Non-Functional Requirements for Machine Learning: Understanding Current Use and Challenges Among Practitioners

Khan Mohammad Habibullah, Gregory Gay and Jennifer Horkoff

Department of Computer Science and Engineering, Chalmers | University of Gothenburg, Gothenburg, Sweden.

Contributing authors: khan.mohammad.habibullah@gu.se; greg@greggay.com; jennifer.horkoff@gu.se;

Abstract

Systems that rely on Machine Learning (ML systems) have differing demands on quality—known as non-functional requirements (NFRs)—from traditional systems. NFRs for ML systems may differ in their definition, measurement, scope, and comparative importance. Despite the importance of NFRs in ensuring the quality ML systems, our understanding of all of these aspects is lacking compared to our understanding of NFRs in traditional domains. We have conducted interviews and a survey to understand how NFRs for ML systems are perceived among practitioners from both industry and academia. We have identified the degree of importance that practitioners place on different NFRs, including cases where practitioners are in agreement or have differences of opinion. We explore how NFRs are defined and measured over different aspects of a ML system (i.e., model, data, or whole system). We also identify challenges associated with NFR definition and measurement. Finally, we explore differences in perspective between practitioners in industry, academia, or a blended context. This knowledge illustrates how NFRs for ML systems are treated in current practice, and helps to guide future RE for ML efforts.

Keywords: Non-Functional Requirements, NFRs, Qualities, Machine Learning, NFR Challenges, Requirements Engineering

1 Introduction

Machine Learning (ML) is increasingly used in as part of complex systems that perform decision-making and prediction tasks, including image recognition, language processing, and autonomous systems. ML systems include algorithms that use large amounts of data to learn and automatically perform tasks that are challenging in traditional software [54].

ML systems are not trivial to develop, and their correctness and quality must meet strict requirements. Such systems often influence critical decision making (e.g., cancer detection and loan approval), and such decisions may suffer from unintended bias [37], unsafe execution [18], or unexplainable decisions [13]. Such systems often exhibit non-deterministic behavior, and exhaustive testing is expensive and time-consuming—if it is even possible in the first place.

Because of these risks, ensuring the successful development of ML systems is challenging. Therefore, despite the advances allowed by ML, much recent attention has been paid to certain qualities of ML systems—particularly regarding fairness [37], transparency [13], privacy [14], security [42], and safety [18]. From a Requirement Engineering (RE) perspective, these quality aspects are known as non-functional requirements (NFRs) [20]. Glinz defines NFRs as "an attribute of, or a constraint on, a system", where attributes are performance or quality requirements [29].

For more than 40 years, research has focused on how to define, measure, and assess NFRs in an effective way as part of RE and software development, e.g., [17]. Although much work has been devoted to NFRs [29], ensuring the attainment of NFRs remains a difficult challenge in modern system development [36]. Despite challenges, progress on NFRs has been made, including, for example, definitions (e.g., [29]), taxonomies (e.g., [28]), classification methods (e.g., [21]), modeling approaches (e.g., [20]), management methods (e.g., [38]), and industrial studies (e.g., [25]) for traditional systems.

However, when considering ML-enabled software, it is not clear if our accumulated knowledge concerning NFRs is still applicable. Some NFRs, such as fairness (e.g., [37]) or bias (e.g., [45]), become more important for ML systems, while others—such as privacy—remain equally relevant. Others, such as usability, become less important. As-yet-unexplored NFRs such as "retrainability"—the ability to re-train a model using new training data—may also emerge. Furthermore, the meaning and interpretation of NFRs may differ from their interpretation for traditional systems and may not yet be well understood [12]. In order to begin to reconsider our knowledge of NFRs, we first need to understand the state of practice concerning NFRs for ML systems.

Existing research has begun to look at challenges with ML use in practice. According to a recent survey, RE is the most challenging activity for ML system development [35]. Research has been done to identify how SE knowledge can be applied to ML system development and engineering challenges for ML systems. [6, 52] In an ML system, the software development process has become more complex and less well-defined—therefore, a large quantity of data is needed

to satisfy quality requirements [56]. Recent studies have also identified and discussed RE-related challenges in ML systems [8, 33, 56]. Still, there is a need to connect such research to practice. We have been unable to find work focusing on NFRs for ML systems from the practitioner perspective.

An additional factor not well explored in emerging research is that NFRs can be defined over different granularity levels of a system—i.e., we can define NFRs for a whole system, a component, or a specific feature. ML is often only a small part of a larger system [52]. NFRs can be identified and measured over ML-related data (e.g., training or test data), over the ML model, or over the whole ML system. We are interested in exploring the scope of how NFRs are defined or measured over specific parts of a ML system. We are also interested in understanding whether distinctions in scope from traditional systems are reflected in practice, and where NFR-related challenges lie for practitioners.

To begin to address these and other questions, we conducted a series of interviews and a broader survey with practitioners to explore their perceptions of NFRs in an ML context. By "practitioner", we refer to software developers in either industry or academia with experience related to defining, measuring, or assessing NFRs during the development of ML systems. The interview and survey covered the importance of NFRs for ML-enabled software, how NFRs are captured and measured, and what challenges the participants face to working with and measuring NFRs for ML.

By conducting this study, we identified the importance of NFRs for ML system development, gained insight in how NFRs are defined over parts of ML systems, and explored challenges related to NFR definition:

- Most participants agreed that NFRs are important in ensuring ML system quality, and that there are differences in how NFRs are defined and measured from traditional systems (e.g., adaptability, maintainability).
- Accuracy, reliability, integrity, and security are particularly important for ML systems. Most NFRs for traditional software are still relevant, while a few become less prominent (e.g., revision, transition). Perception on the importance of efficiency, fairness, flexibility, portability, reusability, testability, and usability are split among participants.
- Most practitioners focused on defining NFRs over the whole system. Several also define NFRs on models. Few have considered NFRs for data.
- NFR challenges relate to uncertainty, domain dependence, awareness, regulations, dependency among requirements, and specific NFRs (e.g., safety, transparency, and completeness). Specific challenges may not emerge in all projects, but are common in some projects.

We also gained insight on how NFRs are measured over parts of a ML system, how NFR measurements are captured in the ML context, and what NFR measurement challenges exist:

- Some NFRs (e.g., accuracy) can be measured using ML-specific or standard measures, but many are difficult to measure (e.g, fairness, explainability) because they are not easily quantifiable. In safety-critical situations, both human and machine judgement should be used.

- Interviewees expressed a preference towards measurements over the model, while survey participants indicated the whole system.
- NFR measurement depends on context, and measurement can be dependent on another NFR defined for other system elements.
- Interviewees capture NFR measurements using checklists, interviews, scripting, and traceability tools. Context is important. Multiple participants found this question difficult to answer.
- Measurement challenges include a lack of knowledge or practices, missing measurement baselines, a complex ecosystem, data quality, cost of testing, bias in results, and domain dependence.

Finally, we examine the differences in perceptions of NFRs based on whether participants come from an industrial, academic, or blended context:

- Participants from academia offered the most consistent ratings of the importance of NFRs, but also the lowest. They placed higher importance on fairness, maintainability, and transparency than industry.
- Participants from industry highly value reliability, accuracy, and integrity. They place higher importance on justifiability, interoperability, and interpretability than academics.
- Participants from a blended context placed high importance on fairness, transparency, explainability, justifiability, and privacy. They placed the highest average importance on NFRs, but also had the largest variance. They placed low emphasis on fault tolerance, portability, and simplicity.
- Regarding NFR challenges, academic participants showed stronger agreement regarding domain dependency and lack of awareness among customers than industrial participants, while industrial participants showed stronger agreement on rigorous testing. Industrial participants were split on lack of awareness among customers. The blended group was particularly split on lack of awareness among engineers, and agreed more weakly than the others on the other challenges.
- All three groups largely agreed with statements regarding NFR measurement dependencies and challenges. However, those from the blended group had more disagreements on dependencies, and weaker agreement on challenges (more "agree" than "strongly agree").

This research study is an extension of a published conference paper [31]. The initial study contained the interview study. In this extended study, we validate and extend the results of the interview study with a broader survey. Using the survey, we identified the degree of importance of each NFR, made additional observations regarding scope and challenges, performed a more detailed qualitative analysis of practitioner experiences, and enabled comparison of perspectives between industrial and academic practitioners.

In Section 2, we present related work. We explain our research questions and methodology in Section 3. We then present our findings in Section 4. Section 5 discusses our findings, threats to validity, and future work. We conclude our study in Section 6.

2 Related Work

In this section, we highlight relevant related work topics, including NFRs, SE for AI, and work on RE for AI.

NFRs: Although NFRs are considered essential and critical for ensuring software quality, there are no agreed-upon guidelines on when and how NFRs should be elicited, documented, and validated [29]. Moreover, there is no consensus in the requirements engineering (RE) community on when NFRs should be considered and applied in the RE process [20]. Although a complete overview of NFR work is out of the scope of this work, we highlight some representative approaches.

Doerr et al. presented the application of a systematic, experience-based method to elicit, document, and analyze non-functional requirements. Their objective was to achieve a sufficient set of measurable and traceable non-functional requirements [24]. Ameller et al. conducted an interview-based survey with 18 different companies from six European countries. They presented the barriers to, and benefits of, the management of NFRs, how NFRs are supported by Model-Driven Development (MDD), and which strategies are followed if some NFRs are not supported by MDD approaches. Their results show that MDD adaptation is a complex process with little or no support for NFRs, and productivity and maintainability should be supported when MDD is adopted [2]. Sachdeva et al. conducted an industrial case study and proposed a novel approach for handling performance NFRs and security for big data and cloud-related projects using Scrum. The results show that their approach helps achieve performance and security requirements both individually and accounting for conflicts between them in an agile methodology [51].

Quality requirements can be a key competitive advantage for market-driven software development organizations, but an increase in quality is not linear with cost increases. The QUPER (Quality Performance) model estimates cost-benefit breakpoints and barriers in quality [49]. Svensson et al. performed a case study in the mobile handset domain to evaluate guidelines on applying QUPER in practice, including the process of handling cost dependencies between quality requirements [9]. Although relevant, this body of work has mainly focused on NFRs for traditional software systems.

SE for AI: Research has looked at how SE knowledge can be applied to AI and ML system development. Previous work in collaboration with companies has identified software engineering challenges for deep learning [6]. The research used seven ML projects and identified twelve challenges, categorized into three areas: development challenges, production challenges, and organizational challenges. An empirical investigation on a taxonomy of SE challenges for ML systems has been presented by Lwakatare et al. [39]. The challenges include problem formulation and specifying the desired outcome, use of a non-representative dataset, lack of well-established ground truth, no deployment

mechanism, and difficulties in building a highly scalable ML pipeline. The challenges were mapped into a proposed taxonomy that depicts the evolution of the use of ML components in the software-intensive system in industrial settings.

Amershi et al. conducted a case study observing software teams at Microsoft to identify their AI integration capabilities into software and services as they develop AI-based applications [3]. They considered a nine-stage ML workflow process and found that various Microsoft teams have integrated this workflow into existing Agile-like processes. This provides insights about several engineering challenges that an organization may face in developing large-scale AI solutions. They identified three aspects of AI systems that differ from traditional systems, including (1) discovering, managing and versioning data for ML systems is more complex and difficult than traditional systems, (2) model customization and model reuse for ML requires a different skill-set than traditional systems, and (3), AI components are more difficult to manage than traditional software components. Siebert et al. presented a quality model (quality aspects and evaluation objects) for software-intensive systems containing one or more components that use ML in an industrial use case [53].

Martínez-Fernández et al. conducted a systematic mapping study to collect and analyze knowledge about Software Engineering for AI-based systems. They found that the most-studied properties of AI-based systems are safety and dependability, software testing and software quality related studies are prevalent, and software maintanance related studies are neglected [40]. Washizaki et al. performed an empirical study combining a systematic literature review and a survey to collect, classify, and analyze architectural and design patterns for ML systems [57]. The aim of this study was to bridge the gap between traditional systems and ML systems with respect to architecture and design, and help developers by providing a comprehensive classification of good and bad design patterns for ML systems.

Further work on SE for AI can be found in workshops such as the Workshop on AI Engineering (https://conf.researchr.org/home/icse-2021/wain-2021). However, most publications found in general SE venues do not focus specifically on existing requirements challenges, and we have been unable to locate publications providing a broad practitioner view of NFRs.

RE for AI: While there are many approaches that use ML to improve RE tasks such as model extraction [5], prioritization [46], and categorization [58]—with much of such work reported in the AIRE Workshop Series [23]—there is not as much research looking at RE for ML systems [56]. However, recent publications point out challenges and issues in RE for AI-based systems.

Vogelsang & Borg have pointed out that the development process for ML systems is more complex, with the need to effectively use large quantities of data, as well as a dependence on other quality requirements (NFRs) [56]. Belani et al. identified, discussed, and tackled challenges for requirement engineering disciplines in developing ML and AI-based complex systems [8]. They reported that one of the challenges in ML-enabled software development is to identify

NFRs throughout the software lifecycle, not only in the initial phases dealing with requirements, but as part of the whole lifecycle. ML-based systems demand interventions to SE processes on many levels, including versioning of the ML models, datasets availability, and the whole system's performance [8]. Along with the shortage of expertise, the challenges for managing ML systems are enormous, but less known and generally underestimated as compared to standard challenges [8]. Heyn et al. used three real use cases of distributed deep learning to describe system engineering challenges relating to requirements engineering [33]. They specifically focus on challenges concerning AI context, defining data quality attributes, testing/monitoring/reporting, and human factors.

Nagadivya et al. explored ethical guidelines for the development of transparent and explainable AI systems, defined by various organizations [7]. Here, they considered transparency and explainability as non-functional requirements. The guidelines found that transparency and explainability relate to several quality requirements, such as fairness, trustworthiness, understandability, traceability, auditability, and privacy. Their study suggests a structured way for practitioners to define explainability requirements for AI systems.

Other publications begin to offer solutions, for example, Rahimi et al. introduced a RE-focused method using domain-specific concepts to find dataset gaps for safety-critical ML components [47]. Further research looking at requirements for AI focuses on specific types of requirements, such as transparency (e.g., [26]) or legal requirements (e.g., [11]). A recent workshop (RE4AI@REFSQ) has begun to explore RE for AI, but thus far, no papers have focused on the broad state of NFRs among practitioners. Sproosh et al. employed a case study to evaluate the expressiveness and usefulness of a conceptual framework named GR4ML for requirements elicitation and design of machine learning systems [44]. Their results confirmed that GR4ML can be useful in ML projects by revealing new requirements that would have been missed unless using the framework. The framework also includes a sufficient set of concepts for expressing machine learning requirements and solution design. Anisetti et al. proposed a methodology based on Multi-Armed Bandit for evaluating NFRs of ML models, which represents the foundation for future certification of ML-systems [4]. The authors considered a scenario with the availability of multiple ML models that can be selectively compared in terms of their non-functional properties to prove the applicability of their approach in evaluating the fairness of different ML models. Nakamichi et al. focused on quality characteristics and measurement methods related to functional correctness and maturity of ML software systems (MLS). They extended the quality characteristics of conventional software defined by ISO25010 to those unique to MLS, defining a quality measurement method. They defined a method to identify requirements to derive the quality characteristics and measurement methods for MLS [43].

Hawkins et al. introduced a methodology that covers six stages of ML lifecycle (ML safety assurance scoping, safety requirements elicitation, data

management, model learning, model verification and model deployment) for the assurance of ML for use in autonomous systems (AMLAS). AMLAS comprises a set of safety case patterns and processes for integrating safety assurance into the development of ML components, and for generating the evidence base for justifying the safety of the ML components that integrate into autonomous system applications [32]. Berry presented how measures are used to evaluate an AI and the criteria for acceptable values of these measures [10]. He also showed how AI context information that inform the criteria and trade offs in these measures, collectively constitute a requirements specification of an AI.

