ML-Enabled Systems Model Deployment and Monitoring: Status Quo and Problems

Eduardo Zimelewicz[1], Antonio Pedro Santos Alves[1], Marcos Kalinowski[1], Daniel Mendez[2,9], Görkem Giray[3], Kelly Azevedo[1], Hugo Villamizar[1], Niklas Lavesson[2], Tatiana Escovedo[1], Helio Lopes[1], Stefan Biffl[4], Jürgen Musil[4], Michael Felderer[5,6], Stefan Wagner[7], Teresa Baldassarre[8], and Tony Gorschek[2,9]

[1] Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Brazil
[2] Blekinge Institute of Technology (BTH), Sweden
[3] Independent Researcher, Turkey
[4] Vienna University of Technology (TU Wien), Áustria
[5] German Aerospace Center (DLR), Germany
[6] University of Cologne, Germany
[7] University of Stuttgart, Germany
[8] University of Bari, Italy
[9] fortiss GmbH, Germany

Abstract. [Context] Systems that incorporate Machine Learning (ML) models, often referred to as ML-enabled software systems, have become commonplace. However, empirical evidence on how ML-enabled systems are engineered in practice is still limited; this is especially true for activities surrounding ML model dissemination. [Goal] We investigate contemporary industrial practices and problems related to ML model dissemination, focusing on the model deployment and the monitoring ML lifecycle phases. [Method] We conducted an international questionnaire-based online survey to gather practitioner insights on how ML-enabled systems are engineered. We gathered 188 complete responses from 25 countries. We analyze the status quo and problems reported for the model deployment and monitoring phases. We conducted statistical analyses on contemporary practices using bootstrapping with confidence intervals and qualitative analyses on the reported problems involving open and axial coding procedures. [Results] Practitioners perceive the model deployment and monitoring phases as relevant but also as difficult. With respect to model deployment, models are typically deployed as separate services, with limited adoption of MLOps principles. Reported problems include difficulties in designing the architecture of the infrastructure for production deployment and legacy application integration. Concerning model monitoring, many of the models in production are not monitored. The main monitored aspects are inputs, outputs, and decisions. Reported problems involve the absence of monitoring practices, the need to create custom monitoring tools, and challenges in selecting suitable metrics. [Conclusion] Our results help already providing a better understanding of the adopted practices and problems in practice and support guiding ML deployment and monitoring research in a problem-driven manner.

Key words: Machine Learning, Deployment, Monitoring

1 Introduction

In recent years, the advancements in Machine Learning (ML) and, altogether, Artificial Intelligence (AI) have helped technological innovation and transformation across various industries. These ML-enabled systems have shown capabilities in automating complex tasks, making data-driven decisions, and enhancing overall efficiency. However, despite their immense potential, the implementation of ML-enabled systems requires practitioners to adapt processes to successfully develop, deploy, and monitor them in production operation. At the same level, software engineering (SE) practices can help speed up the development of such features. However, ML-enabled systems are inherently different by nature, rendering traditional SE practices insufficient to be directly applied, thus revealing new challenges [1, 2].

In regard to the current increase in ML system usage, it is important to identify potential industrial problems and the current status quo in terms of practices applied in the development of ML-enabled software systems. With the main goal of understanding the pain points and how those systems are made, we conducted a questionnaire-based online survey. Although many other concerns appeared in the responses, such as issues in Requirements Engineering and Data Quality [3], the work presented in this paper focuses on the model deployment and monitoring of ML-enabled systems. Our focus is on evaluating experienced challenges as well as approaches employed.

The main findings show that practitioners perceive the model deployment and monitoring phases as relevant but also challenging. With respect to model deployment, we observed that models are mainly deployed as separate services and that embedding the model within the consuming application or platform-as-a-service solutions are less frequently explored. Most practitioners do not follow MLOps principles and do not have an automated pipeline to retrain and redeploy the models, where the reported deployment problems include difficulties in designing the architecture of the infrastructure for production, considering scalability and financial constraints, and legacy application integration.

Concerning model monitoring, many of the models in production are not monitored at all, with the main aspects in the scope of monitoring being outputs and decisions taken. Reported problems include not having model-appropriate monitoring practices in place, the need to develop customized monitoring tools, and difficulties choosing the appropriate metrics.

As per the discussed results, this study lays the foundation for more problem-driven research, such as on the impact of MLOps adoption in industry, what appropriate practices could be, and how they can improve the production deployment.

The remainder of this paper is organized as follows. Section 2 provides the background and related work. In Section 3, we describe the research method. Section 4 presents then the results which we discuss further in Section 5. In

Section 6, we critically reflect upon the threats to validity and mitigation actions before concluding our paper with Section 7.

2 Background and Related Work

Machine Learning (ML) has witnessed various advancements in recent years, transforming various industries by enabling intelligent decision-making systems. Deploying ML models into real-world applications, however, presents complex challenges related to model performance, reliability, and maintenance. This section provides an overview of the research landscape concerning the deployment and monitoring of ML systems.

The use of ML in practical applications dates back to the year of 1952 when English Mathematician Arthur Samuel created the first Machine Learning program to play championship-level game of checkers [4]. However, it is in the past decade that ML deployments have gained widespread attention in practice due to the availability of large datasets, more powerful computing hardware, and improved algorithms. Despite the rapid growth in ML adoption, there still exists a significant gap between the development of ML models in testing environments and their successful deployment in real-world settings, as reported by Paleyes *et. al.* [5], especially in the fields of integration, monitoring, and updating a model. Further discussions show that, within the model deployment phase, adapting existing techniques such as DevOps could be extremely helpful to bring development and production environments closer. The term MLOps (Machine Learning Operations) follows the same concept by bringing together data scientists and operations teams, with Meenu *et. al.* [6] identifying activities in which organizations can improve their MLOps adoption.

