Pontifícia Universidade Católica
do Rio de Janeiro

Eduardo Zimelewicz

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

Dissertação de Mestrado

Dissertation presented to the Programa de Pós-graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática.

Advisor: Prof. Marcos Kalinowski

Rio de Janeiro
June 2024

Pontifícia Universidade Católica
do Rio de Janeiro

Eduardo Zimelewicz

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

Dissertation presented to the Programa de Pós-graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática. Approved by the Examination Committee:

Prof. Marcos Kalinowski
Advisor
Departamento de Informática – PUC-Rio

Prof. Daniel Mendez
BTH

Prof. Hélio Côrtes Vieira Lopes
Departamento de Informática – PUC-Rio

Rio de Janeiro, June 14th, 2024

Eduardo Zimelewicz

Graduated in Computer Science by the Fluminense Federal University.

Bibliographic data

Zimelewicz, Eduardo

ML-Enabled Systems Model Deployment and Monitoring: Status Quo and Problems / Eduardo Zimelewicz; advisor: Marcos Kalinowski. – 2024.

51 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2024.

Inclui bibliografia

1. Informática – Teses. 2. Aprendizado de Máquina. 3. Implantação. 4. Monitoramento. 5. Survey. I. Kalinowski, Marcos. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

To my wife, my family, and friends for their support
and encouragement.

Acknowledgments

To my wife Cecília, whose work to support me in the hardest moments inspires me every day.

To my family and friends, through understanding my absence in precious moments to finish this work.

To my advisor Professor Marcos Kalinowski for the stimulus and partnership to carry out this work.

To CNPq and PUC-Rio, for the aids granted, without which this work does not could have been accomplished.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Zimelewicz, Eduardo; Kalinowski, Marcos (Advisor). **ML-Enabled Systems Model Deployment and Monitoring: Status Quo and Problems**. Rio de Janeiro, 2024. 51p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

[Context] Systems that incorporate Machine Learning (ML) models, often referred to as ML-enabled 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 life cycle phases. [Method] We conducted an international survey to gather practitioner insights on how ML-enabled systems are engineered. We gathered a total of 188 complete responses from 25 countries. We analyze the status quo and problems reported for the model deployment and monitoring phases. We analyzed contemporary practices using bootstrapping with confidence intervals and conducted qualitative analyses on the reported problems applying open and axial coding procedures. [Results] Practitioners perceive the model deployment and monitoring phases as relevant and 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 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 the selection of suitable metrics. [Conclusion] Our results help provide a better understanding of the adopted practices and problems in practice and support guiding ML deployment and monitoring research in a problem-driven manner.

Keywords

Machine Learning; Deployment; Monitoring; Survey.

Resumo

Zimelewicz, Eduardo; Kalinowski, Marcos. **Implantação e monitoramento de modelos de sistemas de aprendizado de máquina: status quo e problemas**. Rio de Janeiro, 2024. 51p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

[Contexto] Sistemas que incorporam modelos de aprendizado de máquina (ML), muitas vezes chamados de sistemas de software habilitados para ML, tornaram-se comuns. No entanto, as evidências empíricas sobre como os sistemas habilitados para ML são projetados na prática ainda são limitadas; isto é especialmente verdadeiro para atividades relacionadas à disseminação do modelo de ML. [Objetivo] Investigamos práticas industriais contemporâneas e problemas relacionados à disseminação de modelos de ML, com foco nas fases de implantação do modelo e no monitoramento dentro do ciclo de vida de ML. [Método] Realizamos uma pesquisa on-line baseada em questionário internacional para coletar informações de profissionais sobre como os sistemas habilitados para ML são projetados. Reunimos 188 respostas completas de 25 países. Analisamos o status quo e os problemas relatados nas fases de implantação e monitoramento do modelo. Realizamos análises estatísticas sobre práticas contemporâneas utilizando bootstrapping com intervalos de confiança e análises qualitativas sobre os problemas relatados envolvendo procedimentos de codificação aberta e axial. [Resultados] Os profissionais consideram as fases de implantação e monitoramento do modelo relevantes, mas também difíceis. No que diz respeito à implantação de modelos, os modelos são normalmente implantados como serviços separados, com adoção limitada dos princípios de MLOps. Os problemas relatados incluem dificuldades no projeto da arquitetura da infraestrutura para implantação de produção e integração de aplicativos legados. No que diz respeito ao monitoramento de modelos, muitos dos modelos em produção não são monitorados. Os principais aspectos monitorados são insumos, produtos e decisões. Os problemas relatados envolvem a ausência de práticas de monitoramento, a necessidade de criar ferramentas de monitoramento personalizadas e desafios na seleção de métricas adequadas. [Conclusão] Nossos resultados já ajudam a fornecer uma melhor compreensão das práticas e problemas adotados na prática que apoiam a

pesquisa em implantação de ML e monitoramento de maneira orientada a problemas.

Palavras-chave

Aprendizado de Máquina; Implantação; Monitoramento; Survey.

Table of contents

List of figures

List of tables

List of algorithms

List of codes

List of Abreviations

ML – *Machine Learning*

AI – *Artificial Intelligence*

SE – *Software Engineering*

MLOps – *Machine Learning Operations*

SLR – *Systematic Literature Review*

GLR – *Grey Literature Review*

R&D – *Research and Development*

BTH – *Blekinge Institute of Technology*

UCLM – *Universidad de Castilla-La Mancha*

Ph.D. – *Doctor of Philosophy*

QAs – *Quality Attributes*

FaaS – *Function as a Service*

IaaS – *Infrastructure as a Service*

PaaS – *Platform as a Service*

SaaS – *Software as a Service*

CI – *Continuous Integration*

CD – *Continuous Deployment*

DevOps – *Development Operations*

RQ1-RQ5 – *Research Question*

D1-D7 – *Demographic Survey Question*

Q1-Q17 – *Open Text Survey Question*

PUC-Rio – *Pontifical Catholic University of Rio de Janeiro*

ExACTa – *Experimentação Ágil. Cocriação. Transformação Digital.*

N – *Survey Population*

n – *Number of valid answers*

P – *Survey Population Proportion*

S – *Bootstrap Resamples*

CRISP-DM – *Cross Industry Standard Process for Data Mining*

et. al. – *et alia - and others*

i. e. – *id est - that is*

Now I'm a pretty lazy person and am prepared to work quite hard in order to avoid work.