Further work has focused on outlining the challenges of NFRs for ML systems, including an outline of research directions [34]. Villamizar et al. conducted a systematic mapping study on requirements engineering for ML and found several contributions in the form of approaches, analysis, quality models, checklists and guidelines, and taxonomies [55]. They identified and discussed gaps by mapping aforementioned contributions and their type of empirical evaluations. They also identified quality requirements relevant for the ML systems. They reported ML related challenges such as lack of validated RE techniques, difficulties in handling customers requirements; and fragmented and incomplete understanding of NFRs for ML. Ali et al. conducted a systematic mapping study to understand, classify and evaluate quality models for AI systems, and found quality models and different quality characteristics applicable for AI systems [1].

In contrast to the research discussed above, we focus on a wider view of NFRs for ML in research and in industry, collecting an overview of NFR perception from practitioners.

Scope of NFRs over ML systems: Sculley et al. focused on hidden technical debt in ML, but also pointed out that ML makes up only a small part of a software system [52]. The ML components may be surrounded by code focusing on configuration, data collection, feature extraction, analysis tools, or monitoring, as well as glue code to make everything work together. Further, as emphasized by Vogelsang & Borg, RE for ML should focus not only on requirements for the system, but on requirements over the data [56]. These consideration raises the question of scope in our investigation. To simplify, in this work we focus on three possible scopes—NFRs over the ML model, NFRs over the whole system (including all the additional surrounding software as described by Sculley et al.), and NFRs for ML-related data, as highlighted by Vogelsang & Borg.

Overall, although previous work has pointed out that handling NFRs in the development of ML systems is difficult, little work focuses specifically on NFRs, or tries to understand the state-of-the-art in defining and measuring NFRs among practitioners.

3 Methodology

To guide our study, we introduce a number of research questions. Our overarching research question is as follows:

RQ0: What is the perception and current treatment of NFRs in ML among practitioners?

By practitioners, we specifically refer to software developers in an industrial or academic role who have experience related to defining, measuring, or assessing NFRs or other software quality concerns during the development of ML systems. We refine this overall question into more detailed questions as follows:

- **RQ1:** Which ML-related NFRs are more or less important in industry?
- **RQ2:** Over what aspects of the system are NFRs defined?
- **RQ3:** What NFR and ML-related challenges are perceived?
- **RQ4:** How are NFRs related to ML currently measured?
- **RQ5:** Over what aspects of the system are NFRs measured?
- **RQ6:** How are NFRs and their measurements captured in practice?
- **RQ7:** What measurement-related challenges for NFRs in ML exist?
- **RQ8:** Is there a difference of perspective for participants working in different contexts: industry, academia or both?

With **RQ1**, we aim to understand if the emphasis on certain NFRs in literature corresponds with interest in reality. **RQ2** is inspired by our scoping question—are NFRs defined over the ML model, the whole system, or the data? **RQ3** aims to identify general challenges in this area.

In **RQ4–7**, we aim to understand whether and how NFRs for ML are measured, over what part of the system they are measured, how these measurements are captured, and what challenges exist in the area of NFR for ML measurement. Finally, in **RQ8**, we aim to understand whether there is different perspective on the above questions between practitioners working in industry, academia, or in a blended role.

To answer these questions, we conducted an interview study followed by a survey with practitioners who are working with ML, RE, and NFRs in industry and academia. With the interview study, we aimed to identify important NFRs, NFR scope, and NFR-related challenges for ML systems. With the survey, we aim to validate and extend the interview results. The detailed process of conducting the interview and survey is described in Sections 3.1 and 3.2.

We have made our interview themes and survey data available at https://doi.org/10.5281/zenodo.6520009

3.1 Interviews

We initially performed a qualitative interview study to answer our research questions and explore experiences and perceptions in the context of NFRs for ML, following the ACM SIGSOFT Empirical Standard [48][1].

[1]This standard can be found at https://github.com/acmsigsoft/EmpiricalStandards/blob/master/docs/QualitativeSurveys.md

Table 1 Interview questions, mapped to the research questions. All 10 interviewees responded to all questions.

Interview Questions	Research Question(s)
Background of Interviewee (Demographic Data)	
1. Please introduce yourself and your role in this company/organization.	N/A
2. Do you consider yourself in more of an academic role or an industrial role?	N/A
3. What is your total number of years of experience in the industry and how long have you been in your current position?	N/A
4. Please describe your responsibilities (e.g., product owner, developer, software architect) in your organization.	N/A
5. Please describe your experience working with NFRs.	N/A
NFR Related Questions	
6. Do you think NFRs play an important role in the success of a software? If yes, how?	RQ1
7. Do you think there are differences in NFRs between traditional software (without ML) and ML-enabled software? If so, what? If not, why not?	RQ1
8. Do you think there are NFRs that are more prominent or important in an ML context? If so, which ones?	RQ1
9. Do you think there are some NFRs that are less important in an ML context that were important for traditional software?	RQ1
10. Do you think of NFRs for the whole system, for the ML model, for the data, or other parts?	RQ2
11. What challenges do you experience with NFRs for ML?	RQ3
NFR Measurement Questions	
12. Do you measure NFRs over ML-enabled software?	RQ4
13. Of the NFRs you mentioned, how do you measure these NFRs in an ML context?	RQ4
14. Are these NFRs measured over the whole system, the ML implementation, the data, or other parts of the system?	RQ2
15. How do you capture NFRs and their measurement for ML-enabled systems?	RQ1.6
16. What are the challenges you face measuring NFRs for ML?	RQ7
17. Do you have anything else you would like to add?	All

Sampling: The goal of the sampling was to find interviewees who had experience with ML, and who were currently working with ML in industry. The sampling method was a mix of convenience, purposive, and snowball sampling. We sent open calls to our colleagues and at industry events to find those with industrial ML experience, then asked interviewees if they knew of further qualified people we could contact. In the end, we interviewed 10 practitioners in different sectors who have experience working with ML in industry. These practitioners often had a mix of industrial and research background. We believe the interviewees we selected are representative of those working in the data science and ML field, including their knowledge (or lack thereof) of NFRs.

Table 2 Demographic information of interviewees, including country, job context (industrial, academic, both), organization domain, role, years of experience, and in the current position, responsibilities, and NFR experience.

ID	Country	Context	Domain	Role	Exp.	Respons.	NFR Exp.
P1	Sweden	Industry	Medical	Section Leader	20 (1.5)	Research on strategic level	NFRs for patient manage. tool
P2	Norway	Industry	Email	RE, Tool Expert	11 (1.5)	Working with RE, tools for RE	Working with RE process, but not require-ments
P3	Canada	Industry	Cloud	Dev. Manager	15 (2)	Leading dev. team	Consider NFRs when creating software
P4	Sweden	Both	ML	Senior Data Scientist	12 (2)	Develop ML, coor-dinate dev. team	Work with NFRs
P5	Sweden	Both	Medical	Chief Data Scientist	10 (3)	Lead team on hospital digital-ization and decision support system	Use, not in very structured way
P6	Sweden	Industry	Sustainable Solutions	Team Manager	15 (2)	Designing software, leading project	Consider NFRs
P7	Sweden	Industry	Biotech	Consultant	3 (1.5)	Product owner, architect, developer	Consider NFRs
P8	Israel	Both	Security	Head of Research	28 (1)	R&D of the model	Consider and discuss NFRs
P9	Sweden	Both	Vehicle	Safety Expert	3 (0.5)	Research, function dev., safety knowledge	Ensure safety NFRs
P10	Canada	Industry	Multi-purpose	R&D Lead	4 (2)	Lead team to make ML explainabile and accountable	Improve ML model NFRs

Participant Demographics: Table 2 shows demographic information on interview participants, including location, the domain of their organization, whether their role is in an industrial or academic context, their role in their organization, their total years of experience and experience in their current position, their responsibilities, and their experience working with NFRs.

Our interviewees cover a wide range of domains, countries, and roles (e.g., Research Leader, Data Scientist, and Team Manager). Six of the 10 interviewees are from Sweden. The interviewees' responsibilities include conducting research, developing and implementing ML, leading development teams, and other roles. The interviewees' experience varies between three to 28 years, but most had more than 12 years of experience. Overall, our interviewees lean towards more senior positions. However, they generally have only a few years experience in their current role.

We observed that the interviewees have a mix of industrial and research backgrounds—it was hard to find interviewees who solely come from an academic or industrial background. This could be a result of our search strategy,

but we also hypothesize that this may be an indication of the novelty of ML systems in industry.

Data Collection: We used semi-structured interviews, with a set of predetermined open-ended questions, so that there remained enough freedom to add follow-up questions to collect in-depth information. The interview guide can be found in Table 1. The interviews lasted 30 to 35 minutes, and were conducted online between December 2020 and February 2021 via Zoom. We recorded all the interview sessions with the permission of interviewees. All interviews were transcribed, and anonymized for further analysis.

The interview started with describing the background of the study and the research gap to help interviewees to gain a sense of their role and make them comfortable with the context of the interview. The questions were divided into three categories. In the first set of questions (Questions 1-5), we collected interviewees' demographic information as well as their experience working with NFRs. In the second set of questions, questions 6 and 7 gather the interviewees' general impressions of NFRs, and if there are differences between NFRs for ML or traditional software. These questions were meant to act as an initial check or filter: if the participants did not believe NFRs were important, or that ML brought specific differences, the rest of the interview may not provide fruitful results. We then asked about NFR-related questions in an ML context, NFR measurements in an ML context, and a final open question so that interviewees could provide more input. For identifying and defining NFRs for ML-enabled software, we relied on the interviewees' own definitions of NFRs, and reported on their perceptions.

In some cases, participants were not explicitly familiar with the term NFR. We believe this is not uncommon for those working with ML in industry. In these cases, we showed them an example hierarchy of NFRs, using McCall's quality hierarchy [17]. They were then able to recognize and talk about software quality aspects. We discuss this as part of our consideration of validity threats in Section 5.2.

Pre-Testing: To improve the validity and reliability of the interview process, we conducted test interviews with two Ph.D. students working with NFRs and ML. This procedure helped to remove ambiguous and redundant questions, revise unclear wording, and rearrange the questions.

Data Analysis: The collected data was qualitative; therefore, we used thematic analysis as a data analysis method [50]. We used a mixed form of coding, where we started with a number of high-level codes based on our RQs, then refined and adapted these codes when going through the transcripts [22]. Two of the authors started to code each transcribed interview separately and afterward reviewed and validated the codes for each interview with each other. We did this for the first five interviews and discussed the results and findings in several iterative meetings, reaching a good level of agreement. Then, the first author coded the remainder of the transcribed interviews. We then combined data from all transcriptions into summary tables and figures, working together to find high-level categories for our codes. We made an effort to maintain

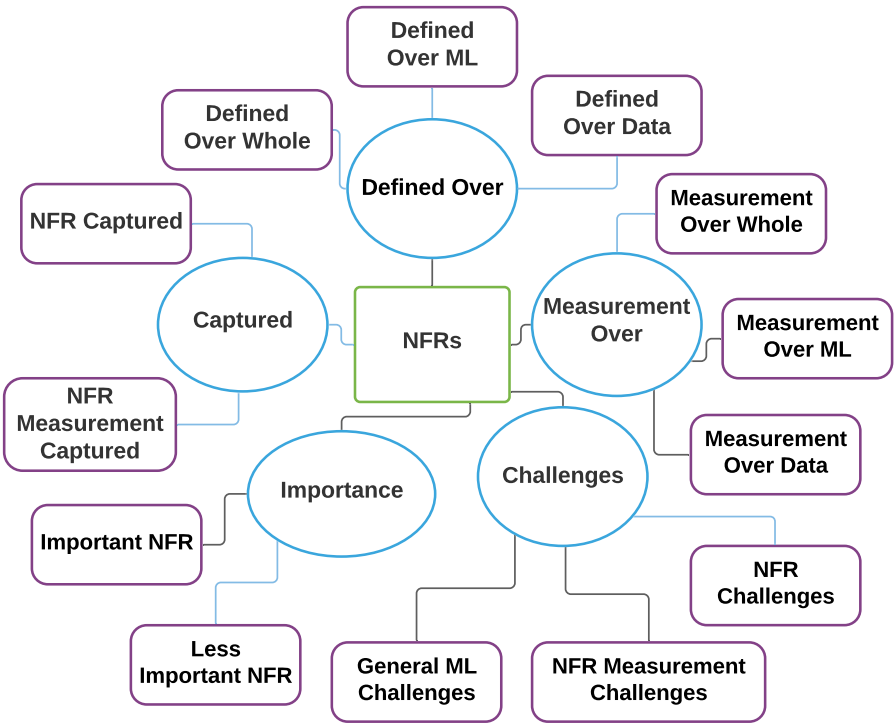

Fig. 1 Overview of Codes/Themes

the original terminology of the participants in developing our codes, e.g., we merged similar NFRs only a few clear cases.

A graphical overview of the thematic codes we derived from the interviews can be found in Fig. 1. Five high-level codes were identified, sub-divided into different codes, e.g., Challenges is a high-level code divided into three sub-codes: NFRChallenges, GeneralMLChallenges and NFRMeasurementChallenges.

The high-level code DefinedOver maps the interview comments that state which part of the system NFRs are defined over, subcategorized into: DefinedOverData—statements where interviewees said NFRs are defined over data—DefinedOverML—where interviewees said NFRs are defined over an ML model— and DefinedOverWhole—where interviewees said NFRs are defined over the whole system. For example:

> "*To be honest, I just see the non-functional requirements just for the machine learning system.*" - P4

We mapped this comment with DefinedOver and DefinedOverML as this statement describes which part of the system the NFRs should be defined over. Similar to DefineOver, MeasurementOver mapped the statements that include comments on which part of the system NFRs are measured and categorized into MeasurementOverData, MeasurementOverML, and MeasurementOverWhole.

The statements on importance of NFRs for ML are coded as ImportantNFR and LessImportantNFR, if an NFR is more or less important to the interviewee. Statements that include the name of specific NFRs for ML are coded with their name, for example, safety, performance, and efficiency are coded as NFRSafety, NFRPerformance, and NFREfficiency. Methods for NFR and measurement capture were coded as NFRCaptured and NFRMeasurementCapture.
As an example, consider this statement:

> "*In terms of explainability, fairness, and other metrics, quality attributes, of course, it's a very important part of making any software as a service better.*" - P10

This statement is coded as ImportantNFR—as the interviewee discusses a number of important NFRs—and with the more specific codes of NFRCorrectnessAccuracy, NFRExplainability, and NFRFairness, capturing the specific NFRs that arose.

3.2 Survey

To validate and extend the interview study results with more participants, we decided to conduct a survey. Our research problem is descriptive in nature since we aim to understand the important NFRs for ML systems, NFR definition and measurement challenges, and the scope of NFRs over different parts of a ML system.

Sampling: We aimed to find people who have experience with ML, and knowledge of requirements engineering. The population of our survey included practitioners in both industrial and academic positions who are working with requirements engineering for ML systems. The sampling method is a mix of purposive and convenience sampling. We sent the online survey to our contacts via email. We also posted the survey links with descriptions in different related groups on LinkedIn, Twitter, and Facebook. The survey link was open from September 22, 2021 to April 7, 2022, and 42 respondents answered at least part of the survey. Up to 30 responses were analyzed, based on those who provided demographic information, and depending on which questions were completed.

Participant Demographics: Table 3 presents demographic information on the 30 examined responses, including their country, whether they describe themselves as industrial, academic, or both, the size of the organization they work for, their role, and their experience in working with machine learning, requirements engineering and non-functional requirements.

The survey participants come from a wide range of countries, contexts, roles, and levels of experience. 15 out of 30 participants are from Sweden. Among the 30 participants, one is a product owner, seven are developers, three are software architects, and five others are in different software engineering and ML positions in their organization. Fourteen participants are researchers. Five work in organizations of less than 50 employees, four in organizations of 50 to 250 employees, and 21 in an organization with more than 250 employees. Among the participants, 13 describe themselves as academics, 10 consider

Table 3 Demographic information of survey participants, including a participant ID, context (academic or industrial practitioner, organization size, role in the organization, and experience in ML, RE, and NFRs (in years).

