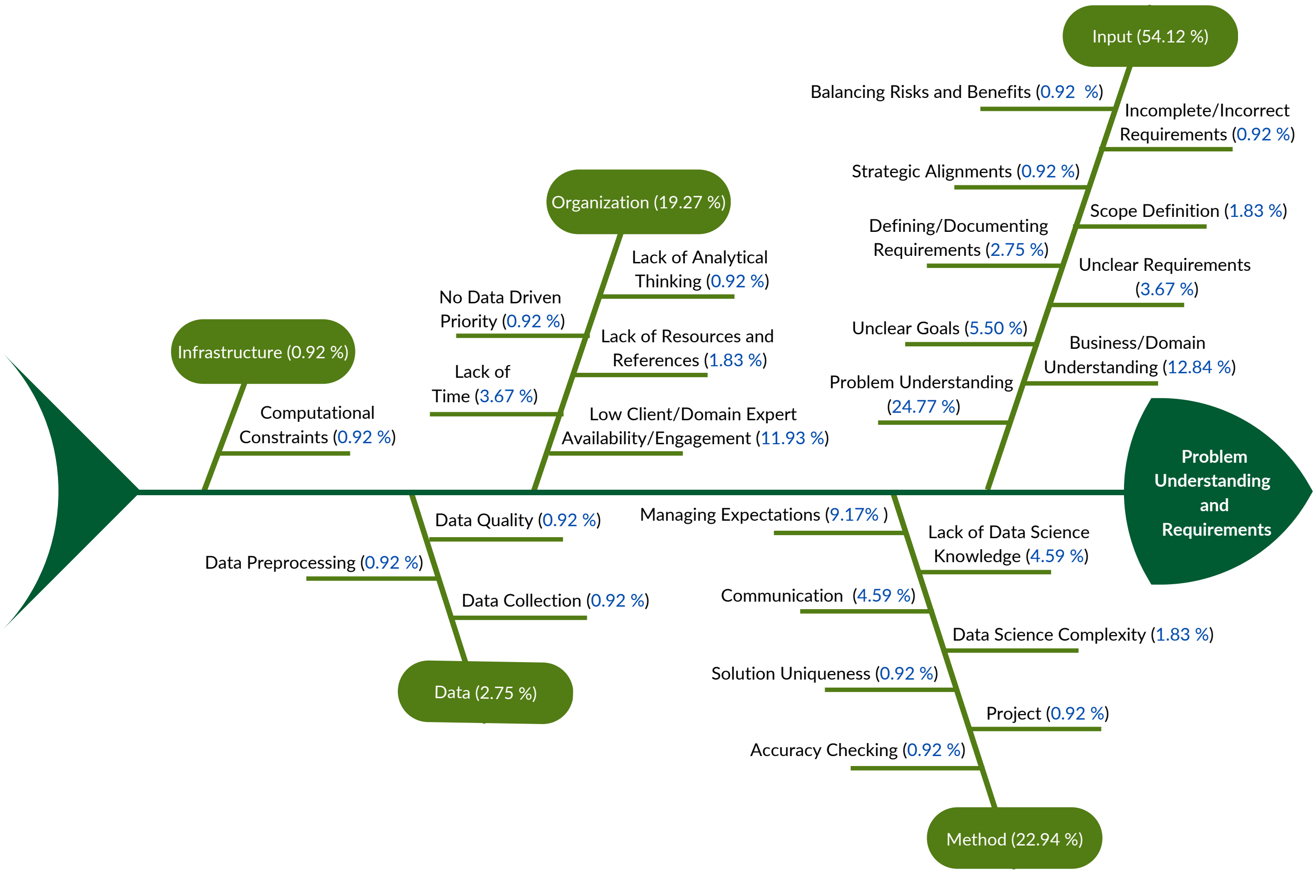


Figure 17. Main problems faced related to problem understanding and requirements (N=109)

have data scientists as the majority, which reassures their importance beyond coding in companies (Correia et al., 2021; Kim et al., 2017). The nature of ML-enabled systems is based on data-driven insights, which may explain the importance of addressing activity in this role. However, lacking well-established methods and practices in this domain may lead to project failure (Fernández et al., 2017).