Martin Fowler, *Refactoring: Improving the Design of Existing Code.*

1
Introduction

1.1
Context and Motivation

In recent years, the advancements in Machine Learning (ML) and, altogether, Artificial Intelligence (AI), have helped the incoming of technological innovation and transformation across various industries. In this case, systems composed of infrastructure and applications that incorporate these Machine Learning algorithms, by leveraging data to automatically learn and improve its activities, are called ML-enabled systems. 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 in production operations. At the same level, Software Engineering (SE) practices can help to speed up the development of such features. Nevertheless, ML-enabled systems are inherently different by nature, which affects rendering traditional SE practices insufficient to be directly applied, thus, revealing new challenges (NAHAR et al., 2023).

The use of Machine Learning in practical applications dates back to the year 1952 when English mathematician Arthur Samuel created the first Machine Learning program to play the championship-level game of checkers (UP, 2022). 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.* (PALEYES; URMA; LAWRENCE, 2022), especially in the fields of integration, monitoring, and updating a model. Further discussions show that, within the model deployment phase (which includes the monitoring part), adapting existing techniques such as DevOps could be extremely helpful to make development and production environments even closer, where the term MLOps follows the same concept by bringing together data scientists and operations teams, where Meenu *et. al.* (JOHN; OLSSON; BOSCH, 2021b) provided a work on identifying the activities and placing the development stages, by conducting a

systematic literature review (SLR) and grey literature review (GLR), in which organizations can improve their MLOps adoption.

Regarding 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. More specifically understanding why systems are built the way they are, what is the life cycle followed, and why production monitoring and deploying are still far away from most teams in the industry. To understand those problems, concerns are identified in two important stages in the life cycle, the monitoring and deployment phases, where this dissertation deep dives to bring forward the status quo and problems for insights that could help the development performance and experience.

1.2 Goal and Research Method

With the main goal of understanding the pain points of how those systems are built, we conducted a questionnaire-based online survey. Although many other concerns appeared in the responses, such as issues in Requirements Engineering and Data Quality (ALVES et al., 2023), the work presented in this dissertation focuses on the model deployment and monitoring of ML-enabled systems. Our focus is on evaluating experienced challenges as well as approaches employed.

For research questions that seek to identify the main problems faced by practitioners involved in engineering ML-enabled systems, specifically in the model deployment and monitoring phases, alongside questions regarding which current practices are being applied and what amount of models are generally available, we had their corresponding survey question designed to be open text. We also conducted a qualitative analysis using open and axial coding procedures from grounded theory (STOL; RALPH; FITZGERALD, 2016) 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 Ph.D. student, reviewed by her advisor at PUC-Rio, and reviewed independently by three researchers from two additional sites (two from BTH Sweden and one independent researcher from Turkey).

1.3 Summary of the Findings

The main findings show that practitioners perceive the model deployment and monitoring phases as relevant but also challenging. Concerning model de-

ployment, 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 is 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 the industry, what appropriate practices could be, and how they can improve production deployment.

1.4 Dissertation Outline

The remainder of this dissertation is organized as follows.

Chapter 2 provides the background to the dissertation and the construction of the survey, alongside the discovered related work that serves as a basis for investigation.

Chapter 3 describes the research method by defining the main goal alongside the description of the research questions for guiding the dissertation. Then, presenting the survey design steps followed, and the data analysis procedures used.

Chapter 4 presents the results with a graphical reference to the collected data regarding respondents' demographics and the survey-related questions to support the answering of the presented research questions.

Chapter 5 discusses the results further by relating the main findings to existing evidence in the literature, and where we also critically reflect upon the threats to validity and its mitigation actions.

Lastly, concluding our dissertation with Chapter 6 by presenting the dissertation's contributions to the research, its limitations, and future work.

2
Background and Related Work

2.1
Introduction

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. In this chapter, we describe the background for this dissertation, which comprises understanding the machine learning life cycle (Section 2.2) and its model deployment and monitoring phases (Section 2.3). Furthermore, we discuss related work providing an overview of the research landscape on model deployment and monitoring practices and challenges (Section 2.4).

2.2
Machine Learning Life Cycle

Before we delve into the related work, it is important to understand and describe what is a software development life cycle, defined as a structured set of phases that comprises the construction of a software system. In the case of Machine Learning development, it differentiates from other systems in a way that it should be constantly revisiting its training data, in a feedback loop, to better fine-tune its model predictions and improve its responses. A visual representation of the structure is presented by Amershi *et. al.* (AMERSHI et al., 2019) at 2.1, where it calls attention to the data-centered essence of the process.

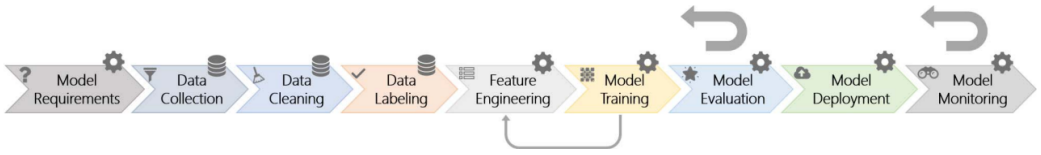

Figure 2.1: The ML life cycle phases as presented by Amershi *et. al.* (AMERSHI et al., 2019). The larger arrows show that Model Evaluation and Model Monitoring may expose information that could loop back to the earlier stages for continuous improvement. While the single arrow from Model Training only loops back to Feature Engineering as per their constant nature of feature data modifications and training.