ID	Country	Context	Org. Size	Role	ML Exp.	RE Exp.	NFR Exp.
I1	Switzerland	Industry	> 250	Software Architect	≤ 1	≥ 3	≥ 3
I2	Brazil	Both	> 250	Software Architect	≤ 1	≥ 3	≥ 3
I3	UK	Both	> 250	Research Software Engineer	≤ 1	≥ 3	≥ 3
I4	Brazil	Both	50–250	Developer	≥ 3	≥ 3	≥ 3
I5	Finland	Academic	> 250	Researcher	1–2	1–2	1–2
I6	Sweden	Both	> 250	Researcher	≥ 3	≥ 3	1–2
I7	USA	Academic	> 250	Researcher	1–2	≥ 3	≥ 3
I8	Sweden	Academic	> 250	Researcher	≥ 3	≤ 1	≤ 1
I9	Sweden	Both	> 250	Researcher	≥ 3	≤ 1	≤ 1
I10	Sweden	Industry	> 250	Developer	≥ 3	≤ 1	≤ 1
I11	Bangladesh	Both	< 50	Researcher	≥ 3	1–2	1–2
I12	Luxembourg	Academic	> 250	Researcher	1–2	≤ 1	≥ 3
I13	Bangladesh	Both	> 250	Developer	1–2	≤ 1	≤ 1
I14	Switzerland	Academic	50–250	Researcher	≥ 3	≤ 1	≤ 1
I15	Germany	Industry	50–250	QA Automation	1–2	≥ 3	≥ 3
I16	Sweden	Both	> 250	Manager	1–2	≥ 3	1–2
I17	Sweden	Academic	> 250	Researcher	≥ 3	-	-
I18	Sweden	Industry	> 250	Developer	≤ 1	1–2	1–2
I19	Sweden	Industry	> 250	Developer	≤ 1	1–2	≥ 3
I20	USA	Academic	< 50	Developer	≤ 1	≤ 1	≤ 1
I21	Sweden	Academic	> 250	Researcher	≥ 3	≤ 1	≤ 1
I22	Sweden	Academic	> 250	Researcher	≤ 1	≥ 3	≥ 3
I23	Denmark	Industry	> 250	Manager	≤ 1	1–2	≥ 3
I24	Sweden	Industry	> 250	Software Architect	≥ 3	1–2	≥ 3
I25	Georgia	Academic	< 50	Researcher	1–2	1–2	1–2
I26	Sweden	Academic	> 250	Researcher	≤ 1	≤ 1	≤ 1
I27	Sweden	Industry	< 50	Developer	≥ 3	1–2	1–2
I28	Sweden	Industry	< 50	Developer	≥ 3	1–2	1–2
I29	Sweden	Academic	> 250	Researcher	≥ 3	≥ 3	≤ 1
I30	Sweden	Industry	50–250	Product Owner	≥ 3	≥ 3	≥ 3

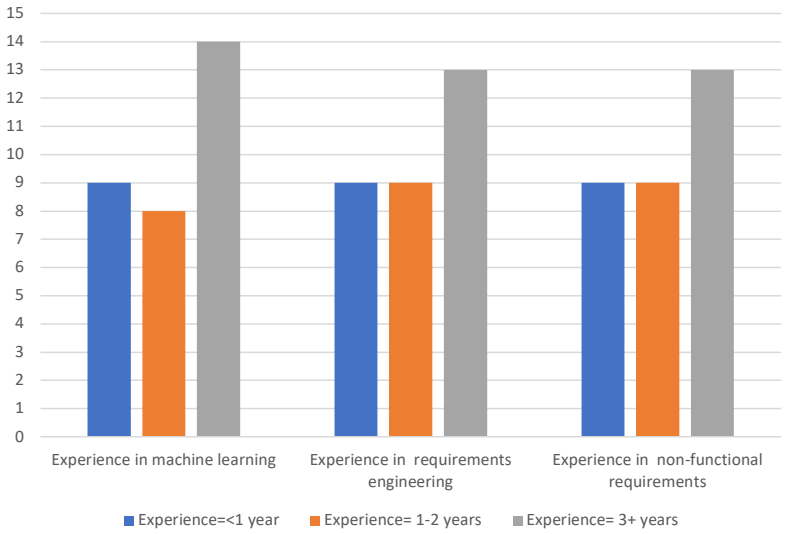

Fig. 2 Machine Learning, requirements engineering, and non-functional requirements related experience of the participants.

themselves as working in industry, and eight consider themselves as working both in academia and industry.

The participants' ML, RE, and NFR experience is presented in Fig. 2. Almost half of the participants (42–45%) have ≥ 3 years experience working with ML, RE, and NFRs. Among the rest, 26-29% have less than a year of experience and 29% have 1–2 years experience in the different areas.

Data Collection: We used partially-structured questionnaires to ensure there remained enough freedom for participants to add their own opinion to collect in-depth information. The survey questions can be found in Table 4. The survey starts by describing the background, research gap, and purpose of the study to help participants to understand the purpose of the study. The survey questions are divided into three categories. In the first set of questions (Questions 1-5), we collect demographic information along with the experience of participants in ML and non-functional requirements. In the second set of questions (Question 6-10), we collected participants' general impressions of NFRs, if the participants think NFRs play an important role in ensuring the quality of ML systems, the degree of importance of each NFR, and the scope on which part of the ML systems NFRs should be defined and measured. We gathered a list of NFRs (38 NFRs) that were considered important NFRs by the interviewees. Based on the similarity and the importance of the NFRs mentioned by the interviewees, we included 25 NFRs in the survey to control its length. We provided a general definition of each NFR to help respondents answer the questions. These definitions are presented in Table 5. In the third set of questions (Questions 11-18), we collected information on NFR challenges, including whether respondents agreed that these challenges could hinder development of ML systems. The respondents did not have to respond every question, and were also given the space to write qualitative comments for most questions.

Pre-Testing: To improve the reliability, validity, and quality of the survey questionnaires, we conducted a test survey with one Ph.D. student, one post-doctoral researcher, and one associate professor. These tests helped us remove redundant questions, revise unclear wording, and rearrange questions based on the suggestions of the participants.

Data Analysis: Although pilot tests produced positive results, many participants filling out the survey did not complete the full set of questions, indicating that our survey may have been too long. In order to utilize the data we collected, we kept the responses for those participants who filled out demographic information, even if the survey was not complete. As a result, we do not have the same number of responses for all questions in the survey. We report the number of answers collected for each question in the last column of Table 4. Each question, beyond the basic demographic questions, has 25–30 answers.

Most of the collected data is quantitative, and we use descriptive statistics to analyze this data. We also collected qualitative data on different questions. However, few participants wrote comments to supplement their answers. Where they existed, we analyze the comments and used them to extend our other qualitative findings.

Table 4 Survey questions, mapped to the research questions, with the type of response and the number of respondents indicated.

Survey Questions	RQ(s)	Type	Num.
Background of participants (Demographic Data)			
1. In which country do you work?	N/A	Text input	38
2. Which statement describes you the best?	N/A	Selection	42
3. What is the size of the organisation you are currently working for?	N/A	Selection	42
4. What is your role in your organisation?	N/A	Selection	42
5.1 How much experience do you have in the following?(Machine Learning)	N/A	Likert scale	42
5.2 How much experience do you have in the following? (Requirement engineering)	N/A	Likert scale	41
5.3 How much experience do you have in the following? (Non-functional requirements)	N/A	Likert scale	39
NFR Related Questions			
6. There is a difference in how NFRs are defined and measured between traditional systems and ML-enabled systems.	RQ1	Likert scale	28
7. NFRs play an important role in ensuring the quality of an ML-enabled system.	RQ1	Likert scale	30
8. Which of the following NFRs are important for ML-enabled software?	RQ1	Likert scale	30
9. Do you define NFRs for the whole system, the ML model, or the data?	RQ2	Selection	29
10. NFR measurements for ML-enabled systems can be dependent on another NFR defined for the other parts of same system, the whole system, the ML model, or the data.	RQ5	Likert scale	26
NFR and ML related challenges			
11. How often do you face challenges defining NFRs for ML-enabled systems?	RQ3	Selection	30
12. Domain dependency of NFRs for ML-enabled systems is a challenge.	RQ3	Likert scale	26
13. Uncertainty is a challenge for identifying, defining and measuring NFRs for ML-enabled software.	RQ3	Likert scale	27
14. Lack of awareness among customers about NFRs for ML-enabled systems is a challenge.	RQ3	Likert scale	26
15. Lack of awareness among engineers about NFRs for ML-enabled systems is a challenge.	RQ3	Likert scale	26
16. Implementing rigorous testing is a challenge for testing NFRs for ML-enabled systems.	All	Likert scale	25
17. Missing measurement baselines is a challenge for measuring NFRs for ML-enabled systems.	RQ7	Likert scale	25
18. NFR measurements for ML-enabled systems are dependent on the context.	RQ5	Likert scale	27

Table 5 Important NFRs for ML systems, as defined in the survey.

NFRs	Definition
Accuracy	The number of correctly predicted data points out of all the data points.
Adaptability	The ability of a system to work well in different but related contexts.
Bias	A phenomenon that occurs when an algorithm produces results that are systematically prejudiced due to erroneous assumptions in the ML process.
Completeness	An indication of the comprehensiveness of available data, as a proportion of the entire data set, to address specific information requirements.
Complexity	When a system or solution has many components, interrelations or interactions, and is difficult to understand.
Consistency	A series of measurements of the same project carried out by different raters using the same method should produce similar results.
Correctness	The output of the system matches the expectations outlined in the requirements, and the system operates without failure.
Domain Adaptation	The ability of a model trained on a source domain to be used in a different—but related—domain.
Efficiency	The ability to accomplish something with minimal time and effort.
Ethics	Concerned with adding or ensuring moral behaviors.
Explainability	The extent to which the internal mechanics of ML-enabled system can be explained in human terms.
Fairness	The ability of a system to operate in a fair and unbiased manner
Fault Tolerance	The ability of a system to continue operating without interruption when one or more of its components fail.
Flexibility	The ability of a system to react to changing demands or conditions.
Integrity	The ability to ensure that data is real, accurate and safeguarded from unauthorised modification.
Interpretability	The extraction of relevant knowledge from a model concerning relationships either contained in data or learned by the model
Interoperability	The ability for two systems to communicate effectively
Justifiability	The ability to be show the output of an ML-enabled system to be right or reasonable.
Maintainability	The ease with which a system or component can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment
Performance	The ability of a system to perform actions within defined time or throughput bounds.
Portability	The ability to transfer a system or element of a system from one environment to another.
Privacy	An algorithm is private if an observer examining the output is not able to determine whether a specific individual's information was used in the computation.
Reliability	The probability of the software performing without failure for a specific number of uses or amount of time.
Repeatability	The variation in measurements taken by a single instrument or person under the same conditions.
Retrainability	The ability to re-run the process that generated the previously selected model on a new training set of data.
Reproducibility	One can repeatedly run your algorithm on certain datasets and obtain the same (or similar) results.
Reusability	The ability of reusing the whole or the greater part of the system component for similar but different purpose.
Safety	The absence of failures or conditions that render a system dangerous
Scalability	The ability to increase or decrease the capacity of the system in response to changing demands.
Security	Security measures ensure a system's safety against espionage or sabotage.
Testability	The ability of the system to support testing by offering relevant information or ensuring the visibility of failures.
Transparency	The extent to which a human user can infer why the system made a particular decision or produced a particular externally-visible behaviour.
Traceability	The ability to trace work items across the development lifecycle.
Trust	A trusted system is a system that is relied upon to a specified extent to enforce a specified security, or a security policy.
Usability	How effectively users can learn and use a system.

4 Results

In this section, we provide our findings in order to answer our RQs. Section 4.1 focuses on general NFR results (**RQ1**–**3**), Section 4.2 focuses on results relating to measuring NFRs (**RQ4**–**7**), and Section 4.3 compares results between industrial and academic practitioners (**RQ8**).

4.1 NFR Importance, Scope, and Challenges

In this section, we provide our general findings on NFRs for ML, addressing importance (RQ1), scope (RQ2), and challenges (RQ3).

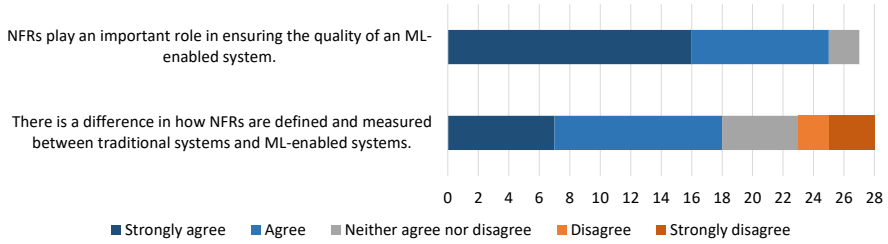

Fig. 3 Perceived NFRs importance and the difference in how NFRs are defined and measured between traditional systems and ML systems.

4.1.1 Perceived NFR Importance (RQ1)

As a baseline question to gauge interest in NFRs, we asked both interviewees and survey participants about the perceived importance of NFRs. All interviewees indicated that NFRs play an important role in the successful delivery of software, and that there are differences between ML systems and traditional systems with respect to NFRs.

In response to the statement "NFRs play an important role in ensuring the quality of an ML system" (illustrated in Fig. 3), 25 survey participants agreed or strongly agreed (93%), while only two participants remained neutral (7%). Similarly, all of the interviewees said they think NFRs play an important role in the success and ensuring the quality of ML systems. While commenting on this statement, survey respondent I14 stated:

> "*They are essential for real-time systems, but it is true for both ML and non ML software.*" - I14

We asked the survey participants whether they agreed that there is a difference in how NFRs are defined and measured between traditional systems and ML-enabled systems (Fig. 3). Most of the respondents (64%) agreed that there is a difference between traditional systems and ML-enabled systems when defining and measuring NFRs. Five participants disagreed (18%), while five more remained neutral (18%).
While providing opinions in the survey, respondent I1 commented:

> "*Most important in industry are acceptance criteria (a set of specific requirements) to derive test scenarios. Testing for ML is different than testing of classic software systems. Therefore, you need different acceptance requirements.*" - I1

I7 said that NFRs for ML systems need to be established for the learning procedure, not just the system that employs the model:

> "*I would expect that NFRs for ML would have additional considerations for the learning procedure in addition to the underlying system itself.*" - I7

Another participant had the view that NFRs are only related to performance, regardless of the system being traditional or ML-related:

> "*NFR for me is hard-core testing of specific performance requirements.*" - I24

Interviewees provided insight on how some NFRs can have a different meaning for ML systems than in traditional software. For example, adaptability or maintainability:

> "*The machine learning models are required to adapt a lot, but a lot of that is done using the data. If you work on a data set and after a month and add a new dataset, you do not have to change a single line of code. But, if I want to add a new feature to the data set, some adaptability is needed. At first, we need to know why, then have to use it. So, adaptability has a different meaning with machine learning.*" - P8

> "*Maintainability would also be different, again, if you throw away your model after getting a better one, don't need them to maintain, but you do need to maintain the pipeline that you generate. So, it's a bit different.*" - P8

Interviewees also indicated that some NFRs should always be taken into account while developing any system. As an example:

> "*Perhaps ethics not that important cause ethics is so in the center already. People use to think ethically, so it is not such a big issue compared to others, like when we work with machines where we do not think about ethics very much. For us, it is very natural to think about ethics.*" - P1

RQ1 (NFR Importance), Finding 1: Most participants agreed that NFRs are important in ensuring ML system quality, and that there are differences in how NFRs are defined and measured between traditional and ML systems (e.g., for adaptability or maintainability).

According to the interview codes, we identified important and less important NFRs for ML, categorizing these into the categories provided by Cavano and McCall [17]: product operation, revision, and transition. Fig. 4 shows the codes related to important and less important NFRs, including counts of the number of the interviewees whose interview included the code (c), and a count of occurrences of the code across all transcripts—the frequency (f). We include the numbers to give an idea of frequency and ranking. However, given the small sample size, this ranking could change with more participants.