Regarding elicitation techniques, our survey revealed again that practitioners don't escape from traditional requirements elicitation techniques (interviews, prototyping, scenarios, workshops, and observation), even with a free-text option available. Unlike our previous results where Workshops were less used (Alves et al., 2023), in Brazil, our results for the elicitation techniques are comparable to traditional RE (Wagner et al., 2019). This could be related to the fact that most practitioners work in large companies, which typically have professionals experienced in conducting such workshops for traditional software systems and have now extended these practices to ML-enabled systems.

In terms of requirements documentation, computational notebooks, which are interactive programming environments that can be used to process data and create ML models, appear, as reported previously (Alves et al., 2023), as the most used tool for documenting requirements. Its rapid way of producing and generating code turned notebooks into an important tool for data scientists; however, like a hammer, it could be misused (Perkel, 2018) as a symptom of the lack of awareness of RE specification practices and tools. Moreover, a proportion of almost 16% mentioned that requirements were not documented at all, which may cause overall software project failure (Fernández et al., 2017). In Brazil, we have reported that Vision Documents are more prominent than in other parts of the world, and despite being closely related to Prototypes, our previous finding had Requirements Lists as the third most used method, and now it appears as the sixth option. ML Canvas, which was designed to tackle this activity, is one of the least used methods, along with BDD Scenarios.

With respect to NFRs, there are slight differences between how worldwide practitioners face NFR (Alves et al., 2023) and how Brazilians do. In general, the most considered concerns are ML-related NFR, such as data quality, model reliability, and model explainability, as previously reported in (Habibullah et al., 2023; Vogelsang and Borg, 2019). However, we also observed system-related concerns like system performance, usability, and maintainability. In conventional software systems, there are several negative impacts of missing NFRs on software-related projects (Fernández et al., 2017). However, again, the same proportion of practitioners (more than 10%) do not even consider NFRs in their ML-enabled system projects, which can be seen as another indicator of the lack of overall attention to the importance of RE in the industrial ML-enabled systems engineering context.

The survey also revealed the most difficult activities perceived by practitioners in Brazil when defining requirements for ML-enabled systems. The difficulties reported by Brazilian practitioners are comparable to the ones reported previously (Alves et al., 2023) and with previous literature, but now it appears in a wider industrial scope. Managing customer expectations (Ishikawa and Yoshioka, 2019), aligning requirements with data (Nahar et al., 2023; Villamizar et al.,