With the stage's representation, and the positioning of the Model Deployment and Monitoring as important paths of feedback loops, this dissertation has the important task of shedding light on the operations stage of the machine learning system. Showing their importance for continuous improvement processes and how practitioners can benefit from evolving industry applying techniques.

2.3 Machine Learning Model Deployment and Monitoring

Those two stages of the ML life cycle are put as the last phase to make a model available. Their absence could make a system unusable and harder to determine its execution performance and to detect optimization parts. For this, Model Deployment is defined as the process of integrating a trained machine learning model into an existing production environment where it can take in input and return output. Some of the tasks involved are:

- **Integration**: The model is integrated into the production environment, which could be a web server, a cloud platform, or an edge device.
- **Testing**: Rigorous testing is performed to ensure the model behaves as expected in the production setting.
- **Scaling**: The deployment setup must be scalable to handle varying loads and performance requirements.
- **Continuous Delivery**: Automating the deployment process to allow for continuous delivery and integration of model updates.

As for Model Monitoring, it is depicted as the practice of tracking the performance of machine learning models in production to identify and address issues that can negatively impact business value. Tasks that are often involved are:

- **Performance Tracking**: Continuously tracking metrics like accuracy, precision, and recall to detect performance degradation.
- **Data Quality**: Monitoring the quality of input data to catch any anomalies or shifts that could affect the model's predictions.
- **Drift Detection**: Identifying changes in data distribution, known as data drift, which can cause the model's performance to decline over time.
- **Bias Detection**: Ensuring the model does not develop or perpetuate bias, which is crucial for ethical AI practices.

– **Alerting**: Setting up alerts to notify relevant stakeholders when certain thresholds are crossed or anomalies are detected.

Their definition, as well as some of the important tasks involved, makes both deployment and monitoring iterative and ongoing processes that are essential to the model operation, which requires collaboration between data scientists, engineers, and business stakeholders to ensure the model remains effective and valuable in a production setting.

2.4 Related Work on Machine Learning Deployment and Monitoring

To represent the main issues to transitioning models to production architectures, some challenges were also identified and categorized by Lewis *et. al.* (LEWIS; OZKAYA; XU, 2021) in four spaces. 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.* (JOHN; OLSSON; BOSCH, 2020). 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 better evaluate systems. Third, monitorability as a driving quality attribute, by having the infrastructure behind the monitoring platform be responsible for collecting specific information related to changes in the dataset, as well as the incorporated user feedback, to observe the impacts to 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, such as the SLR and a GLR conducted by Meenu *et. al.* (JOHN; OLSSON; BOSCH, 2021a), by providing an overview of the AI deployment's status quo and practices reported in the literature to further design a deployment framework for these systems. Today's approaches range from traditional batch processing (ZAHARIA et al., 2016) to real-time streaming deployments (SYAFRUDIN et al., 2018) and, most currently, an increase in the use of the cloud service offerings such as FaaS (Function as a Service) (CHAHAL et al., 2020), SaaS (Software as a Service) (NOWRIN; KHANAM, 2019), PaaS (Platform as a Service) (MROZEK; KOCZUR; MAŁYSIAK-MROZEK, 2020) and IaaS (Infrastructure as a Ser-

vice) (ABDELAZIZ et al., 2018), representing the benefits of cloud adoption by the practitioners such as the relief from the burden of servers' 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 (GARG et al., 2021), micro-services (AL-DOGHMAN et al., 2023), and serverless computing (PARASKEVOULAKOU; KYRIAZIS, 2023) have gained prominence in ensuring model deployment flexibility and scalability.

Recent studies have focused on the monitoring and maintenance of ML models, where researchers have proposed techniques for detecting Machine Learning specific metrics such as model drift, handling concept drift, and ensuring that models remain accurate and reliable over time (KOUROUKLIDIS et al., 2021; SCHRöDER; SCHULZ, 2022), which involves concepts such as statistical process control, anomaly detection, and continuous integration and deployment (CI/CD) practices.

2.5 Concluding Remarks

The presented literature demonstrates the diverse nature of ML deployment and monitoring challenges. While numerous strategies and techniques have been proposed, there remains a need for a better understanding of industrial practices and their related challenges, providing an empirical basis for conducting research in a problem-driven manner. To address the issue of gathering information on industrial practice, an international survey on ML-enabled systems was conducted. In this dissertation, we analyze the data from this survey related to model monitoring and deployment. In the subsequent chapter, we delve into the details of our research method.

3
Research Method

3.1
Introduction

This chapter provides the goal of the dissertation by describing its research questions to gather insights on the challenges and current practices of the ML-enabled systems industry (Section 3.2). In conjunction, showing the survey design method that was put in place for information gathering (Section 3.3). Followed by the explanation of the executed collection (Section 3.4) and analysis of the resulting data, and presenting the connection of the survey question with the created research (Section 3.5).

3.2
Goal and Research Questions

The main goal of the research study focused on surveying the current status quo and problems through the entire development life cycle of an ML system, but for the context of the current dissertation, 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 retraining 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?
  Under this question, we aim to identify the in-use practices and trends of the *deployment* stage, so we can refine it 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?
  Under this question, we aim at identifying the in-use practices and trends of the *monitoring* stage, so we can refine it further into two 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?