All ten interviewees brought up important NFRs for ML systems, and nine named less important NFRs. We observed that interviewees could identify important NFRs for ML quickly compared to less important NFRs.
While talking about important NFRs for ML, P3 named a number of NFRs:

> "*Repeatability, accuracy, these things are often important in ML or deep learning-based software which is not generally that much present in traditional software.*" - P3

Concerning new NFRs, P4 stated:

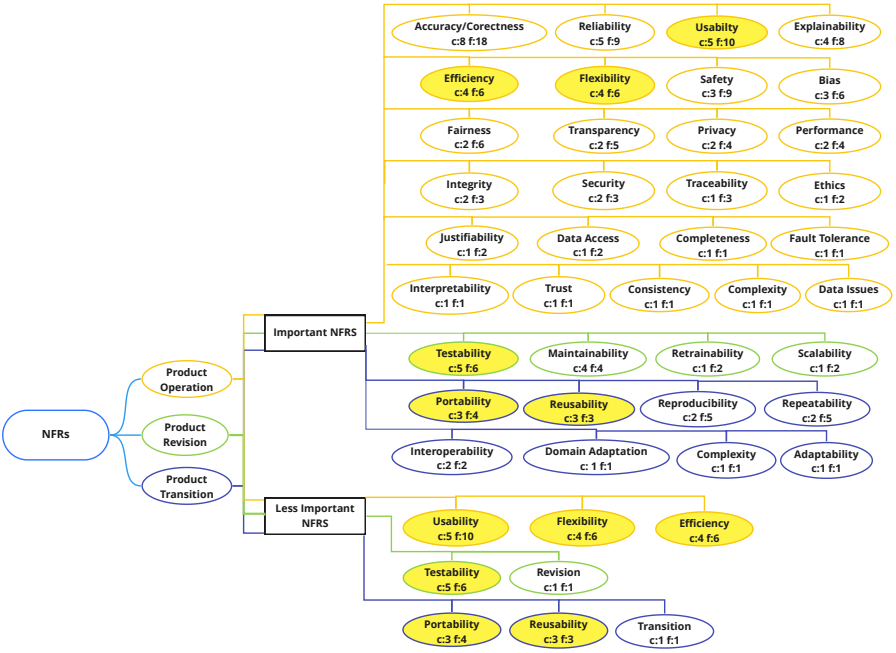

Fig. 4 Important and less important NFRs for ML. **c** is the number of interviewees who discussed the NFR, **f** is the occurrences of the code across all transcripts, a **yellow background** indicates NFRs with split opinion (important to some, less important to others).

> "*Retrainability is a new non-functional requirement for the Machine Learning system. When to retrain, how to retrain, which data use to retrain those are the requirements those you don't define in traditional software.*" - P4

Several discussed less important NFRs:

> "*Flexibility right now is not so important. If you need to scale up, you can do some changing, so we don't consider that as much important thing yet. The same with reusability. I think as AI is not so much mature yet, so we are not considering it yet.*" - P1

> "*The usability is more related to the front end part. Machine learning is a more background component. If you need to be effective in machine learning, you want to collect the right information where the human is in the loop; it is not so important like traditional software.*" - P6

The results illustrate that most NFRs are still considered important in an ML context, and few NFRs are considered less applicable. It is also important to note that there was a disagreement among the interviewees on which NFRs are less important. A few NFRs mentioned by some interviewees as less important are identified as important by other interviewees (colored yellow in Fig. 4). Most of the interviewees could provide answers to the related interview questions. However, not everyone could answer this question easily, and

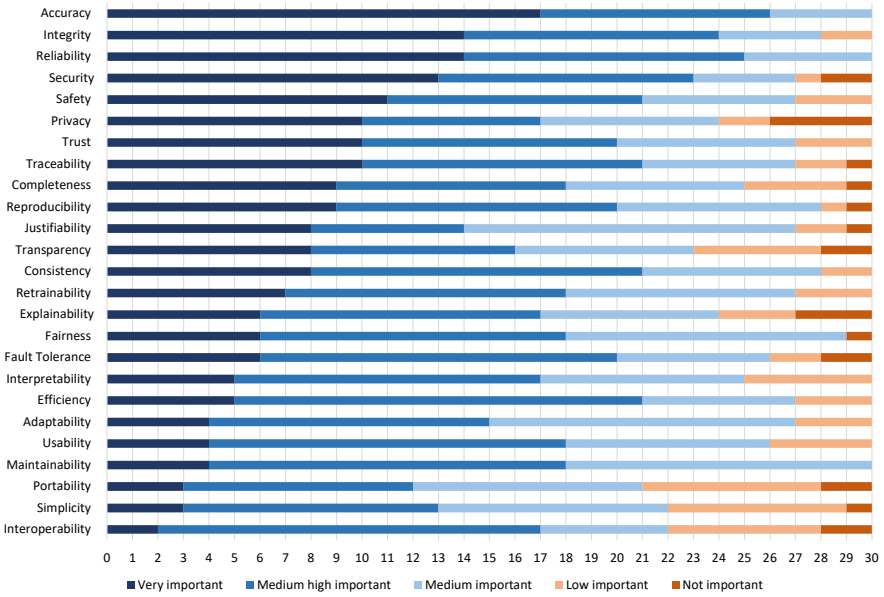

Fig. 5 Importance Level of NFRs for ML systems.

they had to be shown a standard NFR hierarchy (McCall's) [17] to illustrate possible NFRs.

We provided a list of selected NFRs to the survey participants and asked them to rank their importance from "not important" to "very important" for ML systems. The responses of the participants are presented in Fig. 5. Based on the results, many participants reported accuracy, reliability, integrity, and security as the most important NFRs for ML systems. In particular, accuracy and reliability were always indicated as having, at least, medium importance—and generally very high importance.

Very few survey participants reported interoperability, portability, and simplicity as being very important NFRs for ML systems. No respondents reported accuracy, reliability, integrity, safety, usability, efficiency, interpretability, trust, consistency, maintainability, retrainability, or adaptability as having no importance for ML systems.

Privacy was listed as not important by four participants, explainability by three, and others by one or two participants. Portability had the most split in opinion (12 high/medium high important vs nine low/not important). One possibility is that the meaning of this NFR in an ML context is not clear. Simplicity also had a split in opinion (13 high/medium important vs eight low/not important). Maintainability was always listed as having, at least, medium importance. However, it was also not widely considered to be very important either. Fairness is similar—most rated it at medium to medium-high importance, with only one low vote and few very important votes.

We generally found similar results while comparing the survey and interview results in terms of the importance of NFRs. Accuracy and reliability are considered important by both interviewees and survey participants. However, integrity and security were given more importance by survey participants than by the interviewees.

The interviewees had split opinions on three NFRs also asked about in the survey—usability, efficiency, and portability. This split can be observed in the survey results in two ways. In the case of usability and efficiency, we see that few respondents ranked them as very important. However, many ranked both as medium-high importance. Both have some importance, but are not considered among the most important. Both also have a few low votes, but no "not important" votes. With regard to portability, we also see a split among survey respondents. Three respondents note it as very important and nine as medium-high importance, while seven rate it as low and two as not important.

RQ1 (NFR Importance), Finding 2: Accuracy, reliability, integrity, and security are particularly important NFRs for ML systems. Most NFRs defined for traditional software are still relevant in an ML context, while only a few become less prominent (revision, transition).

RQ1 (NFR Importance), Finding 3: Perceptions of efficiency, fairness, flexibility, portability, reusability, testability, and usability are split among participants, with some votes for high importance and other for low.

4.1.2 Scope of NFRs (RQ2)

In this section, we describe the scope of NFR definitions over parts of ML systems. We summarize the answers and codes regarding what part of the system NFRs are defined over—the ML model, the data, or the whole system. Out of ten interviewees, eight said NFRs are defined over the ML model. As an example:

> "*To be honest, I just see the non-functional requirements just for the machine learning [part of the] system*" - P4

Two interviewees said NFRs are defined over the data (testing and/or training data), while four participants said NFRs are defined over the whole software. As an example, P7 mentioned:

> "*Machine learning projects also software projects. So, I guess they all match all over.*" - P7

NFRs for ML are mostly defined over the ML model or the system as a whole. However, we see some disagreement, and note that this question was not easy to answer for many participants.

We also asked the survey participants "Do you define NFRs for the whole system, the ML model, or the data?" In total, 21 participants voted for the whole system (72%), five for the ML model (17%), and three (11%) for data. As an example, I7 commented NFRs should be defined over all parts:

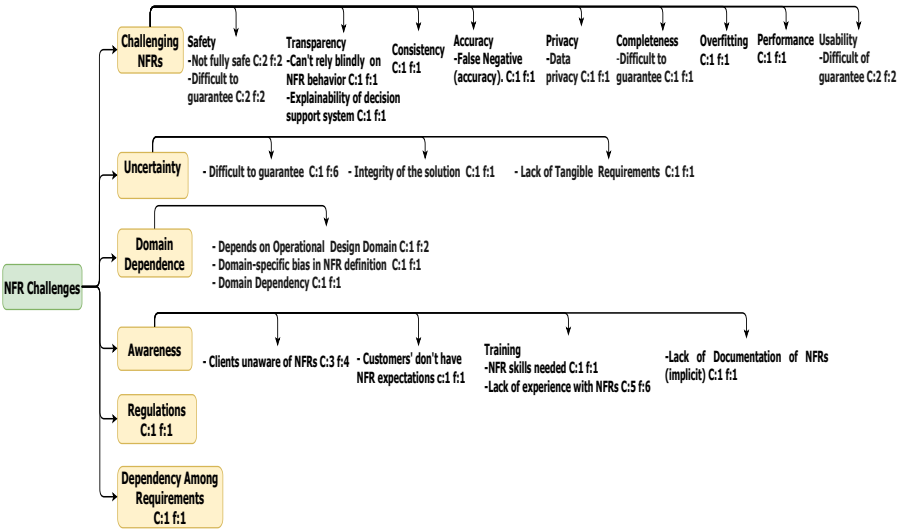

Fig. 6 NFR-Related Challenges with ML Systems.

> "*I'd say you write NFRs for all parts, not just the system as a whole.*" - I7

Another participant favored models over data:

> "*Makes sense for model, hard to do for data.*" - I14

While 80% of the interviewees said that they define NFRs over the ML model, only 17% of survey participants did the same. We found similar results for the definition of NFRs over data as very few participants stated that they define NFRs over the data.

RQ2 (NFR Scope): Most practitioners focused on defining NFRs over the whole system. Many interviewees, and some survey respondents, also define NFRs on models. Few practitioners have explicitly considered NFRs for ML-related data.

4.1.3 NFR and ML-related Challenges (RQ3)

All ten interviewees identified NFR-related challenges. These challenges are presented in Fig. 6. Leaf-level challenges include interviewee counts (c) and frequencies (f). These challenges often related to uncertainty (difficulty in guaranteeing consistent behavior), domain dependence (dependency on a product domain in NFR definition), awareness (lack of awareness about NFRs for ML among customers and practitioners), and regulations (rules and practices imposed by organization, domain, or government). The interviewees also brought up specific NFRs as challenges.

For example, interviewees discussed safety. Some ML applications exhibit non-deterministic behavior. This can make it difficult to demonstrate safety and can hinder satisfaction of safety NFRs:

> "*I think that's very tricky; you can really mess with safety. I think that's why companies always afraid of using machine learning techniques over traditional system where you can really check that.*" - P2

Interviewees pointed out that transparency can be crucial for sensitive ML applications—such as decision support systems—and that—while measuring accuracy—false negatives can be dangerous. According to the interviewees, maintaining privacy and consistency of the system can also be challenging.

We found several challenges related to uncertainty. For ML systems, it is challenging to guarantee deterministic behavior, preserve the integrity of solutions, and there may be requirements that cannot be assessed quantitatively.

Other challenges relate to domain dependence of NFRs. Some NFRs for ML depend on, for example, an Operational Design Domain (ODD)—the specific operating domain(s) in which an automated function or system is designed to properly operate—and there can be domain-specific bias in NFR definition.

The responses also pointed out a lack of awareness of NFRs. Clients are often unaware of NFRs. Therefore, they do not have expectations regarding them. To define NFRs, special skills are needed, and the engineers and researchers lack skills in this regard. P8 said:

> "*Then I think that we don't have enough experience in the field (NFRs for ML) to define them well.*" - P8

The interviewees also mentioned a lack of proper documentation of NFRs, which made it more challenging to define them for ML. Finally, at least one interviewee reported that regulations and laws constrain definition NFRs in ML systems, and that this can be challenging.

Survey participants were asked how often they face challenges defining NFRs for ML-enabled systems. The results are presented in Fig. 7. Among the participants who answered the question, seven (24%) answered they never face challenges in defining NFRs for ML-enabled systems. The remaining 76% encountered challenges in at least some of the projects—two (7%) in 80%–100% of projects, eight (28%) in 50%–80%, and twelve (41%) in a small percentage of projects (<50%). These results indicate that challenges exist, but they are either not completely pervasive, or that current challenges are not clearly classified as being NFR-related.

Specific NFR and measurement-related challenges are presented in Fig. 8. For each, we asked survey participants for their opinion on the challenge listed. These challenges were derived from the interviews above.

Sixteen participants (62%) agreed that lack of awareness among engineers is a challenge, while four (15%) disagreed. One participant stated:

> "*Engineers care for function (unluckily) not for quality - although quality is always mentioned as important.*" - I1

Lack of awareness among customers about NFRs is also a challenge—20 participants agreed (77%), while two disagreed (8%). To reduce lack of awareness, one participant suggested that efforts be made to educate customers:

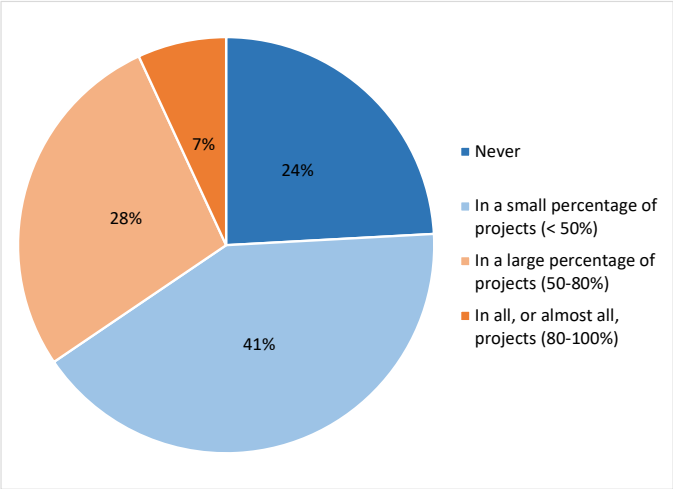

Fig. 7 How often survey participants face challenges defining NFRs for ML systems.

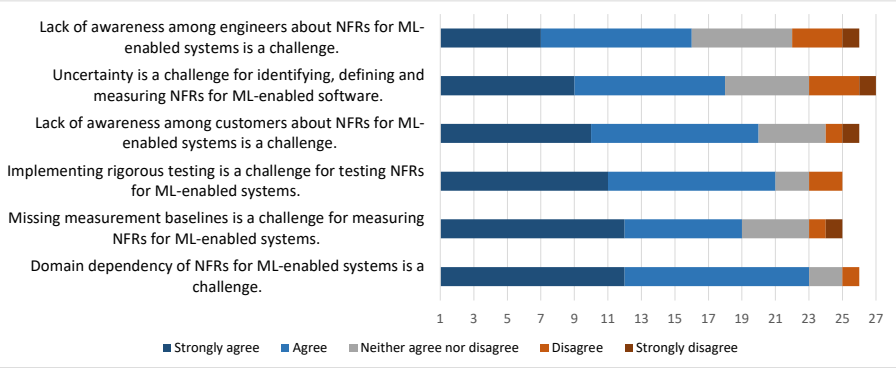

Fig. 8 NFR and NFR-measurements related challenges.

> "*This has to do with a mental process and only education can handle this.*" - I14

Similarly, we could confirm challenges found in the interviews related to uncertainty of defining and measuring NFRs for ML systems, domain dependency of NFRs for ML systems, and implementing rigorous testing of NFRs for ML systems. Most of the participants agreed on these statements, while very few disagreed.