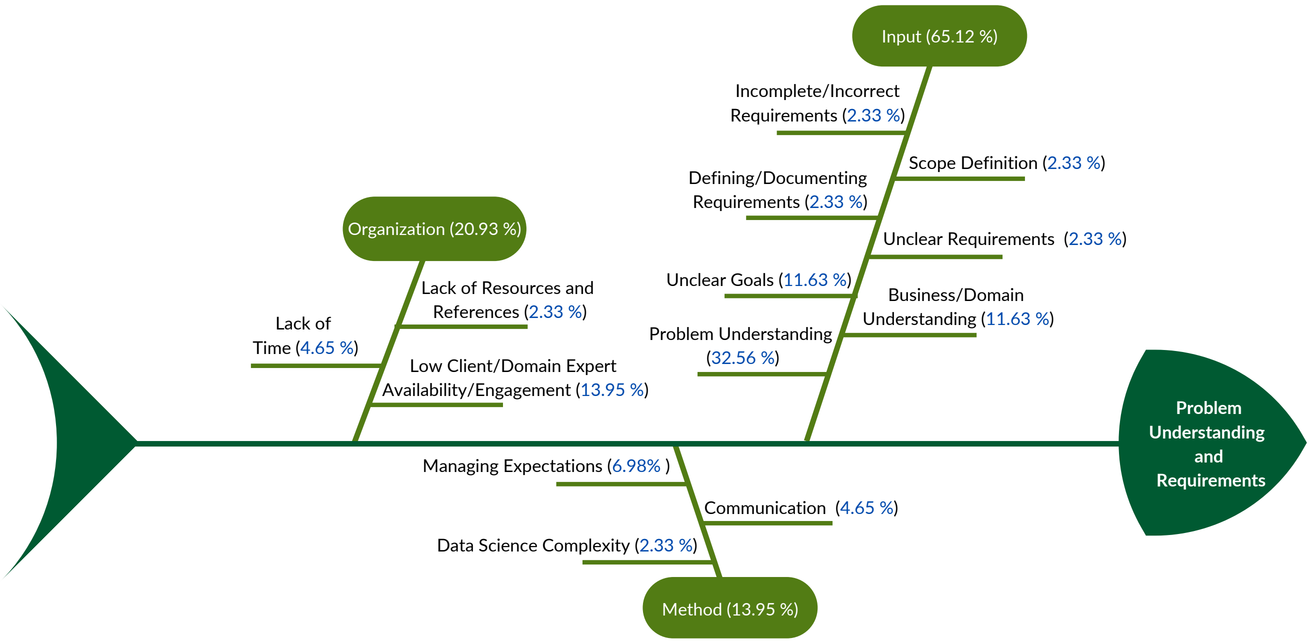


Figure 18. Main Problems faced during Problem Understanding and Requirements in Agile Project Context (N=43)

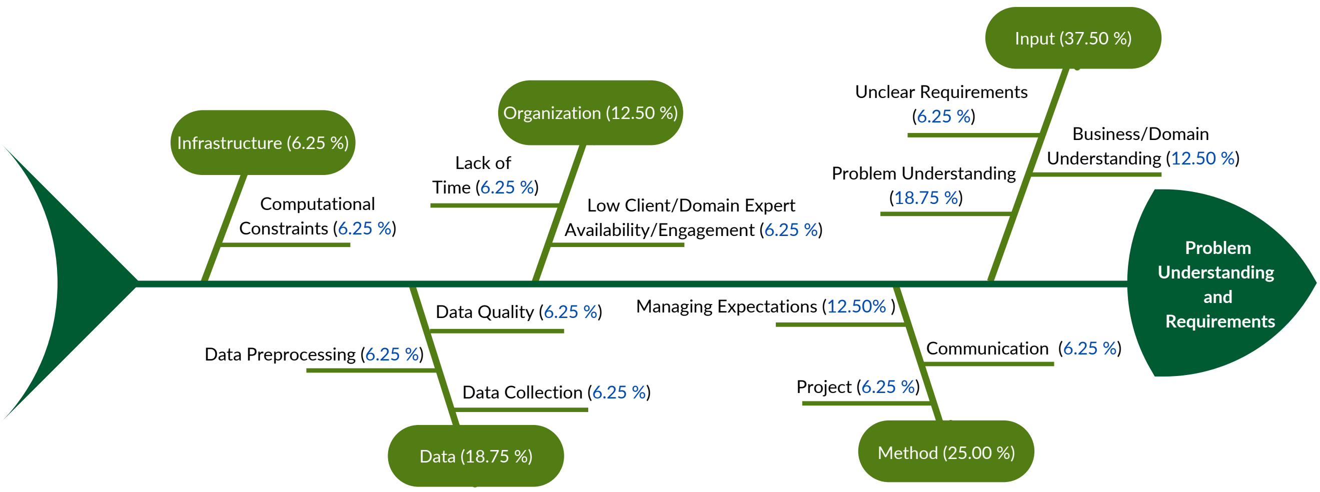


Figure 19. Main Problems faced during Problem Understanding and Requirements in Traditional Project Context (N = 16)

2021), changing requirements (Khalajzadeh et al., 2018), and selecting proper metrics (Vogelsang and Borg, 2019) were previously reported as difficulties, which emphasizes the importance of effective communication and technical expertise to bridge the gap between aspirations and technological feasibility.