3.3 Survey Design

We designed our survey based on best community practices of survey research (WAGNER et al., 2020), carefully conducting, in essence, the following steps:

- **Step 1. Initial Survey Design**. We conducted a literature review on ML deployment and monitoring and combined our findings with previous results on problems and the status quo to provide the theoretical foundations for questions and answer options. From there, we drafted the initial survey by involving Software Engineering and Machine Learning researchers of PUC-Rio (Brazil) with experience in R&D projects involving ML-enabled systems.
- **Step 2. Survey Design Review**. The survey was reviewed and adjusted based on online discussions and annotated feedback from Software Engineering and Machine Learning researchers of BTH (Sweden). Thereafter, the survey was also reviewed by the other co-authors.
- **Step 3. Pilot Face Validity Evaluation**. This evaluation involves a lightweight review by randomly chosen respondents. It was conducted with 18 Ph.D. students taking a Survey Research Methods course at UCLM (Spain) taught by the second 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 is 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 dissertation focuses on the ML life cycle problems related to model deployment and aspects of monitoring, and the specific questions regarding problem motives. The excerpts of the questions we deem relevant in the context of the dissertation at hand are shown in Table 3.1. The survey was implemented using the Unipark Enterprise Feedback Suite.

3.4 Data Collection

Our target population concerns professionals involved in building ML-enabled systems, including different activities, such as management, design, and development. Therefore, it includes practitioners in positions such as project leaders, requirements engineers, data scientists, and developers. We used convenience sampling, sending the survey link to professionals active in our partner companies, and also distributed it openly on social media. We excluded participants who informed us 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.

3.5 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 (LUNNEBORG,

Table 3.1: Research questions and survey questions

RQ	Survey 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 each of the seven ML life cycle stages.	Open
RQ4	Q4	According to your personal experience, please outline the main problems or difficulties (up to three) faced during each of the seven ML life cycle stages.	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 Free Field
RQ1.2	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 Free Field
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 Free Field
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 Free Field
-	...	...	...

2001; WAGNER et al., 2020). Hence, we use bootstrapping to calculate confidence intervals for our results, similar to done in (WAGNER et al., 2019). 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 (EFRON; TIBSHIRANI, 1993), and we set 1000 for S, which is a value that is reported to allow meaningful statistics (LEI; SMITH, 2003).

For the research questions *RQ1.2, RQ1.3, RQ2, RQ3.1, and RQ4*, which seek 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 (STOL; RALPH; FITZGERALD, 2016) 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 Ph.D. student, reviewed by her advisor at PUC-Rio, and reviewed independently by three researchers from two additional sites (two from BTH Sweden and one independent researcher from 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].

3.6 Concluding Remarks

In conclusion, this chapter focused on presenting the research method applied for the creation of this dissertation. Presenting the main goal of understanding the challenges and practices in deploying and monitoring Machine Learning (ML) systems, by describing the research questions we sought to answer.

Then, the survey design steps were defined to enforce its alignment with the current literature. Thereafter, understanding the data collection and analysis procedures for extracting the data points that will support the answers for the research questions, by showing the connection between the specific survey question and the research question.

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

Finally, providing access to the questionnaire and code to recreate the quantitative and qualitative analysis. In the next chapter, we present the results of the designed survey that will serve as the fundamental basis for the dissertation discussion.

4
Results

4.1
Introduction

In this chapter, the survey results are presented in graphical and numerical representations to improve the understanding of the current practices and challenges regarding Model Deployment and Model Monitoring, composed of bar charts and fishbone diagrams. First, the demographics charts are presented (Section 4.2). Followed by the model deployment and model monitoring relevance and difficulty evaluations (Section 4.3). Then, for the research questions in contemporary practices for deployment (Section 4.4), the main problems faced during the deployment phase (Section 4.5), the contemporary practices for monitoring (Section 4.6), the main problems faced during the monitoring phase (Section 4.7), and finishing with what percentage of projects do go into production (Section 4.8).

All of the data that follows the study come with the bootstrapped samples together with the 95% confidence interval. The N in each figure caption is the number of participants that answered this question. We report the proportion P of the participants that checked the corresponding answer and its 95% confidence interval in square brackets.

4.2
Study Population

Figure 4.1 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 diverse countries (Brazil, Turkey, Austria, Germany, Sweden, and Italy).

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

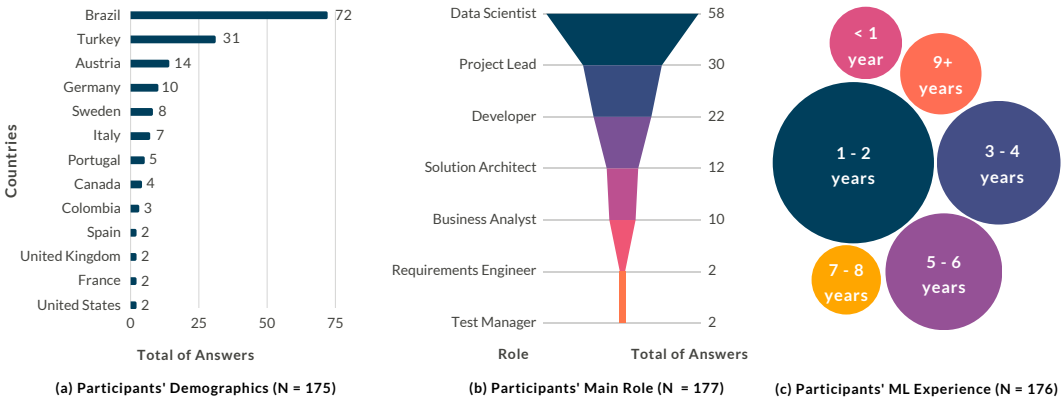

Figure 4.1: Demographic graphs for participant's countries, roles, and ML work experience

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.

4.3 Model Deployment and Monitoring evaluation

In the survey, we used the same abstraction of seven generic life cycle phases of a popular Brazilian textbook on software engineering for data science (KALINOWSKI et al., 2023): problem understanding and requirements, data collection, data pre-processing, model creation and training, model evaluation, model deployment, and model monitoring. These phases were abstracted based on the nine ML life cycle phases presented by Amershi *et al.* (AMERSHI et al., 2019) and the CRISP-DM industry-independent process model phases (SCHRÖER; KRUSE; GÓMEZ, 2021). We asked about the perceived relevance and difficulty of each of the seven phases. For the purpose of this dissertation and the sake of simplicity, we represent only the deployment and monitoring life cycle phases.