Regarding uncertainty, participants stated:

> "*Lack of common ground and terminology about uncertainty is the major source of uncertainty of people that deal with such systems.*" - I14

> "*We are dealing with complex system. As a non expert in AI and ML and do not expect that results of an ML system are repeatable, i.e. a defined degree of uncertainty is a system property.*" - I1

Regarding domain dependency, participants added:

> "*Domain dependency is a challenge because retraining, adaptation, and lack nondeterminism are still major issues of ML-enabled systems.*" - I14

> "*Whether ML delivers valuable results is strongly dependent on the problem to solve. As many problems are domain specific, I argue that NFRs for ML enabled systems are domain specific as well.*" - I1

> "*I would say a safety-critical domain requires a different worldview than a non-safety-critical system.*" - I7

Regarding rigorous testing, one participant was hopeful—but also noted a relation to domain dependency:

> "*It has challenges, but it can be done, even though it will be domain-dependent.*" - I14

RQ3 (NFR Challenges), Finding 1: NFR challenges relate to uncertainty, domain dependence, awareness, regulations, dependency among requirements, and specific NFRs (e.g., safety, transparency, and completeness).

RQ3 (NFR Challenges), Finding 2: Specific challenges may not emerge in all projects. However, 76% of survey respondents have encountered at least one of these challenges in their ML projects.

When asked about NFR-related challenges, some interviewees answered with both NFR-related challenges and more general challenges regarding ML. Eight interviewees described challenges not specifically related to NFRs. For example, incorrect training and testing data selection, complexity in data pre-processing, unexpected results over time, uncertain system behaviour, an expensive and time consuming testing process, and an unstructured development process.

Fig. 9 reports general ML challenges, related to training, runtime, testing, the development process, and others. Training-related challenges include training data selection, data pre-processing, incomplete or incorrect identification of training and test data, and usage of the same data set for both training and testing. Runtime challenges involve unexpected behavior and systems changes over time, and that deterministic execution is not guaranteed. According to P2:

> "*Actually the system will change something at run time*" - P2

Testing challenges involve complex, expensive, and time-consuming testing. P2 mentioned:

> "*It's getting more and more complex, so the testing needs to become more and more complex.*" - P2

Finally, the interviewees agreed that—in most of cases—the development process of ML-enabled systems is not well structured and well defined:

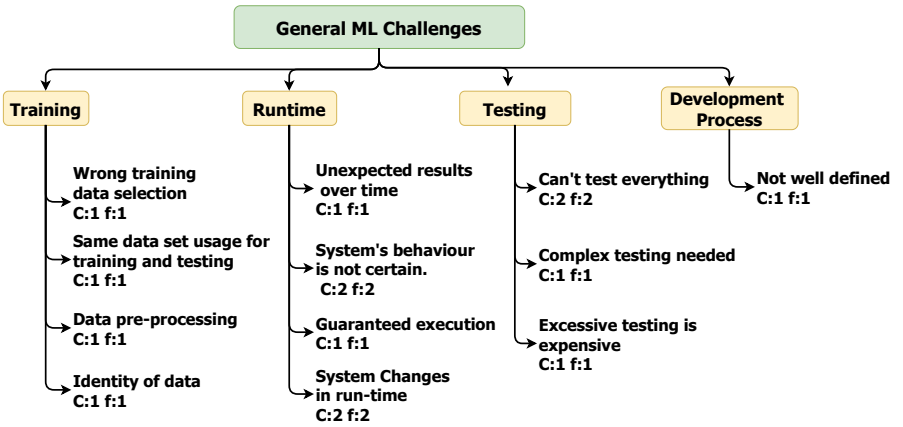

Fig. 9 General ML-related Challenges.

> "*I will say that in the case of Machine Learning, sometimes the development process is not that really well defined*" - P4

RQ3 (NFR Challenges), Finding 3: Interviewees presented ML system development challenges—not specifically related to NFRs—related to training, runtime, testing, and the development process.

4.2 NFR Measurement Scope, Capture, and Challenges

We collected NFR measurement-related information in the third part of the interview and survey, and present our findings in this section. Some points were covered only in the interview and omitted from the survey to reduce the length of the survey.

4.2.1 NFR Measurements (RQ4)

While answering the question "Do you measure NFRs over ML-enabled software?", all interviewees answered that they measure or need to measure NFRs over ML system.

Answers to the question "Of the NFRs you mentioned, how do you measure these NFRs in an ML context?" varied depending on the functionalities the software provides. For example, NFRs can be measured based on response time, statistical analysis, different performance metrics, or user feedback. According to P10:

> "*Lots of NFRs (e.g., accuracy, repeatability, consistency of execution, etc.) are quantifiable, and those quantifiable NFRs can be measured by statistical analysis. For example, accuracy can be measured by accuracy matrix-like f1 score, root mean square error, etc.*" - P10

Measurement should be conducted using a combination of machine and human judgment, along with statistical analysis, in safety-critical situations:

> "*If you set up a clinical trial of something, then you compare with or without machine or with a doctor's judgement with machine, then compare those and in the end if you do statistical analysis to see whether it is significant difference.*" - P1

Usability can be measured using interview results:

> "*We do perform interviews and use the result of the system and see how they find the usability.*" - P5

Additionally, usability can be measured using ad-hoc methods, with some difficulty. P4 said:

> "*The usability of machine learning system is a bit tricky to measure, and sometimes you have to come up with this ad hoc matrix to know about how usable the system is.*" - P4

Further NFRs were also identified as challenging to measure because they may be subjective and not quantifiable. For example, according to P10:

> "*Measurements should be done according to standard baseline, but some measurements are not quantifiable (e.g., usability, adaptability, flexibility, etc.), therefore tricky.*" - P10

RQ4 (NFR Measurement): While some NFRs (e.g., accuracy) can be measured using ML-specific or standard measures (e.g., precision, recall, f1 score), many are difficult to measure (e.g, fairness, explainability)—as with traditional software—because they are not easily quantifiable. In safety-critical situations, both human and machine judgement should be used.

4.2.2 NFR Measurement Scope (RQ5)

We summarize our results concerning what parts of the system NFRs are measured over—the data, the ML model, or the whole system. Six interview participants said NFRs were measured over the ML model, while four interviewees indicated measurements over the whole system. P1 explained:

> "*Before you bring the system into production, you need to measure NFRs for the whole system.*" - P1

Three interviewees said NFRs for ML are measured over data:

> "*Measurement is for the data. If you have labeled data for all cases, then you can measure the performance*" - P8

Generally, looking at the interview results, we see even more disagreement on the scope of measurement than on the scope of NFR definition, with still a slight preference for measuring over the model rather than the whole system or the data.

We asked the opinion from the survey participants on the statement "NFR measurements for ML-enabled systems are dependent on the context", where a context represents a specific scenario, surrounding circumstances, event, or environment (see Fig. 10). Almost all respondents (93%), except for two, agreed

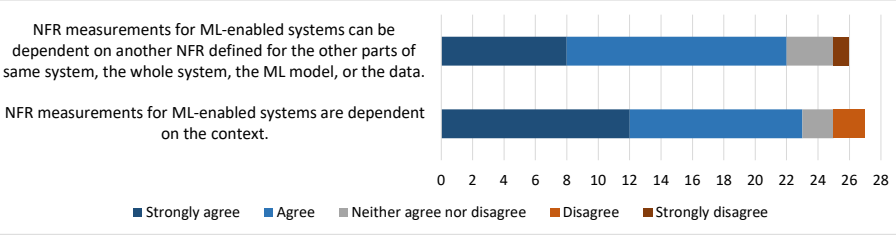

Fig. 10 Questions related to the measurement scope of NFRs.

with the statement. One participant added that measurement for NFRs in ML is dependent on the domain:

> "*As on domains it is dependent on context as well*" - I1

We then asked the participants for their opinion on the statement "NFR measurements for ML-enabled systems can be dependent on another NFR defined for the other parts of same system, the whole system, the ML model, or the data" (also shown in Fig. 10). We received 26 responses, among them 22 participants agreed with the statement (85%), while one disagreed (4%) and three gave neutral responses (12%). One participant suggested correlation analysis to find out the dependencies:

> "*Dependencies are possible, correlation analysis can help reveal them.*" - I14

Another participant commented about the uncertainty of the dependency:

> "*We are in a complex system, variables are strongly dependent in a way we do not know.*" - I1

RQ5 (NFR Measurement Scope), Finding 1: As with definitions, there is variance in the scope of NFR measurements for ML systems. Interviewees expressed a preference towards measurements over the model, while survey participants indicated the whole system.

RQ5 (NFR Measurement Scope), Finding 2: NFR measurement for ML systems depends on context, and measurement can be dependent on another NFR defined for other parts of system, the whole system, the ML model, or the data.

4.2.3 NFR Measurement Capture (RQ6)

We asked the interviewees how NFR measurements for ML systems were captured, e.g., in a tool, or via documentation. Many interviewees had difficulties answering this question—we discuss this further in Section 5.2. Some answered in terms of process. One respondent captures NFRs via interviews, while another mentioned use of checklists. P8 said:

> "*I saw plenty of systems, and we still don't have a good enough methodology for that. Like, these are some checklists that you should go and do*" - P8

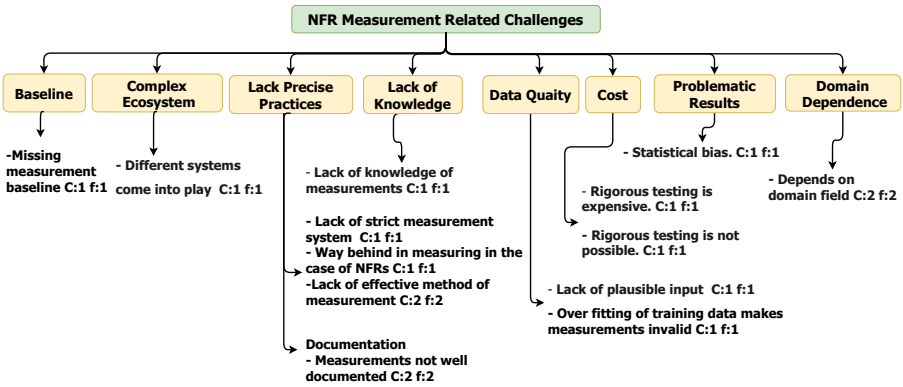

Fig. 11 NFR measurement challenges

For technical means to capture measurements, engineers use different methods. For example, they implement algorithms that run and measure the result against time:

> "*I think for this model, we should develop specific code. But we did not do it. My idea is that we have to write specific software to measure*" - P6

One participant mentioned traceability tooling as a way to measure the fulfillment of NFRs:

> "*Well, normally we have one requirement tracing tool. So, if we have certain non-functional safety requirements, we define tests to prove that we fulfill this non-functional requirement*" - P9

In general, NFR measurement capture depends on the context:

> "*The measurement depends on their functionalities, some are time-based, and some are based on output. Sometimes measurement is captured using different tools and compared with journals in the field of healthcare*" - P1

RQ6 (NFR Measurement Capture): Interviewees capture NFR measurements using checklists, interviews, scripting, and traceability tools. Context is important. Multiple participants found this question difficult to answer.

4.2.4 NFR Measurement Challenges (RQ7)

Fig. 11 summarizes NFR measurement-related challenges found via the interview coding process. Although many challenges could apply to both NFR definition and measurement—e.g., domain dependence—the purpose is different. Here we discuss challenges that specifically arise while measuring attainment of the NFRs.

The first challenge concerns a missing measurement baseline and lack of a strict and effective measurement system. According to P6:

> "*The main challenge is to find an effective way to measures it.*" - P6

While measuring NFRs, different systems come into play. The engineers lack knowledge of measurements, and they are behind in their knowledge of how to measure attainment of NFRs:

> "*If you compare the functional requirements, we are probably way behind when it comes to non-functional requirements. We do not have the same strict system for that as we do in the functional requirements.*" - P5

> "*Not that I am aware of. I mean, testing the system of course, and based on those tests, we decide whether it is safe or not, but in Machine Learning, I am not aware of any possible measure yet.*" - P9

We asked the survey participants their opinion on the statement "Missing measurement baselines is a challenge for measuring NFRs for ML-enabled systems". According to the results (see Fig. 8), 12 participants strongly agreed (48%) and seven participants agreed on the statement (28%). One participant disagreed (4%), one strongly disagreed (4%), and four (16%) remained neutral. One participant commented:

> "*Many datasets are available, but the accuracy on some test sets does not guarantee anything about the performance of the model in operation when exposed to real-world inputs that may differ substantially from those observed in the field.*" - I14

Lack of proper documentation on NFR measurements in the context of ML creates further challenges:

> "*Sometimes it is a lack of documents that contains non-functional requirements compatible with ML-enabled systems.*" - P4

Further challenges include domain-dependency of NFR measurements and statistical bias. ML systems depend on having plausible input—which can be difficult to find—and overfitting of training data makes measurements invalid. Participants also complained about the cost and plausibility of rigorous testing. Furthermore, the ML model may exhibit non-deterministic behavior during runtime, making the measurement process difficult:

> "*Yes, a challenges is that machine learning will not behave in the same way. So, I do not know how you want to measure that. If you want to test on run time, so the time or if you want to keep a complete log out of how the system behaves to understand their problems. This is really tricky, I think. Because usually, implementation will not behave the same in the same situation. Whereas machine learning could behave differently depending on how it trained and how it perceives, how it interprets sensor information as well, all these aspects make it really difficult.*" - P2

We asked the survey participants about their opinion on the statement "Uncertainty is a challenge for identifying, defining and measuring NFRs for ML-enabled software." Eighteen participants agreed (66%), while four (15%) disagreed, and five remained neutral (19%).

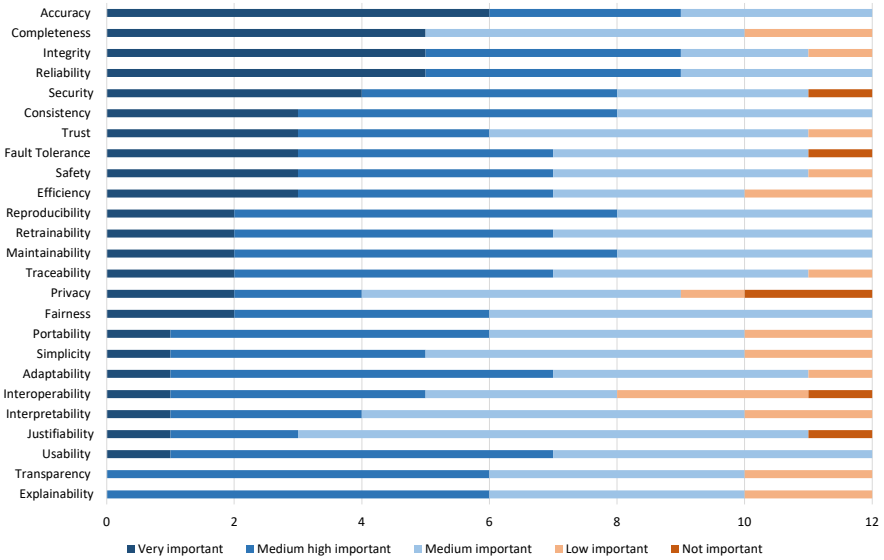

Fig. 12 The importance of NFRs, as identified by participants in **academic positions**.

RQ7 (NFR Measurement Challenges): NFR measurement challenges include a lack of knowledge or practices, missing measurement baselines, a complex ecosystem, data quality, cost of testing, bias in results, and domain dependence.

4.3 Differences Between Industry and Academia (RQ8)

In this section, we describe differences in how practitioners working in different contexts (academic, industry, or in both concurrently) perceive NFRs for ML systems. We examine perspective differences between the three contexts for each applicable research question.

4.3.1 Differences in Perceived NFR Importance (RQ1)

Participants from different contexts differed in their ranking of the importance of NFRs. We show the full results for each role in Fig. 12–14. In addition, in Table 6, we list the average importance for each group, where importance is scaled from 1–5 ("Not Important" to "Very Important"). We indicate the overall median, average, and standard deviation for each group at the bottom of this table. We discuss potential interpretations of these results in Section 5.