Finally, we contributed to the RE-related problems faced by practitioners in ML-enabled system projects in Brazil. The main issues relate to difficulties in problem and business understanding, managing expectations, and low customer/domain expert availability/engagement. Literature has shown that agility affects RE-related practices and challenges (Wagner et al., 2017). Therefore, we decided to analyze the challenges faced by practitioners working in agile and traditional contexts. The results present intriguing insights. Given the sample size and the inherent threats to qualitative research, we cannot assert conclusion validity. However, in ML-enabled system projects, agile methods seem to improve the management of customer expectations compared to traditional approaches, while also introducing greater challenges in problem understanding and customer involvement. One possible explanation is that the expectation of rapid, iterative value delivery in agile projects may complicate the overall problem-understanding process. Additionally, the absence of more formal, plan-driven responsibilities could make it harder to ensure proper customer involvement. On the other hand, frequent deliveries provide opportunities for continuous feedback, which can help align customer expectations more effectively.

The reported challenges in Brazilian ML-enabled systems projects have comparable counterparts in the conventional RE problems (Fernández et al., 2017). Table 2 shows the strong relationship between problems in ML-enabled system projects and conventional software contexts. As comparable problems may have comparable solutions, adopting established RE practices (or adaptations of such practices) may help improve ML-enabled system engineering. However, proposing adapted practices, guidelines, or solutions will demand further empirical evaluations that are not in the scope of this paper.

Table 2. Comparison between problems on ML-enabled and conventional software systems

Traditional RE Problem	ML RE Problem
Incomplete and/or hidden requirements	[Input] Incomplete/incorrect requirements
Communication flaws between project team and customer	[Method] Communication
Moving targets (changing goals, business processes, and/or requirements)	[Input] Unclear goals
Underspecified requirements that are too abstract	[Input] Unclear requirements
Timeboxing/Not enough time in general	[Organization] Lack of time
Stakeholders with difficulties in separating requirements from known solution designs	[Organization] Lack of analytical thinking
Insufficient support by customer	[Organization] Low client/domain expert availability/engagement
Weak access to customer needs and/or business information	[Organization] Lack of resources and references

6 Threats to Validity

We identified some threats while planning, conducting, and analyzing the survey results. Henceforward, we list these potential threats organized by the survey validity types presented in Linaker et al. (2015).

Face and Content Validity. Face and content validity threats include bad instrumentation and inadequate explanation of constructs. To mitigate these threats, we involved several researchers in reviewing and evaluating the questionnaire regarding the format and formulation of the questions, piloting it with 18 Ph.D. students for face validity and five experienced data scientists for content validity.

Criterion Validity. Threats to criterion validity include not surveying the target population. We clarified the target population in the consent form (before starting the survey). We also considered only complete answers (*i.e.*, answers of participants who answered all four survey sections) and excluded participants who reported having no experience with ML-enabled system projects. Moreover, an important aspect is a possible bias in our results, given that only one requirements engineer answered our survey. We explain that the explicit Requirements Engineer position is uncommon even in conventional software engineering contexts and that mainly other positions, like Business Analysts, are responsible for RE-related tasks (Wang et al., 2018). Hence, we believe that not having many requirements engineers in our sample is expected and positive in terms of representativeness. It just reflects that they are typically not part of ML-enabled system projects.

Construct Validity. We ground our survey's questions and answer options on theoretical background from previous studies on RE (Fernández et al., 2017; Wagner et al., 2019) and a literature review on RE for ML (Villamizar et al., 2021). A threat to construct validity is inadequate measurement procedures and unreliable results. To mitigate this threat, we follow recommended data collection and analysis procedures (Wagner et al., 2020).

Reliability. One aspect of reliability is statistical generalizability. We could not construct a random sample that systematically covers all types of professionals involved in developing ML-enabled systems, as there is still no generalized understanding of what such a population looks like. Nevertheless, the experience and background profiles of the subjects are comparable to the profiles of ML teams, as shown in Microsoft's study (Kim et al., 2017). We used bootstrapping to deal with the random sampling limitation and only employed confidence intervals, conservatively avoiding null hypothesis testing. Another reliability aspect concerns interobserver reliability, which we improved by including independent peer review in all our qualitative analysis procedures and making all the data and analyses openly available online Alves et al. (2025).