The relevance evaluation in Figure 4.2 shows that the majority of respondents view these activities as highly to extremely relevant, it signifies the critical role they play in the software development life cycle, but still open to an increase in their value for projects.

Although respondents find those relevant, it does not necessarily reflect the expectations with the difficulty represented in Figure 4.3, where the

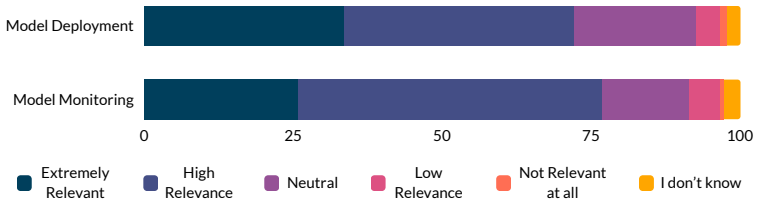

Figure 4.2: Perceived relevance percentages of the Model Deployment and Model Monitoring activities according to survey participants

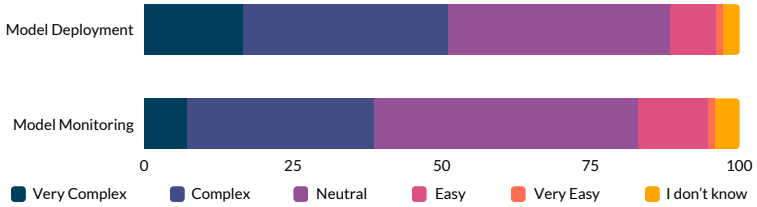

Figure 4.3: Perceived difficulty percentages of Model Deployment and Model Monitoring activities according to survey participants

minority of practitioners find it complex up to very complex, possibly due to the new solutions that come with a complete platform ready to have models deployed and, consequently, getting monitored out of the box to facilitate both of the phases to be applied.

4.4 What are contemporary practices for deployment? (RQ1)

4.4.1 [RQ1.1] What kind of approaches are used to deploy ML models?

For the first question of the survey regarding deployments, the participants were asked about which approach they usually take for hosting their models as shown in Figure 4.4, where respondents could select more than one option. For the most part, *Service* was the top choice with $\mathbf{P = 59.457}$ **[59.219, 59.695]**, followed by *Embedded Models* with $\mathbf{P = 42.719}$ **[42.476, 42.962]** and *PaaS* with $\mathbf{P = 23.826}$ **[23.628, 24.024]**. Other solutions were also opened for answers and grouped in *Others* with $\mathbf{P = 5.47}$ **[5.359, 5.58]**. This demonstrates an increase in industry selection for quicker approaches to make models available, by using a service specific tool or bundling it with existing applications although sacrificing customization of the life cycle stage.

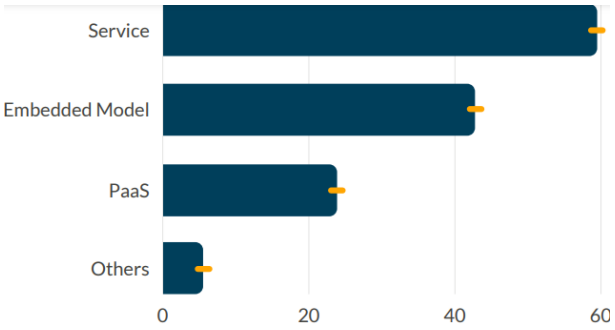

Figure 4.4: Percentage of deployment approaches used by survey participants (N=168)

4.4.2 [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, followed by a follow-up question if a foundational practice, such as an automated retraining pipeline, was used. The results are summarized in Figure 4.5. The majority answered *No* with $\mathbf{P = 70.911}$ **[70.694, 71.128]** and, followed by *Yes* with $\mathbf{P = 29.089}$ **[28.872, 29.306]**. In regards to MLOps, some of the answers were between having their pipeline built on top of a continuous delivery tool (e.g. Gitlab CI/CD (GITLAB, 2023) and Azure DevOps (DEVOPS, 2022)) and Machine Learning specific development platforms such as BentoML (BENTOML, 2023), MLFlow (MLFLOW, 2023) and AWS Sagemaker MLOps (AWS, 2023), which follows practices as model re-training and monitoring of relevant aspects. It signifies an important opportunity to enable MLOps practices adoption to increase the industry development life cycle speed, through automation and structured processes that are proven to be useful.

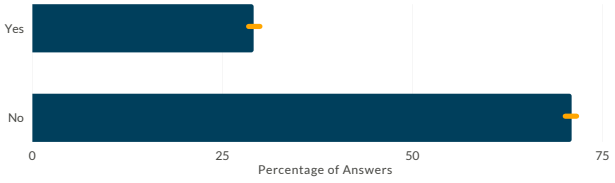

Figure 4.5: Answers regarding the survey participant's organization usage of MLOps principles (N=168)

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

The survey had two questions regarding the main problems faced by practitioners through the deployment and monitoring of models. Figure 4.6 presents the results of the open and axial coding of the answers for the deployment phase using the probabilistic cause-effect diagrams introduced by Kalinowski *et al.* (KALINOWSKI et al., 2010; KALINOWSKI; MENDES; TRAVASSOS, 2011).