To represent the main issues of transitioning models to production architectures, some challenges were also identified and categorized by Lewis *et. al.* [7]. First, utilizing software architecture practices that are proven effective to traditional applications but do not take into account the data-driven aspect of such projects, meaning that the design and development of ML models will have to be approached with new frameworks, as the one presented by Meenu *et. al.* [8]. Second, creating patterns and tactics to achieve ML Quality Attributes (QAs), where existing metrics will need to be revisited and new ones will be created to evaluate systems better. Third, employ monitorability as a driving quality attribute by having the infrastructure behind the monitoring platform responsible for collecting specific information related to changes in the dataset, as well as the incorporated user feedback, to observe the impacts on deployed ML systems. Fourth, co-architecting and co-versioning, where the architecture of the ML system itself, alongside the architecture that supports its life cycle, will have to be developed in sync, like the MLOps pipeline and the system integration, and the existing dataset as well as the programming code.

Apart from the architecture challenges, previous research has explored different deployment models for ML systems. Meenu *et. al.* [9] provided a literature review on AI deployment to design a deployment framework for these systems.

Today's deployment approaches range from traditional offline batch processing [10] to real-time streaming deployments [11], with an increase in the use of the cloud service deployments, based on FaaS (Function as a Service) [12], SaaS (Software as a Service) [13], PaaS (Platform as a Service) [14], and IaaS (Infrastructure as a Service) [15] solutions. Benefits of cloud adoption include the relief from the burden of server management, faster time to go into production, cost optimization, and performance increase. Alongside the deployment models, the existing software architecture approaches are also getting adapted to ML models such as containerization [16], microservices [17], and serverless computing [18] have gained prominence in ensuring model deployment flexibility and scalability.

Recent studies have focused on the monitoring and maintenance of ML models. Researchers have proposed techniques for detecting ML-specific metrics such as model drift, handling concept drift, and ensuring that models remain accurate and reliable over time [19, 20], which involves concepts such as statistical process control, anomaly detection, and continuous integration/continuous deployment (CI/CD) practices. The presented literature demonstrates the diverse nature of ML deployment and monitoring challenges.

3 Research Method

3.1 Goal and Research Questions

The main goal of the research study focused on surveying the current status quo and problems through the entire development lifecycle of an ML system, but for the context of the current paper, the analysis will be based on two of the most problematic concerns in maintaining the model: (i) making the model available as quickly as possible in production and (ii) managing the model and re-training it along its continuous deployment based on monitored aspects. From this goal, we inferred the following research questions:

- RQ1. What are contemporary practices for deploying ML models?
  This question aims at identifying the in-use practices and trends of the deployment stage. We refined this question further into three more detailed questions:
  - RQ1.1. What kind of approaches are used to deploy ML models?
  - RQ1.2. Which tools are used for automating model retraining?
  - RQ1.3. What are the MLOps practices and principles used?
- RQ2. What are the main problems faced during the deployment in the ML life cycle stage?
- RQ3. What are contemporary practices for monitoring ML models?
  This question aims at identifying the practices and trends of the *monitoring* stage. We refined this question further into three more detailed questions:
  - RQ3.1. What percentage of the ML-enabled system projects that get deployed into production have their ML models actually being monitored?
  - RQ3.2. What aspects of the models are monitored?

– RQ4. What are the main problems faced during the monitoring in the ML life cycle stage?
– RQ5. What is the percentage of projects that effectively go into production?

3.2 Survey Design

We designed our survey based on best practices of survey research [21], carefully conducting the following steps:

– **Step 1. Initial Survey Design**. We conducted a literature review on ML deployment and monitoring to provide the theoretical foundations for the related questions and answer options. From there, we drafted the initial survey by involving Software Engineering and Machine Learning researchers from PUC-Rio/Brazil with experience in R&D projects involving ML-enabled systems.
– **Step 2. Survey Design Review**. The survey was reviewed and adjusted based on online discussions and annotated feedback from Software Engineering and Machine Learning researchers from BTH/Sweden. Thereafter, the survey was also reviewed by the other co-authors.
– **Step 3. Pilot Face Validity Evaluation**. This evaluation involves a lightweight review by randomly chosen respondents. It was conducted with 18 Ph.D. students taking a Survey Research Methods course at UCLM/Spain (taught by the third author). They were asked to provide feedback on the clearness of the questions and to record their response time. This phase resulted in minor adjustments related to usability aspects and unclear wording. The answers were discarded before launching the survey.
– **Step 4. Pilot Content Validity Evaluation**. This evaluation involves subject experts from the target population. Therefore, we selected five experienced data scientists developing ML-enabled systems, asked them to answer the survey, and gathered their feedback. The participants had no difficulties answering the survey, and it took an average of 20 minutes. After this step, the survey was considered ready to be launched.

The final survey started with a consent form describing the purpose of the study and stating that it was conducted anonymously. The remainder was divided into 15 demographic questions (D1 to D15) followed by three specific parts with 17 substantive questions (Q1 to Q17): 7 on the ML life cycle and problems, 5 on requirements, and 5 on deployment and monitoring. This paper focuses on the ML life cycle problems related to model deployment and monitoring. The excerpts of the questions we deem relevant in the context of the paper at hand are shown in Table 1. The survey was implemented using the Unipark Enterprise Feedback Suite.