7 Conclusion

Literature suggests that RE can help to tackle challenges in ML-enabled system engineering (Villamizar et al., 2021). Recent literature studies (*e.g.*, (Ahmad et al., 2021; Nahar et al., 2023; Villamizar et al., 2021)) and industrial studies (*e.g.*, (Correia et al., 2021; Vogelsang and Borg, 2019)) on RE for ML-enabled systems have been important to help understand the literature focus and industry needs. However, the study of industrial practices, perceptions, and challenges is still isolated and not yet representative.

We build upon prior research to enhance the empirical evidence on current practices, perceptions, and challenges in the field of RE for ML. This study analyzes a subset of data from our previous study, which presented an international survey (Alves et al., 2023), focusing on responses from 72 practitioners involved in the development of ML-enabled systems in Brazil. We applied bootstrapping with confidence intervals for quantitative statistical analysis and open and axial coding for qualitative analysis of RE challenges. The results reinforce the findings of previous studies (Habibullah et al., 2023; Vogelsang and Borg, 2019), emphasizing the importance of non-functional requirements, such as data quality, model reliability, and explainability. They also highlight challenges, including managing customer expectations and addressing ambiguities in requirements specifications (Nahar et al., 2023; Villamizar et al., 2021).

In addition, the analysis of data uncovered several new and noteworthy aspects. Notably, data scientists are leading RE activities in the development of ML-enabled systems, with interactive notebooks serving as a primary method for documenting requirements. The survey also highlighted several challenges faced by practitioners, such as difficulties in problem and business understanding, difficulties in managing expectations, unclear requirements, and a lack of domain expert availability and engagement.

These challenges are further influenced by the development approach. Agile methods, while beneficial in improving the management of customer expectations through frequent feedback loops and iterative delivery, also present new obstacles. The fast-paced nature of agile projects can make it harder to establish a clear and shared understanding of the problem, particularly in ML-enabled systems where requirements are often uncertain or evolving. Moreover, the reduced emphasis on formal, plan-driven processes may contribute to difficulties in engaging domain experts and ensuring their sustained involvement throughout the project.

Overall, when comparing RE practices and challenges within ML-enabled systems with conventional RE practices (Wagner et al., 2019) and challenges (Fernández et al., 2017), we identified significant variations in the practices but comparable underlying problems. Proposing solutions for these problems is part of future research and is not in the scope of this paper, as it would require proper empirical evaluations through different empirical strategies (*e.g.*, action research, case studies, controlled experiments). However, we truly believe that comparable challenges may have comparable solutions. In this sense, we advocate for adapting and disseminating RE-related practices for engineering ML-enabled systems.

Declarations

The contributions of the authors, following the CREDIT Taxonomy (https://credit.niso.org/), follow: Antonio Pedro Santos Alves: Data curation, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. Marcos Kalinowski: Conceptualization, Investigation, Methodology, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. Daniel Mendez: Conceptualization, Methodology, Validation, Writing – review & editing. Hugo Villamizar: Conceptualization, Methodology, Writing – review & editing. Tatiana Escovedo: Conceptualization, Writing – review & editing. Helio Lopes: Conceptualization.

Declaration of Competing Interest

The authors declare that they have no competing interests.

Acknowledgements

The authors would like to thank the international survey team Kalinowski et al. (2025). While this paper focuses on the Brazilian context, they contributed to conceptualizing and reviewing the survey instrument. Thanks also to all the Brazilian survey respondents.

Funding

We express our gratitude to CAPES (Finance code 001), CNPq (Grant 312275/2023-4), FAPERJ (Grant E-26/204.256/2024), Instituto Kunumi - project *AI Engineering*, and Instituto Stone - project *AI Engineering*, for their generous support.