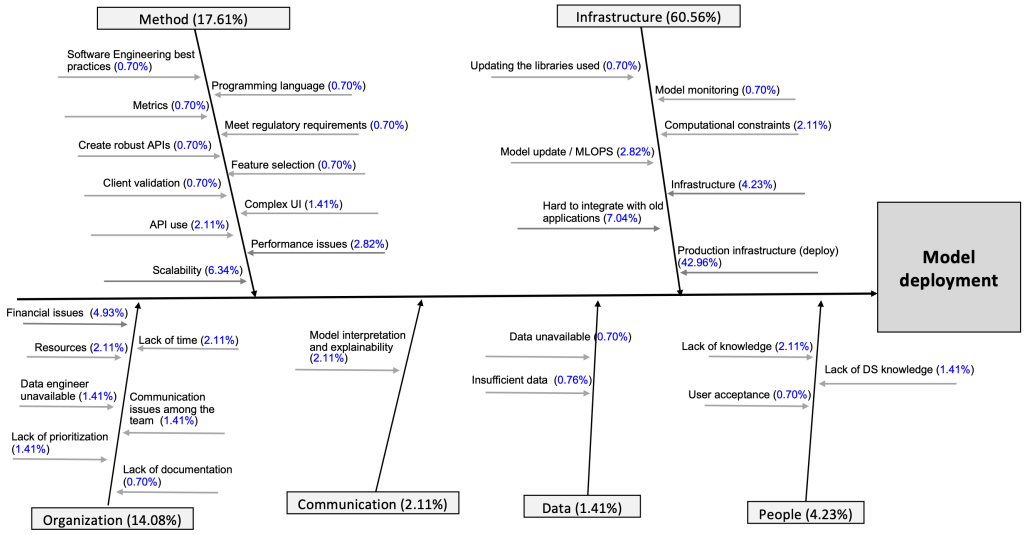

Figure 4.6: Probabilistic cause-effect diagram related to answers regarding the main problems faced during the model deployment stage (N=142)

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. Exposing the lack of expertise among professionals of how ML systems should be deployed, and scaled, with its optimized performance.

4.6 What are contemporary practices for monitoring? (RQ3)

4.6.1 [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 4.7 shows that $\mathbf{P = 33.079}$ $\mathbf{[32.842, 33.316]}$

participants responded that less than 20% of projects do get into production with their aspects monitored, followed by **P = 21.143 [20.942, 21.344]** responding from 20% to 40%, **P = 19.13 [18.943, 19.317]** answering that 80% to 100%, **P = 18.64 [18.456, 18.824]** from 40% to 60% and, finally, **P = 8.009 [7.874, 8.144]** with 60% to 80% get the released project monitored somehow. This also signifies a opportunity to MLOps adoption, where applying its practices includes a feedback loop of monitoring information to continuously improve its model operation.

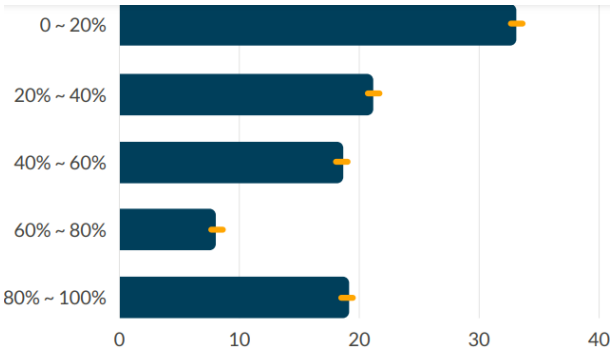

Figure 4.7: Percentage of answers for models, deployed to production, that have their aspects monitored (N=160)

4.6.2 [RQ3.2] What aspects of the models are monitored?

Concerning the model monitoring, respondents described which monitoring aspects were monitored as in Figure 4.8. 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]**. The numbers showcase that monitoring practices are still at the beginning of its full potential, when only the input and output are evaluated where other metrics, or aspects, could be monitored such as the response quality and fairness of the predictions.

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

Figure 4.9 presents the results of the open and axial coding of the answers for the main problems of the monitoring phase.

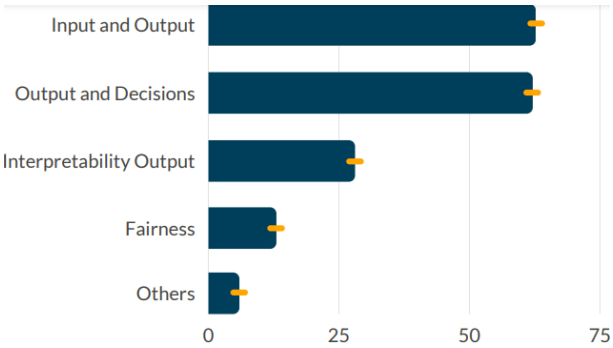

Figure 4.8: Percentage of answers regarding which of the ML system aspects are monitored (N=153)

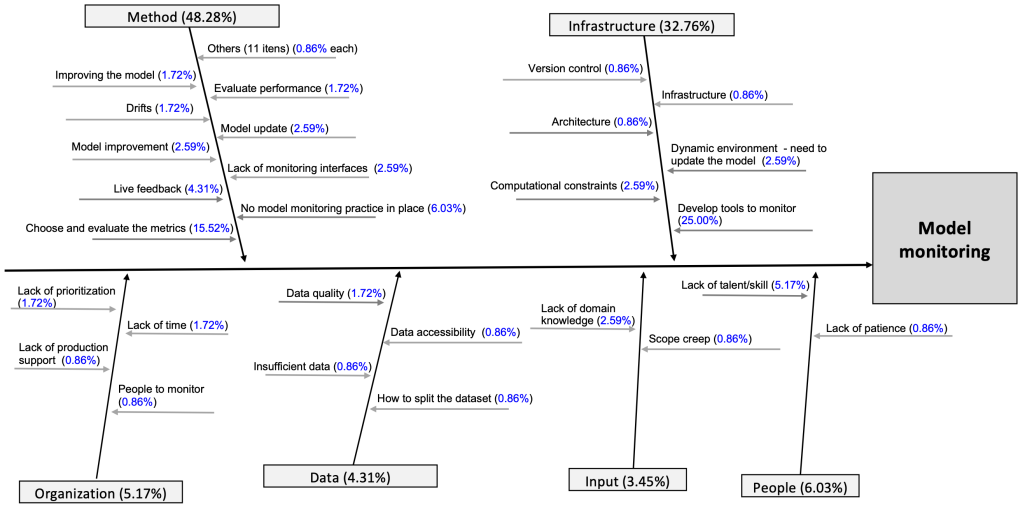

Figure 4.9: Probabilistic cause-effect diagram related to answers regarding the main problems faced during the model monitoring stage (N=116)