3.3 Data Collection

Our target population concerns professionals involved in building ML-enabled systems, including different activities, such as management, design, and devel-

Table 1. Research questions mapped to survey questions

RQ	Question No.	Description	Type
-	...	...	...
RQ5	D7	How many ML-enabled system projects have you participated in? Please, provide your best estimate:	Open
RQ5	D8	Of all the ML-enabled system projects you have participated in, how many were actually deployed into a production environment (e.g., released to the final customer)? Please, provide your best estimate:	Open
-	...	...	...
RQ2	Q4	According to your personal experience, please outline the main problems or difficulties (up to three) faced during the Model Deployment ML life cycle stage.	Open
RQ4	Q4	According to your personal experience, please outline the main problems or difficulties (up to three) faced during the Model Monitoring ML life cycle stage.	Open
-	...	...	...
RQ1.1	Q13	In the context of the ML-enabled system projects you participated in, which approach is typically used to deploy ML models?	Multiple Option and Open Text
RQ1.2 RQ1.3	Q14	Do you/your organization follow the practice and principles of ML-Ops in ML-enabled system projects? For instance, do you have an automated pipeline to retrain and deploy your ML models?	Single Option and Open Text
RQ3.1	Q15	Based on your experience, what percentage of the ML-enabled system projects that get deployed into production have their ML models actually being monitored?	Open
RQ3.2	Q16	Which of the following ML model aspects are monitored for the deployed ML-enabled system projects you have worked on?	Multiple Option and Open Text
-	...	...	...

opment. Therefore, it includes practitioners in positions such as project leaders, requirements engineers, data scientists, and developers. We used convenience sampling, sending the survey link to professionals active in our partner companies, and also distributed it openly on social media. We excluded participants who informed that they had no experience with ML-enabled system projects. Data collection was open from January 2022 to April 2022. In total, we received responses from 276 professionals, out of which 188 completed all four survey sections. The average time to complete the survey was 20 minutes. We conservatively considered only the 188 fully completed survey responses.

3.4 Data Analysis Procedures

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

Research questions *RQ1.1, RQ3.1, RQ3.2, and RQ5* concern a mix of closed questions and optional free fields, so we decided to use inferential statistics to analyze them. Our population has an unknown theoretical distribution (*i.e.*, the distribution of ML-enabled system professionals is unknown). In such cases, resampling methods - like bootstrapping - have been reported to be more reliable and accurate than inference statistics from samples [22, 21]. Hence, we use bootstrapping to calculate confidence intervals for our results, similar as done in [23]. In short, bootstrapping involves repeatedly taking samples with replacements and then calculating the statistics based on these samples. For each question, we take the sample of n responses for that question and bootstrap S resamples (with replacements) of the same size n. We assume n as the total valid answers of each question [24], and we set 1000 for S, which is a value that is reported to allow meaningful statistics [25].

For research questions *RQ1.2, RQ1.3, RQ2, RQ3.1, and RQ4*, which seeks to identify the main problems faced by practitioners involved in engineering ML-enabled systems, related to model deployment and monitoring, alongside questions regarding which current practices are being applied, what amount of models that are generally available for users and the current monitored aspects, had their corresponding survey question designed to be open text. We conducted a qualitative analysis using open and axial coding procedures from grounded theory [26] to allow the problems to emerge from the open-text responses reflecting the experience of the practitioners. The qualitative coding procedures were conducted by one PhD student, reviewed by her advisor at one site (Brazil), and reviewed independently by three researchers from two additional sites (Sweden and Turkey).

The questionnaire, the collected data, and the quantitative and qualitative data analysis artifacts, including Python scripts for the bootstrapping statistics and graphs and the peer-reviewed qualitative coding spreadsheets, are available in our open science repository [1].

[1] `https://doi.org/10.5281/zenodo.10092394`

4 Survey Results

In this section, we present the study results. First, we describe the study population and the perception of the relevance and difficulty of the ML deployment and monitoring phases. Thereafter we answer each of the research questions. The N in each figure caption is the number of participants that answered this question. We report the bootstrapping proportion P of the participants that checked the corresponding answer and its 95% confidence interval in square brackets.

4.1 Study Population

Figure 1 summarizes demographic information on the survey participants' countries, roles, and experience with ML-enabled system projects in years. It is possible to observe that the participants came from different parts of the world, representing various roles and experiences. While the figure shows only the ten countries with the most responses, we had respondents from 25 countries. As expected, our convenience sampling strategy influenced the countries, with most responses being from countries in which the authors had the most industrial contacts (Brazil, Turkey, Austria, Germany, Sweden, and Italy).