Data Availability

All datasets and analysis scripts are available in our open science repository (Alves et al., 2025).

References

Ahmad, K., Bano, M., Abdelrazek, M., Arora, C., and Grundy, J. (2021). What's up with requirements engineering for artificial intelligence systems? In *2021 IEEE 29th International Requirements Engineering Conference*, pages 1–12. IEEE.

Aho, T., Sievi-Korte, O., Kilamo, T., Yaman, S., and Mikkonen, T. (2020). Demystifying data science projects: A look on the people and process of data science today. In *Product-Focused Software Process Improvement: 21st International Conference, PROFES 2020, Turin, Italy, November 25–27, 2020, Proceedings 21*, pages 153–167. Springer.

Alves, A. P. S., Kalinowski, M., Giray, G., Mendez, D., Lavesson, N., Azevedo, K., Villamizar, H., Escovedo, T., Lopes, H., Biffl, S., et al. (2023). Status quo and problems of requirements engineering for machine learning: Results from an international survey. In *Product-Focused Software Process Improvement: 24st International Conference, PROFES 2023, Dornbirn, Austria, December 10–13*, pages 153–167. Springer.

Alves, A. P. S., Kalinowski, M., Mendez, D., Villamizar, H., Azevedo, K., Escovedo, T., and Lopes, H. (2024). Industrial practices of requirements engineering for ml-enabled systems in brazil. In *Anais do XXXVIII Simpósio Brasileiro de Engenharia de Software*, pages 224–233, Porto Alegre, RS, Brasil. SBC.

Alves, A. P. S., Kalinowski, M., Méndez, D., Villamizar, H., Escovedo, T., and Lopes, H. (2025). Artifacts: Industrial practices of requirements engineering for ml-enabled systems in brazil: An extended analysis. available at: https://doi.org/10.5281/zenodo.15029764.

Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., and Zimmermann, T. (2019). Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference* *on Software Engineering: Software Engineering in Practice*, pages 291–300. IEEE.

Beede, E., Baylor, E., Hersch, F., Iurchenko, A., Wilcox, L., Ruamviboonsuk, P., and Vardoulakis, L. M. (2020). A human-centered evaluation of a deep learning system deployed in clinics for the detection of diabetic retinopathy. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–12.

Challa, H., Niu, N., and Johnson, R. (2020). Faulty requirements made valuable: On the role of data quality in deep learning. In *2020 IEEE 7th International Workshop on Artificial Intelligence for Requirements Engineering*, pages 61–69. IEEE.

Correia, J. a. L., Pereira, J. A., Mello, R., Garcia, A., Fonseca, B., Ribeiro, M., Gheyi, R., Kalinowski, M., Cerqueira, R., and Tiengo, W. (2021). Brazilian data scientists: Revealing their challenges and practices on machine learning model development. In *Proceedings of the XIX Brazilian Symposium on Software Quality*, SBQS '20, New York, NY, USA. Association for Computing Machinery.

Dalpiaz, F. and Niu, N. (2020). Requirements engineering in the days of artificial intelligence. *IEEE Software*, 37(4):7–10.

Damian, D. (2007). Stakeholders in global requirements engineering: Lessons learned from practice. *IEEE software*, 24(2):21–27.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap*. Chapman & Hall/CRC.

Fernández, D. M., Wagner, S., Kalinowski, M., Felderer, M., Mafra, P., Vetrò, A., Conte, T., Christiansson, M.-T., Greer, D., Lassenius, C., et al. (2017). Naming the pain in requirements engineering: Contemporary problems, causes, and effects in practice. *Empirical Software Engineering*, 22:2298–2338.

Fry, H. (2018). *Hello World: How to be Human in the Age of the Machine*. Random House.

Gartner (2020). Gartner identifies the top strategic technology trends for 2021.

Giray, G. (2021). A software engineering perspective on engineering machine learning systems: State of the art and challenges. *Journal of Systems and Software*, 180:111031.