In general, practitioners from a blended context assigned the most importance to NFRs, with a median importance of 4.00 (approximately "medium-high"). However, they also had the most variation between NFRs, as shown by the high standard deviation. Industrial practitioners fell in between, with a median importance of 3.78 (between medium and medium-high) and a standard deviation between academic and mixed contexts. Academics were the

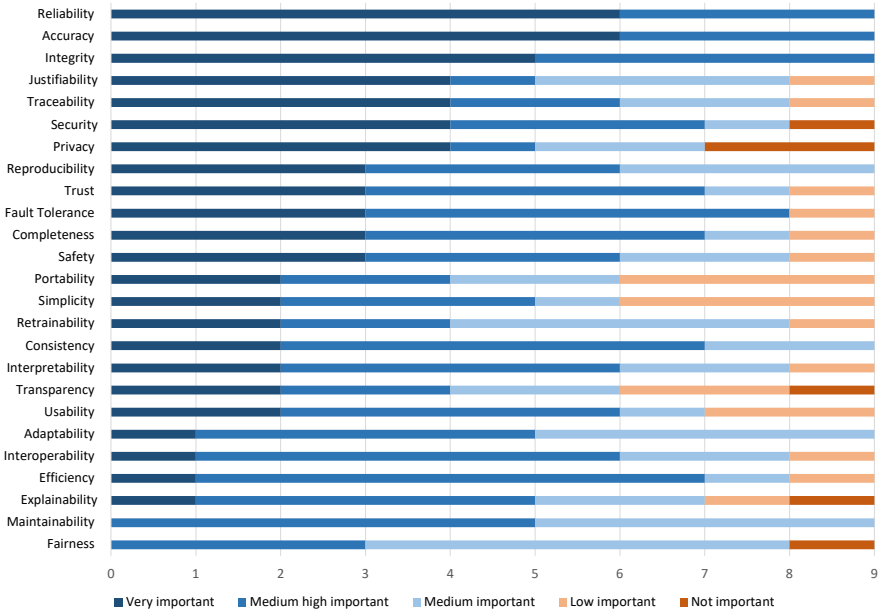

Fig. 13 The importance of NFRs, as identified by participants in **industrial positions**.

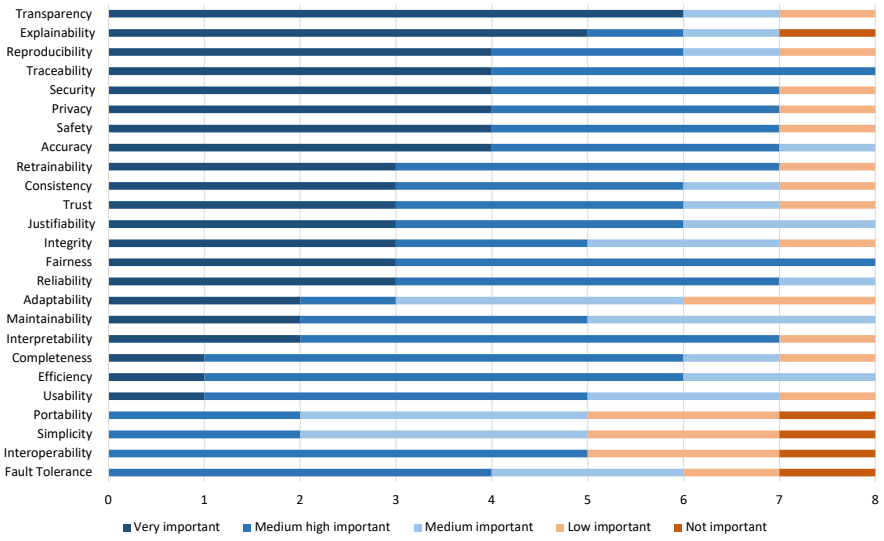

Fig. 14 The importance of NFRs, as identified by participants in **combined academic and industrial positions**.

Table 6 Average opinion on each NFR for each context, where 5 = "Very Important" and 1 = "Not Important". Sorted by average across the three contexts.

NFR	Academic	Industrial	Both
Reliability	4.17	4.67	4.38
Accuracy	4.25	4.67	4.13
Integrity	4.08	4.56	3.88
Traceability	3.67	4.00	4.50
Security	3.83	4.00	4.25
Consistency	4.00	4.00	4.00
Reproducability	3.83	4.00	4.13
Safety	3.75	3.89	4.25
Trust	3.67	4.00	4.00
Completeness	3.67	4.00	3.75
Interpretability	3.25	3.78	4.00
Retrainability	3.67	3.56	4.13
Efficiency	3.67	3.78	3.88
Justifiability	3.17	3.89	4.13
Fairness	3.67	3.11	4.38
Usability	3.67	3.67	3.63
Transparency	3.33	3.22	4.38
Maintainability	3.83	3.56	3.50
Privacy	3.08	3.56	4.25
Explainability	3.33	3.33	4.13
Fault Tolerance	3.67	4.11	2.88
Adaptability	3.58	3.67	3.38
Interoperability	3.08	3.67	3.13
Portability	3.42	3.33	2.75
Simplicity	3.33	3.33	2.75
Median	3.67	3.78	4.00
Average	3.63	3.81	3.86
Std. Dev.	0.32	0.41	0.52

most consistent group, but also assigned more low and not important scores than the other contexts.

Industrial participants placed a higher level of importance on reliability, accuracy, and integrity than other contexts. These NFRs also ranked highly among academic participants. However, these three NFRs are noteworthy as industrial participants ranked them as—at least—medium-high, while those from academic or mixed contexts included lower ratings.

Comparing academic and industrial participants, academic participants placed higher importance—in particular—on fairness, maintainability, and transparency. Industrial practitioners were split on the topic of transparency, with four participants rating it as very important or medium-high, and three rating it as low or not important. Industrial participants placed disproportionately higher importance than academics on justifiability, interoperability, and interpretability. Academic participants were split on the importance of privacy.

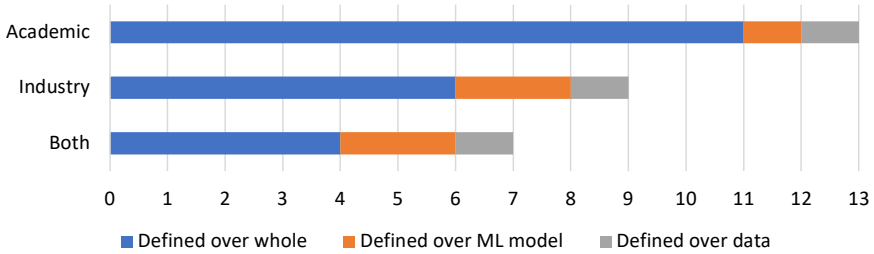

Fig. 15 NFR definition scope indicated by participants from different contexts.

RQ8 (Industry and Academia), Finding 1: Participants from academia offered the most consistent ratings of the importance of NFRs, but also the lowest. They placed a higher importance on fairness, maintainability, and transparency than industrial participants.

RQ8 (Industry and Academia), Finding 2: Participants from industry most highly value reliability, accuracy, and integrity. They place higher importance on justifiability, interoperability, and interpretability than academics.

Participants in a blended context placed a far higher focus on fairness, transparency, explainability than industrial participants, and on privacy, justifiability, and transparency than academic participants. This is particularly notable, because no participants from an industrial context rated fairness as very important, and one referred to it as not important. Similarly, no participants from an academic context indicated that transparency or explainability were very important. Participants in a blended position seem to be more interested in being able to understand how a model comes to a decision than those in either individual context.

Those in a blended position are also disproportionately less interested in fault tolerance, portability, and simplicity than participants in either an industrial or an academic role. All three received no very important votes and 1-2 low or not important votes from blended participants, and had the three lowest average scores for this group in Table 6. Fault tolerance—in particular—is relatively high in importance for purely industrial participants.

RQ8 (Industry and Academia), Finding 3: Participants from a blended context placed a higher importance on fairness, transparency, explainability, justifiability, and privacy than other groups. They also placed the highest average importance on NFRs, but had the largest variance as well. They placed a lower emphasis on fault tolerance, portability, and simplicity.

4.3.2 Differences in Scope of NFRs (RQ2)

In Fig. 15, we indicate the scope of NFR definition for each group. Those from an academic role place this highest emphasis on NFRs for the complete

system, while only one participant indicated that NFRs should be defined over the model and one over the data. The results are similar for the other groups, with an additional vote for the model in each group.

RQ8 (Industry and Academia), Finding 4: Participants from all groups largely favored definition of NFRs over the whole system.

4.3.3 Differences in NFR Challenges (RQ3)

We asked survey respondents for their opinions on four NFR challenges. The responses are shown, by context of the participant, in Fig. 16–19.

All academic participants agreed that domain dependency was a challenge (Fig. 16). This view is largely shared by those from an industrial context (89%), with only one neutral statement (11%). The only disagreement comes from the blended group. However, 67% of respondents from a blended background still agree with the challenge.

Regarding rigorous testing (Fig. 17), both academic and industrial practitioners largely agreed on the importance of the challenge. One industrial participant was neutral, while one academic participant disagreed. However,industry participants found this challenge more important—75% of industrial participants strongly agreed that rigorous testing is a challenge, where only 30% from an academic context strongly agreed. Those from the blended context largely agreed (71%) with the challenge, but there was one neutral vote and one disagreement—as well as a relatively low (29%) proportion of strong agreements.

On the challenge of lack of awareness among customers (Fig. 18), academic participants had the strongest level of agreement—91% agreed or strongly agreed. Industry was split on this challenge. 66% agreed or strongly agreed, but there were also neutral, disagree, and strongly disagree votes. The blended group was somewhat neutral on this challenge—66% agreed, but did not strongly agree, and 33% neither agreed or disagreed.

Finally, we examined the lack of awareness among engineers (Fig. 19). Industrial and academic practitioners show similar distributions of opinions, with 66% and 70% of respondents agreeing with the challenge. There is slightly more strong agreement from industry (44%, compared to 30%), but the disagreement is the same. In contrast, those from a blended context were sharply divided on this question—33% agreed, 33% were neutral, and 33% either disagreed or strongly disagreed.

RQ8 (Industry and Academia), Finding 5: Academic participants showed stronger agreement regarding domain dependency and lack of awareness among customers than industrial participants, while industrial participants showed stronger agreement on rigorous testing. Industrial participants were split on lack of awareness among customers.

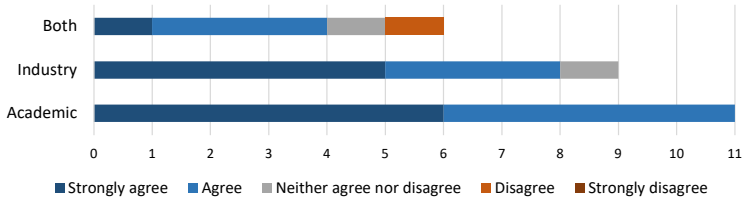

Fig. 16 Comparison of the opinions on the statement "Domain dependency of NFRs for ML-enabled systems is a challenge."

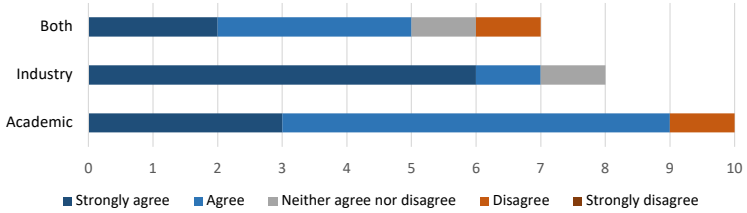

Fig. 17 Comparison of the opinions on the statement "Implementing rigorous testing is a challenge for testing NFRs for ML-enabled systems."

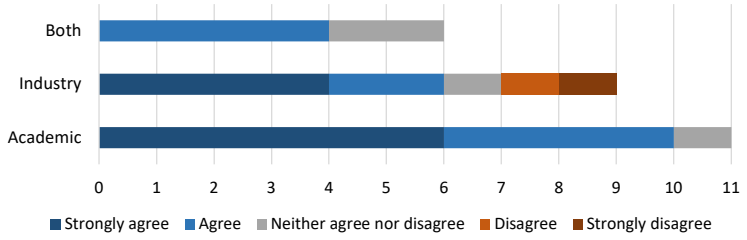

Fig. 18 Comparison of the opinions on the statement "Lack of awareness among customers about NFRs for ML-enabled systems is a challenge."

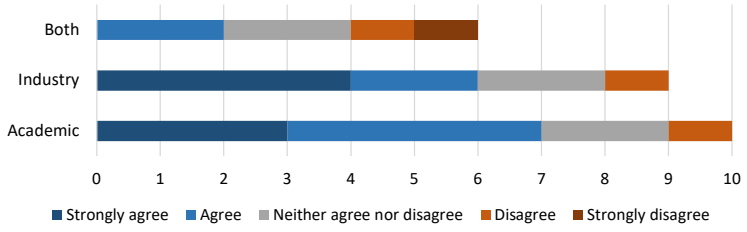

Fig. 19 Comparison of the opinions on the statement "Lack of awareness among engineers about NFRs for ML-enabled systems is a challenge."

RQ8 (Industry and Academia), Finding 6: Those from both academia and industry showed the largest disagreements from the other groups. The blended group was particularly split on lack of awareness among engineers, and agreed more weakly than the other groups on the other challenges.

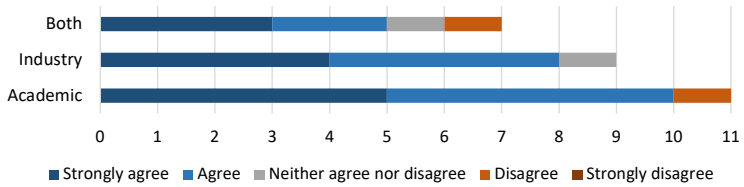

Fig. 20 Comparison of the opinions on the statement "NFR measurements for ML-enabled systems are dependent on the context."

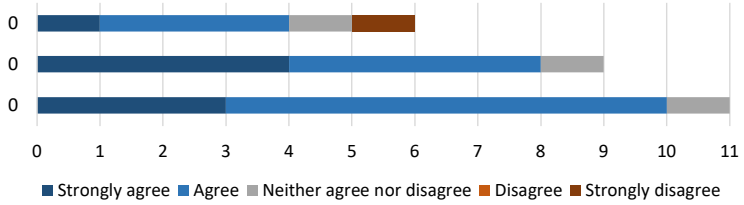

Fig. 21 Comparison of the opinions on the statement "NFR measurements for ML-enabled systems can be dependent on another NFR defined for the other parts of same system, the whole system, the ML model, or the data."

4.3.4 Differences in NFR Measurements (RQ4, RQ5, RQ6)

We also compared opinions of practitioners from different groups regarding two statements about NFR measurement. The results are shown in Fig. 20–21.

Regarding the statement, "NFR measurements for ML-enabled systems are dependent on the context", approximately 90% of both industrial practitioners and academics agreed. The only difference between the two groups was that one academic participant disagreed, while one industrial participant remained neutral. Those from the blended group also largely agreed. However, there was one neutral vote and one disagreement.

Regarding "NFR measurements for ML-enabled systems can be dependent on another NFR defined for the other parts of same system, the whole system, the ML model, or the data.", those from a pure industrial or academic context almost universally agreed—with more from industry strongly agreeing (44% versus 27%). Again, however, there is a higher level of disagreement from the blended group, with one neutral vote and one strong disagreement.

RQ8 (Industry and Academia), Finding 7: All three groups largely agreed with statements regarding NFR measurement dependencies. Those from the blended group had a larger level of disagreement.