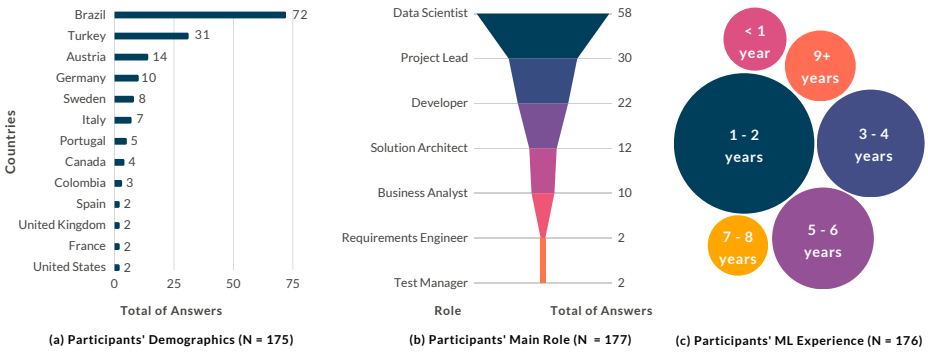

Fig. 1. Demographics for participants' countries, roles, and ML work experience

Regarding employment, 45% of the participants are employed in large companies (2000+ employees), while 55% work in smaller ones of different sizes. It is possible to observe that they are mainly data scientists, followed by project leaders, developers, and solution architects. Regarding their experience with ML-enabled systems, most of the participants reported having 1 to 2 years of experience. Following closely, another substantial group of participants indicated a higher experience bracket of 3 to 6 years. This distribution highlights a balanced representation of novice and experienced practitioners. Regarding the participants' educational background, 81.38% mentioned having a bachelor's degree in computer science, electrical engineering, information systems, mathematics, or statistics. Moreover, 53.72% held master's degrees, and 22.87% completed Ph.D. programs.

4.2 Model Deployment and Monitoring Relevance and Difficulty

In the survey, we asked about the perceived relevance and difficulty of each ML life cycle stage. In this paper, we focus on the monitoring and deployment life cycle phases.

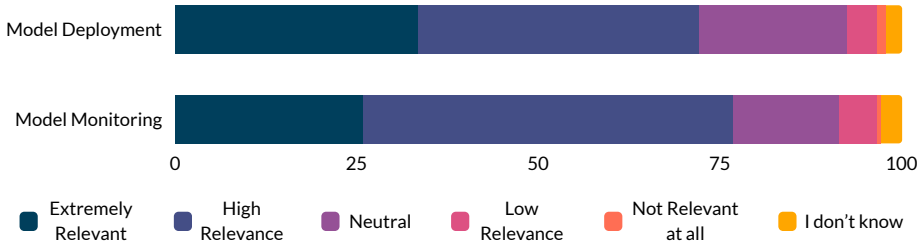

Fig. 2. Relevance of Model Deployment and Model Monitoring activities according to survey participants

The relevance evaluation in Figure 2 shows that the majority of respondents perceive these activities as highly to extremely relevant; it signifies the critical role they play in the ML life cycle.

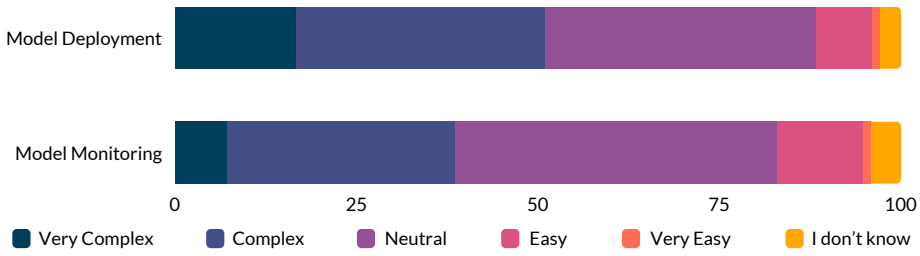

Fig. 3. Difficulty of Model Deployment and Model Monitoring activities according to survey participants

Regarding difficulty, Figure 3 shows that in the perception of the practitioners, the balance mainly swings towards complexity. Of course, the varying perceptions of difficulty could be due to the use of different strategies and solutions for model deployment and monitoring.

4.3 What are contemporary practices for deployment? (RQ1)

[RQ1.1] What kind of approaches are used to deploy ML models? For the first question of the survey regarding deployment, the participants were asked about which approach they usually take for hosting their models, as shown

in Figure 4, where respondents could select more than one option. For the most part, *Service* was the top choice with **P = 59.457 [59.219, 59.695]**, followed by *Embedded Models* with **P = 42.719 [42.476, 42.962]** and *PaaS* with **P = 23.826 [23.628, 24.024]**. Other solutions were also opened for answers and grouped in *Others* with **P = 5.47 [5.359, 5.58]**.

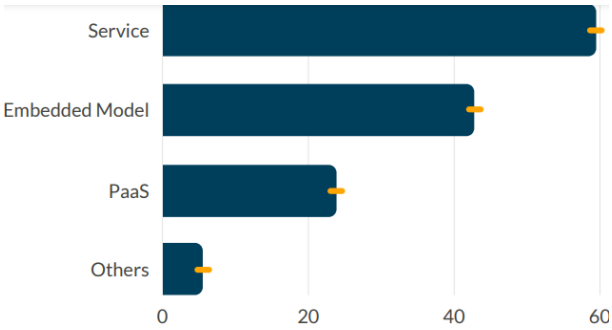

Fig. 4. Percentage of deployment approaches used by survey participants (N=168)

[RQ1.2] Which tools are used for automating model retraining? and [RQ1.3] What are the MLOps practices and principles used? To describe the usage of MLOps in the life cycle, we asked if the respondents' organizations follow any of the practices or principles, such as having an automated pipeline to retrain and deploy ML models. The results are summarized in Figure 5. The majority answered *No* with **P = 70.911 [70.694, 71.128]** and, followed by *Yes* with **P = 29.089 [28.872, 29.306]**. With regards to the free text field on their MLOps approach, some of the answers were between having their own pipeline built on top of a continuous delivery tool (e.g. Gitlab CI/CD and Azure DevOps) and ML-specific development platforms such as BentoML, MLFlow, and AWS Sagemaker MLOps.