Habibullah, K. M., Gay, G., and Horkoff, J. (2023). Non-functional requirements for machine learning: Understanding current use and challenges among practitioners. *Requirements Engineering*, 28(2):283–316.

Herrmann, A. (2013). Requirements engineering in practice: There is no requirements engineer position. In Doerr, J. and Opdahl, A. L., editors, *Requirements Engineering: Foundation for Software Quality*, pages 347–361, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ishikawa, F. and Matsuno, Y. (2020). Evidence-driven requirements engineering for uncertainty of machine learning-based systems. In *2020 IEEE 28th International Requirements Engineering Conference*, pages 346–351.

Ishikawa, F. and Yoshioka, N. (2019). How do engineers perceive difficulties in engineering of machine-learning systems? - questionnaire survey. *2019 IEEE/ACM Joint 7th International Workshop on Conducting Empirical Studies in Industry (CESI) and 6th International Workshop on Software Engineering Research and Industrial Practice (SER&IP)*, pages 2–9.

Jordan, M. I. and Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260.

Kalinowski, M., Card, D. N., and Travassos, G. H. (2012). Evidence-based guidelines to defect causal analysis. *IEEE Software*, 29(4):16–18.

Kalinowski, M., Escovedo, T., Villamizar, H., and Lopes, H. (2023). *Engenharia de Software para Ciência de Dados: Um guia de boas práticas com ênfase na construção de sistemas de Machine Learning em Python*. Casa do Código.

Kalinowski, M., Mendes, E., Card, D. N., and Travassos, G. H. (2010). Applying dppi: A defect causal analysis approach using bayesian networks. In *Product-Focused Software Process Improvement: 11th International Conference, PROFES 2010*, pages 92–106. Springer.

Kalinowski, M., Mendes, E., and Travassos, G. H. (2011). Automating and evaluating probabilistic cause-effect diagrams to improve defect causal analysis. In *Product-Focused Software Process Improvement: 12th International Conference, PROFES 2011*, pages 232–246. Springer.

Kalinowski, M., Mendez, D., Giray, G., Alves, A. P. S., Azevedo, K., Escovedo, T., Villamizar, H., Lopes, H., Baldassarre, T., Wagner, S., et al. (2025). Naming the pain in machine learning-enabled systems engineering. *Information and Software Technology*, 187:107866.

Khalajzadeh, H., Abdelrazek, M., Grundy, J., Hosking, J., and He, Q. (2018). A survey of current end-user data analytics tool support. In *2018 IEEE International Congress on Big Data*, pages 41–48. IEEE.

Kim, M., Zimmermann, T., DeLine, R., and Begel, A. (2017). Data scientists in software teams: State of the art and challenges. *IEEE Transactions on Software Engineering*, 44(11):1024–1038.

Kuwajima, H., Yasuoka, H., and Nakae, T. (2020). Engineering problems in ml systems. *Machine Learning*, 109(5):1103–1126.

Lei, S. and Smith, M. (2003). Evaluation of several nonparametric bootstrap methods to estimate confidence intervals for software metrics. *IEEE Transactions on Software Engineering*, 29(11):996–1004.

Lewis, G. A., Bellomo, S., and Ozkaya, I. (2021a). Characterizing and detecting mismatch in machine-learning-enabled systems. In *2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)*, pages 133–140. IEEE.

Lewis, G. A., Ozkaya, I., and Xu, X. (2021b). Software architecture challenges for ml systems. In *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 634–638.

Linaker, J., Sulaman, S. M., Höst, M., and de Mello, R. M. (2015). Guidelines for conducting surveys in software engineering v. 1.1. *Lund University*, 50.

Lunneborg, C. E. (2001). Bootstrap inference for local populations. *Therapeutic Innovation & Regulatory Science*, 35(4):1327–1342.

Martínez-Fernández, S., Bogner, J., Franch, X., Oriol, M., Siebert, J., Trendowicz, A., Vollmer, A. M., and Wagner, S. (2022). Software engineering for ai-based systems: a survey. *ACM Transactions on Software Engineering and Methodology*, 31(2):1–59.