4.3.5 Differences in NFR Measurement Challenges (RQ7)

Finally, we compared the groups on their agreement with NFR measurement challenges. The results are shown in Fig. 22–23

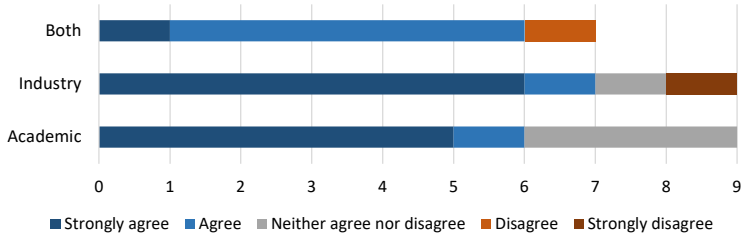

Fig. 22 Comparison of the opinions on the statement "Missing measurement baselines is a challenge for measuring NFRs for ML-enabled systems."

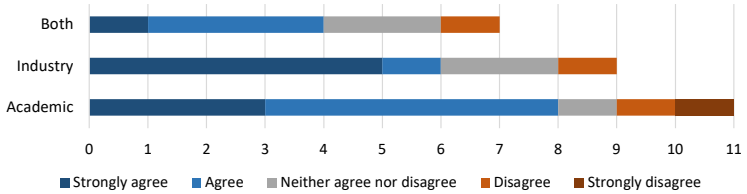

Fig. 23 Comparison of the opinions on the statement "Uncertainty is a challenge for identifying, defining and measuring NFRs for ML-enabled software."

All three groups largely agree that missing measurement baselines are a challenge for measuring NFR attainment (Fig. 22). They differ, however, in the level of strong agreement—78% of industrial participants strongly agree, compared to 56% of academics and only 14% of those from the blended group. The academic group has the largest percentage of neutral opinions (33%).

As shown in Fig. 23, The majority in all groups agree that uncertainty is a challenge (57% for blended, 66% for industry, 73% for academia). However, there are disagree and strong disagree votes among all groups, indicating a split in opinion regardless of participant context. Again, the blended group shows the weakest level of strong agreement (14%).

RQ8 (Industry and Academia), Finding 8: Those from all three contexts largely agree that missing baselines and uncertainty are challenges. However, those from the blended group show weaker levels of agreement.

5 Discussion and Future Work

In this section, we discuss our findings. We aim to identify the level of emphasis on individual NFRs for ML systems from the practitioner perspective, and to identify challenges practitioners face working with NFRs for ML systems.

NFR Importance (RQ1): The interview participants identified NFRs as either more or less important for ML, while the survey participants rated each NFR on a five-point scale from "Not Important" to "Very Important". Together, this data offers an indication of how important each NFR is for ML systems.

All participants believe that NFRs play a vital role in the successful development of ML systems. Participants also indicated that the scope of definition and measurements of NFRs are not the same for ML systems, and that some NFRs—such as adaptability or maintainability—have a very different meaning in an ML context.

Our results show that some NFRs are agreed upon as very important for ML systems (e.g., accuracy, reliability, integrity, and security), while opinions are split about others (e.g., efficiency, fairness, flexibility, portability, reusability, testability, and usability). Several recent mapping studies have created NFR rankings, based on literature searches, which are comparable to our results. A comparison of the most important and most frequently mentioned NFRs in these recent studies with our results in our study, including the interviews and survey results from participants with different backgrounds, is presented in Table 7. While these previous studies use scientific literature as their sources, we use an interview and survey with mixed responses from industrial and academic participants. We can see some similarities in these results, with performance/correctness/accuracy being high on all lists. In general, many of the top NFRs as found through our study appear to some degree in the literature lists (viewing integrity as related to reliability). However, some NFRs that have been focused on in the literature were not of significant interest to our participants (e.g., privacy, fairness [1]). Security is near the end of the list of the top NFRs on six out of eight lists, with our interviewees and industrial participants seeming to value this quality to a lesser degree. Furthermore, reliability appears in our lists, but only appears in one review from the literature—indicating that this quality is valued in practice, but may be less so in academic work. Similarly, our industrial survey participants identify justifiably and traceability as important, with these qualities do not appear in the top positions of other lists. This could be another indication of industrial needs differing from the focus of researchers. Further studies with a larger pool of respondents are needed to confirm these results.

Table 7 A Comparison of Most important/Most mentioned NFRs in Recent Systematic Mapping Studies (SMS) compared to our Survey Results

				Survey Results		
SMS [30]	**SMS [40]**	**SMS [1]**	**Our Interview Results**	**Academic Participants**	**Industrial Participants**	**Participants with Both Backgrounds**
Performance	Safety	Privacy	Accuracy	Accuracy	Reliability	Accuracy
Accuracy	Functional correctness	Fairness	Reliability	Completeness	Accuracy	Integrity
Efficiency	Robustness	Accuracy	Usability	Integrity	Integrity	Reliability
Security	Reliability	Performance	Testability	Reliability	Justifiability	Security
Complexity	Security	Security	Explainability	Security	Traceability	Safety

Fairness, in particular, is worth discussing. Fairness has received emphasis in recent literature and discussion on ML [27, 41], but the view of fairness among our participants is more mixed. Fairness is relatively important on average—ranked roughly in the middle in Table 6—but industrial practitioners place less importance on the topic than academic participants. This may

indicate that the emphasis on fairness in the literature is not yet reflected in practice. These results could also be dependent on the industry domains of our participants. For example, automotive practitioners may be more concerned with safety than fairness; however, to minimize the number of questions, we did not specifically ask about domain in our survey data.

Interestingly, those from a blended academic and industrial context rate fairness as being far more important than those in either a pure academic or industrial role. In addition, we see—in general—that transparency, traceability, explainability, and justifiability were particularly important for those who worked in the blended context. This suggests that the blended group has a great concern with the black-box nature of ML models, and places importance on understanding how models make decisions. We are not sure exactly why this difference in opinion occurs, but the combination of theoretical knowledge and in-the-field experience among this pool of participants may lead to this concern. Those in this group may work more closely with decision-making, with model design, or with development of new ML techniques (rather than pure usage of techniques).

NFRs are often seen as a theoretical concept, and industrial participants in our interviews sometimes needed the term clarified. Still, we see that participants from industry or a blended context generally rated specific NFRs as more important than people working in only academia. While they may not use the same terminology in practice, they understood the practical implications of NFRs for their practice. There are differences in opinion between industry and the academic group on the importance of some NFRs. In particular, industrial participants placed higher importance on justifiability, interoperability, and interpretability than the academic group.

Building on these results, we are working on a systematic literature review for selected NFRs in ML-based systems. We also recently published as exploratory mapping study [30] exploring interest in NFRs in literature. These efforts will allow us to more precisely compare the focus of research literature to the focus of practitioners. We will try to identify differences in literature on the treatment of NFRs between traditional and ML systems, and would also like to explore the identification of more NFRs for ML from an ethical and safety context.

Scope of NFRs (RQ2): The scope considered for definition of NFRs varied somewhat between interviewees and survey participants. Most interview participants focused on the ML model, with less emphasis on the whole system. On the other hand, the majority of the survey participants defined NFRs over the whole system, with less emphasis on the model. However, the answers also varied based on the participants' roles. Among both interview and survey participants, there was little interest in NFRs over data—in contrast to recommendations and challenges in recent literature [33, 56].

We wish to further investigate how NFR treatment differs depending on scope—identifying and differentiating NFRs over different system parts. We plan to develop definitions, guidelines, and methods for treating NFRs over

different parts of the ML systems (e.g., how to define and achieve reproducibility over ML results or adaptability in the ML model). Our recent mapping study offers a starting point for this research, with a preliminary assessment of potential scope [30]. In order to produce concrete results, we will focus on specific NFRs for particular domains (e.g., safety and accuracy in automotive perception systems for self-driving vehicles).

NFR and ML-related Challenges (RQ3): Many NFR and ML-related challenges are discussed in previous studies. In [34], we described different challenges in terms of NFRs for ML, such as fragmented and incomplete understanding of NFRs for ML, the effects of ML algorithms on desired qualities, lack of understanding of how ML-based solutions integrate with typical software from a quality perspective, and so on. Chazette et al. discussed the difference of opinions from survey participants regarding explainability as an NFR and identified explainability as a challenging NFR [19]. Though we could discuss many other "known" challenges in terms of NFRs and ML, we focus on the specific findings from our interview and survey.

Our results illustrate that most of the practitioners experience challenges in defining and measuring NFRs for ML systems, including uncertainty, domain dependence and dependencies among requirements, rigorous testing, and regulations. The results show that 76% of survey participants have encountered at least one of these challenges in some portion of projects. Academic participants showed stronger agreement on whether domain dependence was a challenge, while industrial participants showed stronger agreement on rigorous testing.

Lack of awareness of NFRs amongst both customers and engineers were also raised as challenges. Survey respondents from a blended context showed less agreement than participants from either a pure academic or industrial context on these two challenges. It is difficult to understand the reasons for this difference, but again we see that the combination of contexts has an effect on our results. A possible reason is that those in a blended context work in a more isolated or senior role with less exposure to non-technical customers or engineers that lack experience in ML. However, we lack the data needed to concretely assess this hypothesis.

Rigorous testing was recognized as a challenge by the participants who come from an academic and industrial context. However, academic participants showed a lower level of "strong" agreement (30%) than industrial participants (75%). The possible reason for this disagreement could be the difference between the size and complexity of the systems the participants handle. In general, industry participants need to test more complex and larger systems, hence, they more strongly believe that rigorous testing is a challenge. However, if we extend to both "strong agreement" and "agreement", 90% of academic participants agreed that rigorous testing is a challenge, compared to 87.5% of industrial participants. Therefore, the core difference is the emphasis.

Although we asked interviewees about NFR-related challenges, they often responded with more general ML challenges. It is likely that it is not so easy for interviewees to separate the sources for these challenges.

NFR Measurements (RQ4): It is important to measure attainment of NFRs for ML systems. All interviewees said that they do measure NFRs for ML systems. It is possible to measure some NFRs (e.g., accuracy, privacy) using standard or ML-specific measures (e.g., precision, recall), but many (e.g., trust, fairness) are difficult to measure because they are not easily quantifiable. In safety-critical situations (e.g., autonomous driving, e-health), the combined judgment of both machines and humans should be used to measure NFRs.

NFR Measurement Scope (RQ5): Interviewees often considered NFRs definition over the whole system, but they generally measured over the ML model. The survey participants both defined and measured over the system. Again, neither survey or interview participants measure over the data.

NFR Measurement Capture (RQ6): Context is important for NFR measurement capture, and interviewees capture NFR measurements using scripting, checklists, interviewees, and traceability tools.

Respondents from a blended context show less agreement on NFR measurement being dependent on context or NFRs defined over different parts of the system than those in a pure academic or industrial context.

NFR Measurement Challenges (RQ7): Many NFR measurement-related challenges (e.g., lack of knowledge or practice, missing measurement baseline, domain dependence, complex ecosystem, etc.) were described by interview participants. Survey participants generally agreed that these challenges exist. Once again, participants from a blended context show weaker agreement than the pure academic or industrial contexts on whether missing measurement baselines and uncertainty are challenges.

Differences Between Industry and Academia (RQ8): Participants from an academic context offer more consistent results than participants from industrial or blended contexts—yet also often rank NFRs as less important than those in the other contexts. Those in the blended group yielded the least consistent results and the highest average importance rating. The blended group also often yielded stronger differences in opinion from the other two contexts.

One possible explanation is a difference in experience level between the three groups. The academic group had the least average experience in all three areas—ML, RE, and NFRs. However, their ML experience was comparable to the industrial participants. The industrial participants had the highest average level of experience in NFRs. They also had more experience in RE than academic participants, but less than the blended group. Finally, the blended group had the most experience in ML and RE, but less in NFRs than the pure industrial group.

The high level of NFR experience in the industrial group could explain their preference for NFRs related to model performance (e.g., accuracy). Similarly, the higher level of ML experience in the blended group could explain to their focus on NFRs related to model explainability. The comparatively lower level of experience in RE and NFRs in the academic group could also help explain their overall lower ratings of importance. However, more data and a wider pool of participants would be needed to draw concrete conclusions.

Our results differ somewhat compared to Vogelsang and Borg [56] as their findings only focused on expainability, freedom from discrimination, and data specific requirements and challenges. These results may be due to the difference in the demographics of interview and survey participants. Vogelsang and Borg focus on data scientists, while only 20% of our interviewees and none of our survey respondents identified as data scientists. However, our results can be seen to echo the findings of Belani et al. [8]. Although we did not ask specifically about NFRs in the software lifecycle, we found many measurement-related challenges related to system operation and testing.

5.1 Research Gaps

Our findings reveal several gaps that can shape future work:

1. We need further work that focuses on those NFRs with a newly increased importance in an ML context—e.g., explainability, transparency, bias, or justifiability—or with different meanings (e.g., adaptability, maintainability). Although importance ratings for these NFRs are mixed, participants generally agreed that NFRs are defined and measured differently for ML-systems, thus requiring special attention. Further work in this area can include new or adjusted definitions, taxonomies, measurements, and methods. Such work has already begun for some NFRs (e.g., fairness [15] and transparency [26]), but it is often approached from a general SE, rather than an RE, perspective.
2. Further work is needed to evaluate the level of importance of different NFRs for ML systems as there is disagreement among practitioners. The directions mentioned above are also important in resolving disputes, as individuals may have different interpretations of these NFRs.
3. The domain specificity of our results should be further confirmed, e.g., differences in NFR importance for medical vs. banking vs. automotive practitioners. We hope that interpretations of NFRs may be domain independent, but the relative importance of NFRs will likely depend on the domain, as well as the context, as recently emphasized in [33].
4. Lack of awareness among both practitioners and customers creates misconceptions about NFRs for ML systems that must be addressed by further research.
5. Conceptualizations and methods are needed to address the scope of NFRs. There are different ways to view the sub-parts of the system, and these views may affect the way we categorize and define NFRs over elements of an ML system [53].
6. The NFR definition challenges that we identified—as well as the general ML development challenges— should be addressed from an RE perspective in future research (e.g., previous work on uncertainty in requirements, such as [16], could be extended to cover ML systems).
7. New measurements for NFRs in an ML context are needed (e.g., [31, 43]). Many NFRs are also difficult to measure in traditional systems, but ML adds new challenges. Furthermore, NFR measurements for ML systems

can be dependent on NFRs defined for the other parts of same system, and NFR measurements for ML systems are often dependent on a specific domain or context.

8. We also found further measurement-related challenges (e.g., missing measurement baselines). From an RE perspective, we must apply methods to understand complex ML ecosystems, to define and refine NFRs, or to make tradeoffs between NFRs (e.g., whether quality improvements are sufficient to justify the cost of rigorous testing).

Our findings provide a view of current practices and challenges experienced regarding NFRs in ML systems, but do not yet offer concrete solutions. This research is useful for practitioners to increase their awareness about NFRs in an increasingly-important ML context. This research also provides initial findings on the relative importance of NFRs for ML systems. We advocate the idea of NFR scope, which can help practitioners to understanding the applicability and meaning of different NFRs over different system parts. For practitioners, it is also useful to see the questions and challenges that other practitioners are facing, to understand that many of their current challenges are not unique, and to gain an indication of what they may expect to see in future projects.

Overall, we see that this area is challenging for practitioners, yet important. Although individual organizations may have their own knowledge and practices, they do not yet have well-established solutions for dealing with NFRs in this context.

5.2 Threats to Validity

Construct Validity: Several of our interviewees were not familiar with either the concept or terminology of NFRs, and wanted examples. One possible reason for this is that the interviewees are representative of the data science and ML field, and may not have software engineering training. As a result, they may not know software engineering terminology or particular concepts. To exemplify NFRs, we showed a version of McCall's software quality hierarchy [17]. We could have used other available NFR hierarchies, as there are many. However, this example was used because of its prominence in RE literature.

In addition, we note that several survey participants had less than one year of experience in RE and NFRs, and were perhaps not familiar with the terminology, and some questions could be difficult for them to understand. To reduce this threat, as part of each question, we included short definitions of terms. In the survey introduction, we also provided a description of survey context and definitions of terms.

We previously noted that questions concerning how NFR measurements were captured were not easy for the interviewees to understand. They could have understood each NFR differently. In retrospect, this question could have been more clearly written. Still, we believe the results collected were interesting.