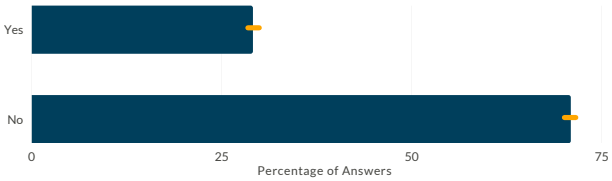

Fig. 5. Answers regarding the survey participant's organization usage of MLOps principles (N=168)

4.4 What are the main problems faced during the deployment in the ML life cycle stage? (RQ2)

The survey had two open-text questions regarding the main problems faced by practitioners through the deployment and monitoring of models. Figure 6 presents the coded answers for the deployment phase.

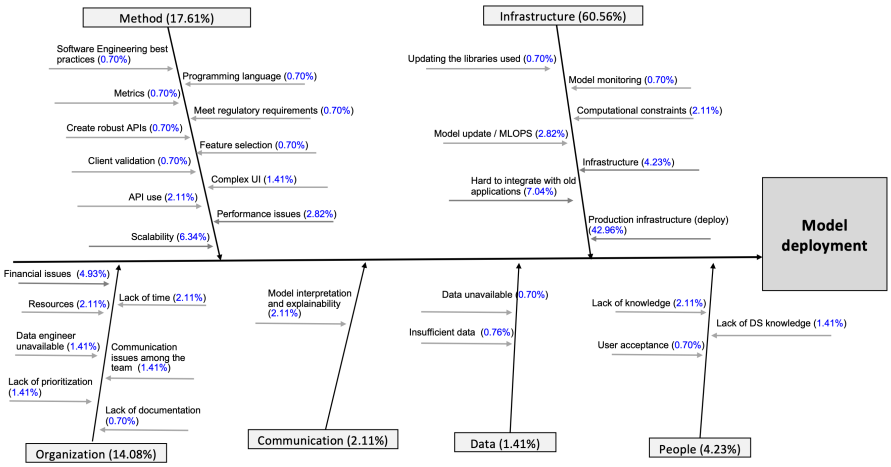

Fig. 6. Fishbone diagram of main problems faced during the model deployment stage

As per the survey respondents, the top problems faced within the deployment phase were preparing the infrastructure for production deployment, the difficulty of integrating with legacy applications, what infrastructure architecture to use, how to scale it, and the financial limitations.

4.5 What are contemporary practices for monitoring? (RQ3)

[RQ3.1] What percentage of the ML-enabled system projects that get deployed into production have their ML models actually being monitored? To evaluate if the deployed projects went through the whole life cycle up until getting monitored, Figure 7 shows that $\mathbf{P = 33.079}$ **[32.842, 33.316]** participants responded that less than 20% of projects do get into production with their aspects monitored, followed by $\mathbf{P = 21.143}$ **[20.942, 21.344]** responding from 20% to 40%, $\mathbf{P = 19.13}$ **[18.943, 19.317]** answering that 80% to 100%, $\mathbf{P = 18.64}$ **[18.456, 18.824]** from 40% to 60% and, finally, $\mathbf{P = 8.009}$ **[7.874, 8.144]** with 60% to 80% getting the released project somehow monitored. Hence, monitoring ML models, which reflects organizational MLOps maturity [6], is still not commonplace.

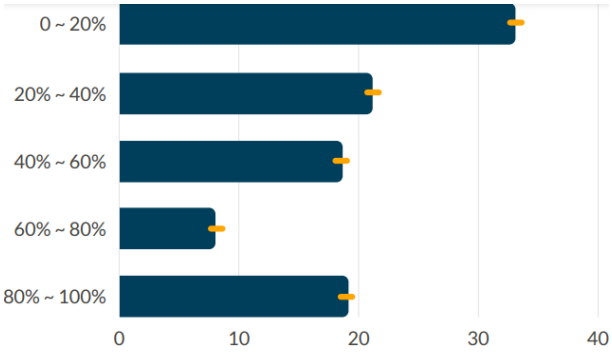

Fig. 7. Percentage of answers for models, deployed to production, that have their aspects monitored (N=160)

[RQ3.2] What aspects of the models are monitored? Respondents described which aspects were actually monitored as in Figure 8. Participants could be selecting more than one option, having *Input and Output* as the most frequent response with **P = 62.675 [62.431, 62.918]**, followed by *Output and Decisions* with **P = 62.082 [61.834, 62.331]**, *Interpretability Output* with **P = 28.034 [27.805, 28.263]**, *Fairness* with **P = 12.965 [12.792, 13.138]**, and other aspects that were grouped in *Others* with **P = 5.874 [5.761, 5.987]**.

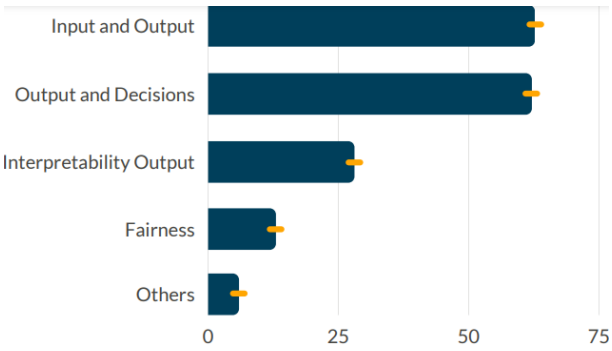

Fig. 8. Percentage of answers regarding which of the ML system aspects are monitored (N=153)

4.6 What are the main problems faced during the monitoring in the ML life cycle stage? (RQ4)

Just as with **RQ2**, Figure 9 contains the coded answers for the problems faced in the monitoring phase.