Here, the most observed concerns were related to the need to develop their monitoring tools, evaluating and choosing the appropriate metrics, while not having any experience in monitoring models and building monitoring platforms. This showcases the current status of model monitoring, where there is still a long path to adapt existing monitoring tools and frameworks, or create new ones, to solve these novel problems.

4.8 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 4.10. 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 **P = 12.465 [12.306, 12.624]** responding with 60% to 80%. Getting all of the percentages calculated and returning the mean value, leaves us with an average of 45.41% of executed projects reaching general availability. It seriously demonstrated that without the according deployment procedure, the industry still lacks the confidence to move forward on making models available, causing significant blockers to provide new functionality and services for better application experience.

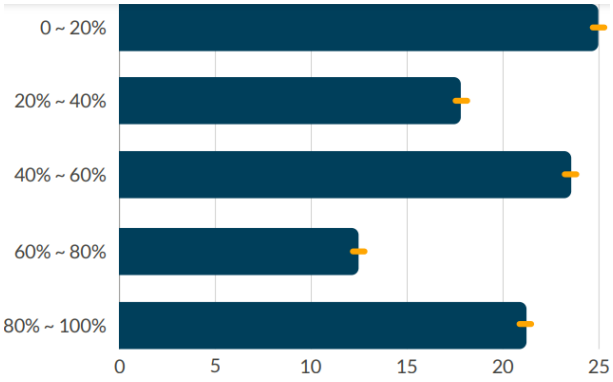

Figure 4.10: The percentage of ML projects that do go into production (N=169)

4.9 Concluding Remarks

This chapter provided an overview of the gathered results of the survey. At first, the sampling technique is explained, and the figure caption to be used is introduced. Beginning with the respondents' demographic charts to identify their origins, then the data representation of the difficulty and relevancy. Followed by the research questions of the contemporary practices for deployment, the main problems encountered, the contemporary practices for monitoring systems, its main problems, and the percentage of projects deployed in production, altogether with its confidence intervals described and briefly discussed. Now, the next chapter enters the discussion regarding the survey results.

5
Discussion and Threats to Validity

5.1
Introduction

This chapter discusses the survey results by comparing the findings to other literature insights regarding the difficulty and relevance of the model deployment and monitor phases, its practices and challenges, and also discussing the issue with the production deploys (Section 5.2). Thereafter, the threats to validity from the dissertation survey are presented alongside its mitigating actions that were applied (Section 5.3).

5.2
Discussion of the Results

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 as well, the model deployment and monitoring phases are found to be relevant by almost 75% of respondents, corroborating the importance of releasing it to the public and the constant performance analysis for a continuous increase in quality. Although to be found important, its difficulty rates decreased to almost 50% for deployment and 30% for monitoring, showing that a lack of opportunity to evaluate a model that is deployed into production could influence the entire development process analysis. For this case, Mäkinen *et. al.* (MäKINEN et al., 2021) surveyed data scientists to observe which type of organization would benefit from the MLOps practices, categorizing some of them as the top beneficiaries where the need for model retraining and deployment were extremely important to their natural next step into production models, showing a potential shift in the evaluation if more automated processes were applied to projects.

Through the deployment practices identified, it is evident that ML engineers are deploying most of their models through the Service approach, identifying a growing reliance on cloud-based services that offer comprehensive and scalable solutions already prepared, but compromising customization. Moreover, if integrating with external systems were found to be hard, the Embedded Model seemed an alternative approach of choice, leveraging the operation efficiency of existing software and faster response times, even though its monitoring and scaling difficulty was increased due to the lack of separation

from the software that includes the model. At last, having the model deployed in a Platform as a Service approach promises to provide full customization of the infrastructure and flexible environment, although the increasing need for specialized expertise to enable its full potential seems important in this approach through such a complex system.

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 Machine Learning in various industries, a significant number of professionals may not be fully engaged with MLOps, although numerous studies have proven its benefits ((RUF et al., 2021),(ARAUJO. et al., 2024)) and guiding on establishing the platform (ZHOU; YU; DING, 2020), unveiling potential research on how MLOps could influence the work of professionals. Although not fully applied, some of the practices do come embedded in ready-to-use platforms, also mentioned in the survey, facilitating the adoption quicker than by creating it from the ground up and seamlessly expanding the usage.

To enforce the main problems encountered as per Figure 4.6, exemplified by this study as issues such as production level infrastructure management and integration with legacy systems, Nahar *et al.* (NAHAR et al., 2023) had a systematic literature review of challenges in building ML components. They revealed similar results related to deployment, the main challenges encountered along shifts 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. For model monitoring, as per Figure 4.9, it shows that choosing the metrics and developing new tools to adequate to project's monitoring necessities are the more prominent problems, where Nahar *et al.* observes that the monitorability of a model being considered late to be implemented, providing data quality due to not having well-supported tools, lack of support to setup an infrastructure for detecting training-serving skew, and difficulty on designing specific metrics are aligned with the participants' feelings within the survey.