Conclusion Validity: Showing a particular NFR hierarchy can bias answers towards that hierarchy. However, the differences between hierarchies are not extensive.

There is a risk of uninformative answers from survey participants who lack familiarity with NFRs, requirements engineering, or NFRs. Therefore, we collected demographic data of the survey participants, as well as on their familiarity with NFRs. We excluded data for one participant who did not fill in the demographic data, and who did not have any experience or familiarity with ML, RE, or NFRs.

The number of responses for both the survey and interviews may affect the reliability of our conclusions. However, given that our target demographic consists of in-demand personnel with knowledge in multiple areas (AI, SE), we feel that our number of participants is sufficient to draw conclusions that can be evaluated and refined in further work.

At times, open responses to the interview or survey were not clear or specific enough. In those cases, interpretation was required. There is a risk that interpretations are biased. However, we interpreted the quotes individually and then discussed among us to form a common understanding.

Internal Validity: In our work, we applied thematic coding. This is a qualitative practice that suffers from known internal validity threats. We mitigated these threats by performing independent coding over half the interviews and comparing results, finding sufficient agreement. We also used standard coding tools (NVivo) to help ease the process. We made our results available for further analysis.

We can consider whether our interview findings were close to reaching saturation after 10 participants. We found towards the end of our analysis that the codes were generally converging to a stable set. However, the code "justifiability" was added in the last interview. An eleventh interview was conducted,but did not reveal any new results. Thus, we believe further interviews could help to enrich our findings, but would not produce significant additions.

Our sampling technique for the interview study found a number of participants who straddle the boundaries between industry and academia. Similarly, our survey participants included a large number of respondents from academia or also on that boundary. This may be a result of our circle of contacts, and reflective of the practitioners interested in responding to a survey. However, we also believe that those who are interested in the topics covered in this paper are often mid- to upper-level management, and often have a strong academic or research-oriented background.

Another threat could that the length of the survey demotivated people to participate. However, we sent the survey questionnaire to three other researchers to test whether they understood the questions before widely distributing the survey. We changed the wording and reduced the number of questions according to their suggestions.

External Validity: Although our participants come from different parts of the world, we still had a large number of respondents from the Nordic countries. However, we found participants from a diverse set of product domains, and we believe that the Nordic countries have a strong and international AI-oriented industry. Thus, our participants are fairly representative of the software development industry as a whole.

6 Conclusions

We have conducted a qualitative interview study followed by a survey to understand the perception of and practices for NFRs in ML systems. The interviewees and survey participants agree that NFRs play an important role in the success of the ML systems. Traditional NFRs like accuracy can still be important for ML, as new NFRs such as transparency, fairness, and explainability are gaining more importance. However, the level of importance of NFRs for ML systems varied based on the background of the participants, therefore, more research is needed in this area. Most practitioners think of NFRs over the whole system, or over the model, few consider data. we also see that all groups generally agreed on the scope of defining NFRs. Therefore, research on developing methods for treating NFRs over different parts or scopes of the ML systems is important. Most practitioners experience challenges in defining and measuring NFRs for ML systems, they also often experience general ML challenges while considering NFRs for ML systems. From an industrial and research perspective, NFRs for ML are not well organized and well developed and their consideration is mainly in an initial stage. The challenges and complexities of NFR-related research remain but are intensified by ML. Further research is needed to develop NFR definition and measurement methods, and to overcome NFR-related challenges for ML systems.

7 Acknowledgements

This work is supported by a Swedish Research Council (VR) Project: Non-Functional Requirements for Machine Learning: Facilitating Continuous Quality Awareness (iNFoRM).

References

[1] Mohamed Abdullahi Ali, Ng Keng Yap, Abdul Azim Abd Ghani, Hazura Zulzalil, Novia Indriaty Admodisastro, and Amin Arab Najafabadi. A systematic mapping of quality models for ai systems, software and components. *Applied Sciences*, 12(17):8700, 2022.

[2] David Ameller, Xavier Franch, Cristina Gómez, Silverio Martínez-Fernández, Joao Araújo, Stefan Biffl, Jordi Cabot, Vittorio Cortellessa,

Daniel Méndez, Ana Moreira, et al. Dealing with non-functional requirements in model-driven development: A survey. *IEEE Transactions on Software Engineering*, 2019.

[3] Saleema Amershi, Andrew Begel, Christian Bird, Robert DeLine, Harald Gall, Ece Kamar, Nachiappan Nagappan, Besmira Nushi, and Thomas Zimmermann. Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pages 291–300. IEEE, 2019.

[4] Marco Anisetti, Claudio A Ardagna, Ernesto Damiani, and Paolo G Panero. A methodology for non-functional property evaluation of machine learning models. In *Proceedings of the 12th International Conference on Management of Digital EcoSystems*, pages 38–45, 2020.

[5] Chetan Arora, Mehrdad Sabetzadeh, Shiva Nejati, and Lionel Briand. An active learning approach for improving the accuracy of automated domain model extraction. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 28(1):1–34, 2019.

[6] Anders Arpteg, Björn Brinne, Luka Crnkovic-Friis, and Jan Bosch. Software engineering challenges of deep learning. In *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 50–59. IEEE, 2018.

[7] Nagadivya Balasubramaniam, Marjo Kauppinen, Kari Hiekkanen, and Sari Kujala. Transparency and explainability of ai systems: Ethical guidelines in practice. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 3–18. Springer, 2022.

[8] Hrvoje Belani, Marin Vukovic, and Željka Car. Requirements engineering challenges in building ai-based complex systems. In *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, pages 252–255. IEEE, 2019.

[9] Richard Berntsson Svensson and Björn Regnell. A case study evaluation of the guideline-supported quper model for elicitation of quality requirements. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 230–246. Springer, 2015.

[10] Daniel M Berry. Requirements engineering for artificial intelligence: What is a requirements specification for an artificial intelligence? In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 19–25. Springer, 2022.

[11] Adrien Bibal, Michael Lognoul, Alexandre de Streel, and Benoît Frénay. Legal requirements on explainability in machine learning. *Artificial Intelligence and Law*, pages 1–21, 2020.

[12] Reuben Binns. Fairness in machine learning: Lessons from political philosophy. In *Conference on Fairness, Accountability and Transparency*, pages 149–159. PMLR, 2018.

[13] Or Biran and Courtenay Cotton. Explanation and justification in machine learning: A survey. In *IJCAI-17 workshop on explainable AI (XAI)*, volume 8, pages 8–13, 2017.

[14] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1175–1191, 2017.

[15] Yuriy Brun and Alexandra Meliou. Software fairness. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 754–759, 2018.

[16] Antoine Cailliau and Axel Van Lamsweerde. Handling knowledge uncertainty in risk-based requirements engineering. In *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, pages 106–115. IEEE, 2015.

[17] Joseph P Cavano and James A McCall. A framework for the measurement of software quality. In *Proceedings of the software quality assurance workshop on Functional and performance issues*, pages 133–139, 1978.

[18] Robert Challen, Joshua Denny, Martin Pitt, Luke Gompels, Tom Edwards, and Krasimira Tsaneva-Atanasova. Artificial intelligence, bias and clinical safety. *BMJ Quality & Safety*, 28(3):231–237, 2019.

[19] Larissa Chazette and Kurt Schneider. Explainability as a non-functional requirement: challenges and recommendations. *Requirements Engineering*, 25(4):493–514, 2020.

[20] Lawrence Chung, Brian A Nixon, Eric Yu, and John Mylopoulos. *Non-functional requirements in software engineering*, volume 5. Springer Science & Business Media, 2012.

[21] Jane Cleland-Huang, Raffaella Settimi, Xuchang Zou, and Peter Solc. Automated classification of non-functional requirements. *Requirements engineering*, 12(2):103–120, 2007.

[22] John W Creswell and J David Creswell. *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage publications, 2017.

[23] Fabiano Dalpiaz and Nan Niu. Requirements engineering in the days of artificial intelligence. *IEEE Software*, 37(4):7–10, 2020.

[24] Joerg Doerr, Daniel Kerkow, Tom Koenig, Thomas Olsson, and Takeshi Suzuki. Non-functional requirements in industry-three case studies adopting an experience-based nfr method. In *13th IEEE International Conference on Requirements Engineering (RE'05)*, pages 373–382. IEEE, 2005.

[25] Jonas Eckhardt, Andreas Vogelsang, and Daniel Méndez Fernández. Are" non-functional" requirements really non-functional? an investigation of non-functional requirements in practice. In *Proceedings of the 38th International Conference on Software Engineering*, pages 832–842, 2016.

[26] Heike Felzmann, Eduard Fosch Villaronga, Christoph Lutz, and Aurelia Tamò-Larrieux. Transparency you can trust: Transparency requirements for artificial intelligence between legal norms and contextual concerns. *Big Data & Society*, 6(1):2053951719860542, 2019.

[27] Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 329–338, 2019.

[28] Matthias Galster and Eva Bucherer. A taxonomy for identifying and specifying non-functional requirements in service-oriented development. In *2008 IEEE Congress on Services-Part I*, pages 345–352. IEEE, 2008.

[29] Martin Glinz. On non-functional requirements. In *15th IEEE International Requirements Engineering Conference (RE 2007)*, pages 21–26. IEEE, 2007.

[30] Khan Mohammad Habibullah, Gregory Gay, and Jennifer Horkoff. Non-functional requirements for machine learning: An exploration of system scope and interest. *SE4RAI workshop, 44th International Conference on Software Engineering (ICSE 2022), (accepted)*, 2022.

[31] Khan Mohammad 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)*, pages 13–23. IEEE, 2021.

[32] Richard Hawkins, Colin Paterson, Chiara Picardi, Yan Jia, Radu Calinescu, and Ibrahim Habli. Guidance on the assurance of machine learning in autonomous systems (amlas). *arXiv preprint arXiv:2102.01564*, 2021.

[33] Hans-Martin Heyn, Eric Knauss, Amna Pir Muhammad, Olof Eriksson, Jennifer Linder, Padmini Subbiah, Shameer Kumar Pradhan, and Sagar Tungal. Requirement engineering challenges for ai-intense systems development. In *1st Workshop on AI Engineering – Software Engineering for AI (WAIN2021)*. IEEE, 2021.

[34] Jennifer Horkoff. Non-functional requirements for machine learning: Challenges and new directions. In *2019 IEEE 27th International Requirements Engineering Conference (RE)*, pages 386–391. IEEE, 2019.

[35] Fuyuki Ishikawa and Nobukazu Yoshioka. How do engineers perceive difficulties in engineering of machine-learning systems?-questionnaire survey. In *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. IEEE, 2019.

[36] Aleksander Jarzębowicz and Paweł Weichbroth. A systematic literature review on implementing non-functional requirements in agile software development: Issues and facilitating practices. In Adam Przybyłek, Jakub Miler, Alexander Poth, and Andreas Riel, editors, *Lean and Agile Software Development*, pages 91–110, Cham, 2021. Springer International Publishing.

[37] Toshihiro Kamishima, Shotaro Akakamishima2011fairnessho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 643–650. IEEE, 2011.

[38] Harsimran Kaur, Gurgaon ASU, and Ashish Sharma. Non-functional requirements research: Survey. *Int. J. Sci. Eng. Appl*, 3(6), 2014.

[39] Lucy Ellen Lwakatare, Aiswarya Raj, Jan Bosch, Helena Holmström Olsson, and Ivica Crnkovic. A taxonomy of software engineering challenges for machine learning systems: An empirical investigation. In *International Conference on Agile Software Development*, pages 227–243. Springer, Cham, 2019.

[40] Silverio Martínez-Fernández, Justus Bogner, Xavier Franch, Marc Oriol, Julien Siebert, Adam Trendowicz, Anna Maria Vollmer, and Stefan Wagner. Software engineering for ai-based systems: a survey. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 31(2):1–59, 2022.

[41] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[42] Payman Mohassel and Yupeng Zhang. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 19–38. IEEE, 2017.

[43] Koji Nakamichi, Kyoko Ohashi, Isao Namba, Rieko Yamamoto, Mikio Aoyama, Lisa Joeckel, Julien Siebert, and Jens Heidrich. Requirements-driven method to determine quality characteristics and measurements for machine learning software and its evaluation. In *2020 IEEE 28th International Requirements Engineering Conference (RE)*, pages 260–270. IEEE, 2020.

[44] Soroosh Nalchigar, Eric Yu, and Karim Keshavjee. Modeling machine learning requirements from three perspectives: a case report from the healthcare domain. *Requirements Engineering*, 26(2):237–254, 2021.

[45] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019.

[46] Anna Perini, Angelo Susi, and Paolo Avesani. A machine learning approach to software requirements prioritization. *IEEE Transactions on Software Engineering*, 39(4):445–461, 2012.

[47] Mona Rahimi, Jin LC Guo, Sahar Kokaly, and Marsha Chechik. Toward requirements specification for machine-learned components. In *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, pages 241–244. IEEE, 2019.

[48] Paul Ralph, Nauman bin Ali, Sebastian Baltes, Domenico Bianculli, Jessica Diaz, Yvonne Dittrich, Niel Ernst, Michael Felderer, Robert Feldt, Antonio Filieri, Breno Bernard Nicolau de França, Carlo Alberto Furia, Gregory Gay, Nicolas Gold, Daniel Graziotin, Pinjia He, Rashina Hoda, Natalia Juristo, Barbara A. Kitchenham, Valentina Lenarduzzi, Jorge Martínez, Jorge Melegati, Daniel Méndez, Tim Menzies, Jefferson Molleri, Dietmar Pfahl, Romain Robbes, Daniel Russo, Nyyti Saarimäki, Federica Sarro, Davide Taibi, Janet Siegmund, Diomidis Spinellis, Miroslaw Staron, Klaas-Jan Stol, Margaret-Anne Storey, Davide Taibi, Damian A. Tamburri, Marco Torchiano, Christoph Treude, Burak Turhan, Xiaofeng Wang, and Sira Vegas. ACM SIGSOFT empirical standards. *CoRR*, abs/2010.03525, 2021.

[49] Björn Regnell, Martin Höst, and Richard Berntsson Svensson. A quality performance model for cost-benefit analysis of non-functional requirements applied to the mobile handset domain. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*, pages 277–291. Springer, 2007.

[50] Per Runeson and Martin Höst. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14(2):131–164, 2009.

[51] Vaibhav Sachdeva and Lawrence Chung. Handling non-functional requirements for big data and iot projects in scrum. In *2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*, pages 216–221. IEEE, 2017.

[52] David Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. Hidden technical debt in machine learning systems. *Advances in neural information processing systems*, 28:2503–2511, 2015.

[53] Julien Siebert, Lisa Joeckel, Jens Heidrich, Koji Nakamichi, Kyoko Ohashi, Isao Namba, Rieko Yamamoto, and Mikio Aoyama. Towards guidelines for assessing qualities of machine learning systems. In *International Conference on the Quality of Information and Communications Technology*, pages 17–31. Springer, 2020.

[54] Alex Smola and SVN Vishwanathan. Introduction to machine learning. *Cambridge University, UK*, 32(34):2008, 2008.

[55] Hugo Villamizar, Tatiana Escovedo, and Marcos Kalinowski. Requirements engineering for machine learning: A systematic mapping study. In *2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 29–36. IEEE, 2021.

[56] Andreas Vogelsang and Markus Borg. Requirements engineering for machine learning: Perspectives from data scientists. In *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, pages 245–251. IEEE, 2019.

[57] Hironori Washizaki, Foutse Khomh, Yann-Gaël Guéhéneuc, Hironori Takeuchi, Satoshi Okuda, Naotake Natori, and Naohisa Shioura. Software engineering patterns for machine learning applications (sep4mla) part 2. In *Proceedings of the 27th Conference on Pattern Languages of Programs*, pages 1–10, 2020.

[58] Jonas Winkler and Andreas Vogelsang. Automatic classification of requirements based on convolutional neural networks. In *2016 IEEE 24th International Requirements Engineering Conference Workshops (REW)*, pages 39–45. IEEE, 2016.