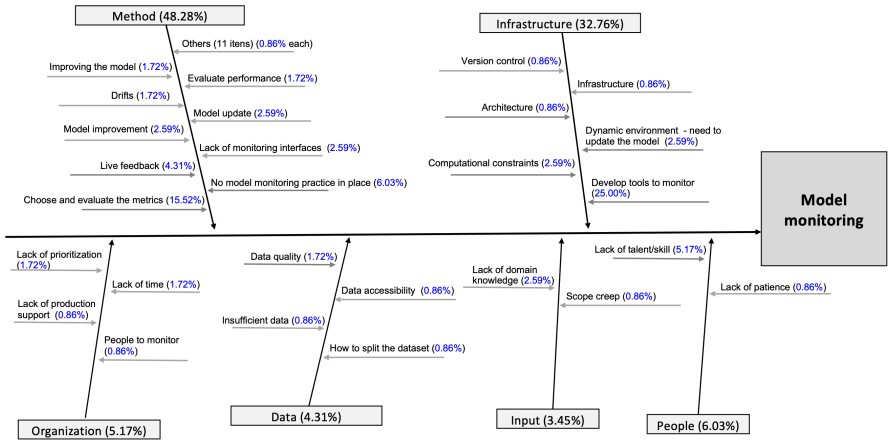

Fig. 9. Fish-bone graphic related to answers regarding the main problems faced during the model monitoring stage

Here, the most concerns were related to the need of developing their own monitoring tools, evaluating and choosing the appropriate metrics, and not having any practice to monitor in place.

4.7 What is the percentage of projects that do go into production? (RQ5)

To describe the population of projects that live up until their general release, data from the demographic questions D7 and D8 (after data cleaning) were combined into Figure 10. As this figure shows, $\mathbf{P = 24.965}$ **[24.759, 25.171]** participants responded that between only 0% to 20% projects went into production, followed by $\mathbf{P = 23.553}$ **[23.337, 23.768]** saying 40% to 60%, then $\mathbf{P = 21.221}$ **[21.029, 21.412]** with 80% to 100%, $\mathbf{P = 17.796}$ **[17.618, 17.974]** saying 20% to 40% and, finally $\mathbf{P = 12.465}$ **[12.306, 12.624]** responding with 60% to 80%. In total, an average of only 45.41% of executed projects go into production.

5 Discussion

Deploying Machine Learning models into production environments can be a complex and challenging task, often accompanied by several problems and considerations. As observed by the survey results, the model deployment and monitoring phases are found to be relevant by almost 75% of respondents, corroborating the importance of releasing models to the public and the constant performance monitoring for avoiding model performance decay and heading towards a continuous increase in quality.

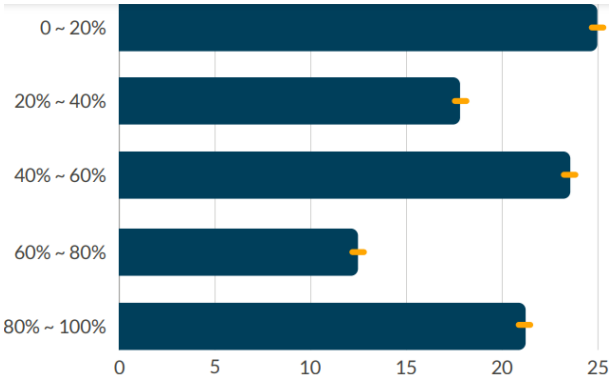

Fig. 10. The percentage of ML projects that do go into production (N=169)

Through the deployment practices identified, it is evident that ML engineers are deploying most of their models to be served as separate services, identifying a growing reliance on cloud-based services that offer comprehensive and scalable solutions. As per the identified lack of MLOps practices used, participants answered that less than 30% apply some of its principles. This suggests that despite the growing importance of ML in various industries, a significant number of professionals may not be fully engaged with MLOps, although numerous studies have proven its benefits [27] and provided guidance on establishing MLOps practices [28]. Although not fully applied, some of the practices do come embedded in ready-to-use platforms, which were also mentioned in the survey.

Regarding the main deployment problems encountered, as per Figure 6, issues include production infrastructure management and integration with legacy systems. Nahar *et al.* [2] conducted a literature review of challenges in building ML-enabled systems. They revealed similar results related to deployment. The main challenges encountered include the shift from model-centric to pipeline-driven developments, difficulties in scaling model training and deployment on different types of hardware, and limited technical support for engineering infrastructure. Similarly, Sculley *et al.* [29] pointed out the complexity of the infrastructure surrounding ML code.

For the monitoring practices, the survey highlights that the number of models that do go into production and have their aspects monitored is less than 50%, which highlights the potential of further supporting monitoring. When participants were asked which aspects were monitored, inputs, outputs, and decisions stood out. Monitoring inputs and outputs emphasizes the critical role of data quality in the performance of ML systems, given that data inconsistencies could impact the accuracy and reliability of model predictions. Monitoring the decisions allows for assessing the correctness and effectiveness of model predictions to validate the alignment between what was predicted and real-world outcomes. Furthermore, the monitoring of interpretability output emerges as another prominent aspect, highlighting the increasing focus on enhancing the transparency and explainability of ML models. Fairness aspects are still rarely

monitored, a scenario that might change with a growing recognition of the potential ethical implications of ML algorithms.