Mitchell, T. M. (1997). Machine learning.

Nahar, N., Zhang, H., Lewis, G., Zhou, S., and Kästner, C. (2023). A meta-summary of challenges in building products with ml components – collecting experiences from 4758+ practitioners. In *2023 IEEE/ACM 2nd International Conference on AI Engineering – Software Engineering for AI (CAIN)*, pages 171–183.

Nahar, N., Zhou, S., Lewis, G., and Kästner, C. (2022). Collaboration challenges in building ml-enabled systems: Communication, documentation, engineering, and process. In *Proceedings of the 44th International Conference on Software Engineering*, pages 413–425.

Perkel, J. (2018). Why jupyter is data scientists' computational notebook of choice. *Nature*, 563:145–146.

Schröer, C., Kruse, F., and Gómez, J. M. (2021). A systematic literature review on applying crisp-dm process model. *Procedia Computer Science*, 181:526–534.

Sharma, L. and Garg, P. K. (2021). Artificial intelligence: technologies, applications, and challenges.

Stol, K.-J., Ralph, P., and Fitzgerald, B. (2016). Grounded theory in software engineering research: a critical review and guidelines. In *Proceedings of the 38th International Conference on Software Engineering*, pages 120–131.

Villamizar, H., Escovedo, T., and Kalinowski, M. (2021). Requirements engineering for machine learning: A systematic mapping study. In *2021 47th Euromicro Conference on Software Engineering and Advanced Applications*, pages 29–36.

Villamizar, H., Kalinowski, M., Lopes, H., and Mendez, D. (2024). Identifying concerns when specifying machine learning-enabled systems: A perspective-based approach. *Journal of Systems and Software*, 213:112053.

Vogelsang, A. and Borg, M. (2019). Requirements engineering for machine learning: Perspectives from data scientists. In *2019 IEEE 27th International Requirements Engineering Conference Workshops*, pages 245–251.

Wagner, S., Fernández, D. M., Felderer, M., and Kalinowski, M. (2017). Requirements engineering practice and problems in agile projects: Results from an international survey. In *Proceedings of the XX Iberoamerican Conference on Software Engineering, Buenos Aires, Argentina, May 22-23, 2017*, pages 389–402. Curran Associates.

Wagner, S., Fernández, D. M., Felderer, M., Vetrò, A., Kalinowski, M., Wieringa, R., Pfahl, D., Conte, T., Christiansson, M.-T., Greer, D., Lassenius, C., Männistö, T., Nayebi, M., Oivo, M., Penzenstadler, B., Prikladnicki, R., Ruhe, G., Schekelmann, A., Sen, S., Spínola, R., Tuzcu, A., Vara, J. L. D. L., and Winkler, D. (2019). Status quo in requirements engineering: A theory and a global family of surveys. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 28(2).

Wagner, S., Fernández, D. M., Kalinowski, M., and Felderer, M. (2018). Agile requirements engineering in practice: Status quo and critical problems. *CLEI Electronic Journal*, 21(1):6–1.

Wagner, S., Mendez, D., Felderer, M., Graziotin, D., and Kalinowski, M. (2020). Challenges in survey research. *Contemporary Empirical Methods in Software Engineering*, pages 93–125.

Wang, C., Cui, P., Daneva, M., and Kassab, M. (2018). Understanding what industry wants from requirements engineers: an exploration of re jobs in canada. In *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, ESEM '18, New York, NY, USA. Association for Computing Machinery.

Zimelewicz, E., Kalinowski, M., Mendez, D., Giray, G., Santos Alves, A. P., Lavesson, N., Azevedo, K., Villamizar, H., Escovedo, T., Lopes, H., Biffl, S., Musil, J., Felderer, M., Wagner, S., Baldassarre, T., and Gorschek, T. (2024). Ml-enabled systems model deployment and monitoring: Status quo and problems. In *Software Quality as a Foundation for Security*, pages 112–131, Cham. Springer Nature Switzerland.