The most prominent reported problems for model monitoring, as per Figure 9, include having to develop new tools for infrastructure monitoring necessities and difficulties in choosing appropriate metrics. Again, the findings by Nahar *et al.*, which include the lack of support for setting up the infrastructure and difficulties in defining metrics, are aligned with the participants' perceptions.

It is noteworthy that our study revealed that less than 50% of projects make it into production, still showing a standing pattern from earlier studies [30, 31] and books [32], which also identified that most of the ML projects fail to get generally available due to several problems.

6 Threats to Validity

We identified some threats while planning, conducting, and analyzing the survey results. Hereafter, we list the most prominent threats organized by the survey validity types presented in [33].

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

Criterion Validity. Threats to criterion validity include not surveying the target population. We clarified the target population in the consent form (before starting the survey). We also considered only complete answers (*i.e.*, answers of participants that answered all survey sections) and excluded participants that informed having no experience with ML-enabled system projects.

Construct Validity. We ground our survey's questions and answer options on theoretical background from previous studies (*e.g.*, [34, 23]) and readings based on identified challenges in model deployment and monitoring (*e.g.*, [5]). A threat to construct validity is inadequate measurement procedures and unreliable results. To mitigate this threat, we follow recommended data collection and analysis procedures [21].

Reliability. One aspect of reliability is statistical generalizability. We could not construct a random sample systematically covering different types of professionals involved in developing ML-enabled systems, and there is yet no generalized knowledge about what such a population looks like. Furthermore, as a consequence of convenience sampling, the majority of answers came from Europe and South America. Nevertheless, the experience and background profiles of the subjects are comparable to the profiles of ML teams as shown in Microsoft's study [35], indicating that the nationality attribute did not interfere with the results. To deal with the random sampling limitation, we used bootstrapping and only employed confidence intervals, conservatively avoiding null hypothesis testing. Another reliability aspect concerns inter-observer reliability, which we

improved by including independent peer review in all our qualitative analysis procedures and making all the data and analyses openly available online.

7 Conclusion

The current study sought to provide a comprehensive overview of the prevailing practices and challenges in model deployment and monitoring within the context of ML-enabled systems. Through our questionnaire-based online survey targeting practitioners, answered by 188 practitioners, we identified several key insights.

Regarding the deployment of models, our observations indicate an increasing approach of deploying models as separate cloud-based services, with less frequent exploration of embedding models within consuming applications or platform-as-a-service solutions. A significant number of practitioners deviate from MLOps principles, lacking automated pipelines for model retraining and redeployment. Deployment challenges reported encompass difficulties in architecting production infrastructure, considering scalability and financial constraints, and integrating the model with legacy applications.

As for model monitoring, a notable finding is that a substantial portion of models in production lack monitoring altogether. The primary focus of monitoring lies in outputs and decisions. Challenges reported in this context include the absence of model-appropriate monitoring practices, the necessity to develop customized monitoring tools, and difficulties in selecting suitable metrics.

Future research endeavors could focus on the development of robust and scalable deployment frameworks that accommodate a wide range of ML models and their applications, focusing on infrastructure management and seamless integration with other services. Additionally, there is a need to advance methodologies for comprehensive and real-time monitoring through incisive metrics, enabling stakeholders to proactively identify and address potential biases, vulnerabilities, and performance bottlenecks in ML models.

References

1. Giray, G.: A software engineering perspective on engineering machine learning systems: State of the art and challenges. Journal of Systems and Software **180** (2021) 111031
2. Nahar, N., Zhang, H., Lewis, G., Zhou, S., Kastner, C.: A meta-summary of challenges in building products with ml components – collecting experiences from 4758+ practitioners. In: 2023 IEEE/ACM 2nd International Conference on AI Engineering – Software Engineering for AI (CAIN), Los Alamitos, CA, USA, IEEE Computer Society (may 2023) 171–183
3. Alves, A.P.S., Kalinowski, M., Giray, G., Mendez, D., Lavesson, N., Azevedo, K., Villamizar, H., Escovedo, T., Lopes, H., Biffl, S., et al.: Status quo and problems of requirements engineering for machine learning: Results from an international survey. In: International Conference on Product-Focused Software Process Improvement, Springer (2023) 159–174

4. Up, S.: Machine learning history: The complete timeline (September 2022)
5. Paleyes, A., Urma, R.G., Lawrence, N.D.: Challenges in deploying machine learning: A survey of case studies. ACM Comput. Surv. **55**(6) (dec 2022)
6. John, M.M., Olsson, H.H., Bosch, J.: Towards mlops: A framework and maturity model. In: 2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA). (2021) 1–8
7. Lewis, G.A., Ozkaya, I., Xu, X.: Software architecture challenges for ml systems. In: 2021 IEEE International Conference on Software Maintenance and Evolution (ICSME). (2021) 634–638
8. John, M.M., Olsson, H.H., Bosch, J.: Ai deployment architecture: Multi-case study for key factor identification. In: 2020 27th Asia-Pacific Software Engineering Conference (APSEC). (2020) 395–404
9. John, M.M., Holmström Olsson, H., Bosch, J.: Architecting ai deployment: A systematic review of state-of-the-art and state-of-practice literature. In Klotins, E., Wnuk, K., eds.: Software Business, Cham, Springer International Publishing (2021) 14–29
10. Zaharia, M., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M.J., Ghodsi, A., Gonzalez, J., Shenker, S., Stoica, I.: Apache spark: A unified engine for big data processing. Commun. ACM **59**(11) (oct 2016) 56–65
11. Syafrudin, M., Alfian, G., Fitriyani, N.L., Rhee, J.: Performance analysis of iot-based sensor, big data processing, and machine learning model for real-time monitoring system in automotive manufacturing. Sensors **18**(9) (2018)
12. Chahal, D., Ojha, R., Ramesh, M., Singhal, R.: Migrating large deep learning models to serverless architecture. (2020) 111 – 116 Cited by: 14.
13. Nowrin, I., Khanam, F.: Importance of cloud deployment model and security issues of software as a service (saas) for cloud computing. In: 2019 International Conference on Applied Machine Learning (ICAML). (2019) 183–186
14. Mrozek, D., Koczur, A., Małysiak-Mrozek, B.: Fall detection in older adults with mobile iot devices and machine learning in the cloud and on the edge. Information Sciences **537** (2020) 132 – 147 Cited by: 72; All Open Access, Hybrid Gold Open Access.
15. Abdelaziz, A., Elhoseny, M., Salama, A.S., Riad, A.: A machine learning model for improving healthcare services on cloud computing environment. Measurement **119** (2018) 117–128
16. Garg, S., Pundir, P., Rathee, G., Gupta, P., Garg, S., Ahlawat, S.: On continuous integration / continuous delivery for automated deployment of machine learning models using mlops. In: 2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE). (2021) 25–28
17. Al-Doghman, F., Moustafa, N., Khalil, I., Sohrabi, N., Tari, Z., Zomaya, A.Y.: Ai-enabled secure microservices in edge computing: Opportunities and challenges. IEEE Transactions on Services Computing **16**(2) (2023) 1485–1504
18. Paraskevoulakou, E., Kyriazis, D.: Ml-faas: Towards exploiting the serverless paradigm to facilitate machine learning functions as a service. IEEE Transactions on Network and Service Management (2023) 1–1
19. Kourouklidis, P., Kolovos, D., Noppen, J., Matragkas, N.: A model-driven engineering approach for monitoring machine learning models. In: 2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C). (2021) 160–164
20. Schröder, T., Schulz, M.: Monitoring machine learning models: a categorization of challenges and methods. Data Science and Management **5**(3) (2022) 105–116

21. Wagner, S., Mendez, D., Felderer, M., Graziotin, D., Kalinowski, M.: Challenges in survey research. Contemporary Empirical Methods in Software Engineering (2020) 93–125
22. Lunneborg, C.E.: Bootstrap inference for local populations. Therapeutic Innovation & Regulatory Science **35**(4) (2001) 1327–1342
23. Wagner, S., Fernández, D.M., Felderer, M., Vetrò, A., Kalinowski, M., Wieringa, R., Pfahl, D., Conte, T., Christiansson, M.T., Greer, D., Lassenius, C., Männistö, T., Nayebi, M., Oivo, M., Penzenstadler, B., Prikladnicki, R., Ruhe, G., Schekelmann, A., Sen, S., Spínola, R., Tuzcu, A., Vara, J.L.D.L., Winkler, D.: Status quo in requirements engineering: A theory and a global family of surveys. ACM Trans. Softw. Eng. Methodol. **28**(2) (2019)
24. Efron, B., Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman & Hall/CRC (1993)
25. Lei, S., Smith, M.: Evaluation of several nonparametric bootstrap methods to estimate confidence intervals for software metrics. IEEE Transactions on Software Engineering **29**(11) (2003) 996–1004
26. Stol, K.J., Ralph, P., Fitzgerald, B.: Grounded theory in software engineering research: a critical review and guidelines. In: Proceedings of the 38th International Conference on Software Engineering. (2016) 120–131
27. Ruf, P., Madan, M., Reich, C., Ould-Abdeslam, D.: Demystifying mlops and presenting a recipe for the selection of open-source tools. Applied Sciences **11**(19) (2021)
28. Zhou, Y., Yu, Y., Ding, B.: Towards mlops: A case study of ml pipeline platform. In: 2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE). (2020) 494–500
29. Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.F., Dennison, D.: Hidden technical debt in machine learning systems. Advances in neural information processing systems **28** (2015)
30. algorithmia: 2020 state of enterprise machine learning. Technical report (2019)
31. Siegel, E.: Models are rarely deployed: An industry-wide failure in machine learning leadership (January 2022)
32. Weiner, J.: Why AI/Data Science Projects Fail: How to avoid project pitfalls. Morgan; Claypool Publishers (2021)
33. Linaker, J., Sulaman, S.M., Höst, M., de Mello, R.M.: Guidelines for conducting surveys in software engineering v. 1.1. Lund University **50** (2015)
34. Fernández, D.M., Wagner, S., Kalinowski, M., Felderer, M., Mafra, P., Vetrò, A., Conte, T., Christiansson, M.T., Greer, D., Lassenius, C., et al.: Naming the pain in requirements engineering: Contemporary problems, causes, and effects in practice. Empirical Software Engineering **22** (2017) 2298–2338
35. Kim, M., Zimmermann, T., DeLine, R., Begel, A.: Data scientists in software teams: State of the art and challenges. IEEE Transactions on Software Engineering **44**(11) (2017) 1024–1038