For the monitoring aspects, the survey highlights that the number of models that do go into production and have their aspects monitored is less than 50%, which highlights to us the potential of monitorability exploration for identifying aspects, detecting metrics, and creating new tools to increase the quality attributes of ML models. Following the current status of the monitoring phase, when participants were asked which aspects were monitored, input and output data stood out. This emphasizes the critical role of data integrity and quality in the overall performance and robustness of Machine Learning

systems by identifying potential biases, anomalies, and inconsistencies that could impact the accuracy and reliability of model predictions. Furthermore, monitoring the decisions assesses the correctness and effectiveness of model predictions and the process of decision-making to validate the alignment between what was predicted and real-world outcomes. It also shows that the monitoring of interpretability output emerges as another prominent aspect, highlighting the increasing focus on enhancing the transparency and explainability of Machine Learning models, particularly crucial in domains such as establishing trust and verifying model behavior. Lastly, fairness monitoring demonstrates the growing recognition of the ethical implications of algorithms, spurring efforts to monitor and mitigate biases and discriminatory outcomes in model predictions, which underscores the commitment to developing inclusive and equitable Machine Learning systems.

As per Figure 4.10, less than 50% of projects go into production, still showing a standing pattern where earlier reports (ALGORITHMIA, 2019; SIEGEL, 2022) and books (WEINER, 2021), alongside fresh ones (KALINOWSKI et al., 2024), identified that most of the ML projects fail to get generally available due to several problems. Some of those were identified in this study and are possibly related, such as the organization being unable to fit the infrastructure to the needs of engineering teams, financial issues and not having sufficient expertise on the software engineering process that are, most likely, the lack of specialized professionals. As per Figure 4.1, qualified personnel such as Cloud Infrastructure Engineers, Data Engineers, and Software Architects were not significantly identified in the team. However, due to the increasing value given to ML models deployment into production, articles such as Heymann *et. al.* (HEYMANN et al., 2022) will be in evidence to set a common place for frameworks, guides, and books responsible for developing production-level ML models and how to apply them.

5.3 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 (LINAKER et al., 2015).

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 concerning 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 had no experience with ML-enabled system projects.

Construct Validity. We ground our survey's questions and answer options on theoretical background from previous studies (FERNÁNDEZ et al., 2017; WAGNER et al., 2019) and readings based on identified challenges in model deployment and monitoring (PALEYES; URMA; LAWRENCE, 2022) and in software architecture (LEWIS; OZKAYA; XU, 2021). A threat to construct validity is inadequate measurement procedures and unreliable results. To mitigate this threat we follow recommended data collection and analysis procedures (WAGNER et al., 2020).

Reliability. One aspect of reliability is statistical generalizability. We could not construct a random sample 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, most of them from Brazil. Nevertheless, the experience and background profiles of the subjects are comparable to the profiles of ML teams as shown in Microsoft's study (KIM et al., 2017), showing 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.

5.4 Concluding Remarks

In this chapter, the discussion of the research questions was presented by providing a comparison to the literature experience on the status quo and problems. First, deep diving into the difficulty and relevance data showing the possible benefits of applying the MLOps techniques. By entering the deployment statuses, explains the increased utilization of service deployments due to its out-of-the-box solutions that make applications easily available. Then, going back to MLOps, explaining its benefits observed from the literature, and also identifying causes for ML systems not getting the deserved monitoring

structure. With this, the main problems regarding deployment and monitoring phases, detected from the survey, were related to existing issues identified in the literature for validating the dissertation. So, ending with a discussion of the production deployment problem by showing other research data that reinforces the state, but also explaining possible causes and studies that could help practitioners move forward.

Finalizing the chapter with the description of the threats to validity, such as Face and Content Validity, Criterion Validity, Construct Validity, and Reliability that could impact the dissertation, being described and its solving actions explained. The following chapter ends the dissertation with its conclusion.

6
Conclusion

6.1
Contributions

The current study sought to provide a comprehensive overview of the prevailing trends on practices and challenges in model deployment and monitoring within the context of Machine Learning. Through our questionnaire-based online survey targeting practitioners, we identified several key insights allowing us to elaborate as well on potential directions for future research and development. Our analysis underscores the increasing approach to leveraging cloud-based services for model deployment, with a notable emphasis on scalability, accessibility, and seamless integration. This should support the growing demand for efficient and user-friendly deployment solutions, catering to the diverse needs and constraints of contemporary applications.

Furthermore, the emphasis on monitoring aspects reflects the heightened awareness of the critical role played by data quality, model accuracy, and transparency in ensuring the reliability and ethical soundness of Machine Learning models. 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.

The findings presented in this study contribute to the broader discourse surrounding the deployment and monitoring of Machine Learning models, highlighting the significance of holistic and adaptive approaches that prioritize reliability, interpretability, and observability. By leveraging the insights gleaned from this research, stakeholders and practitioners can take their efforts towards the responsible and impactful development of Machine Learning technologies and researchers can better root their ongoing research on practically relevant needs.

Related to this dissertation, we have also submitted a paper that summarizes the results found with the survey. As of March 2024, a preview paper can be downloaded at (ZIMELEWICZ et al., 2024a). Official version is now published in Springer at (ZIMELEWICZ et al., 2024b)

Table 6.1: Publications related to this dissertation

Paper Title	Venue	Status
ML-Enabled Systems Model Deployment and Monitoring: Status Quo and Problems	SWQD 2024	Accepted

6.2 Limitations

In this scope of the dissertation, we did not evaluate the practices using other empirical strategies such as case studies with companies that could benefit from the model deployment and monitoring usage for afterwards evaluation, or focus groups with Machine Learning professionals from PUC-Rio's ExACTa to validate the findings and discuss their meaning.

Additionally, our approach to data collection, utilizing convenience sampling, resulted in a predominance of respondents from the nationalities of the survey's collaborators, notably Brazil, and Turkey. For this, we couldn't have a significant representation of other large countries such as the US, China, and India.

6.3 Future Work

While the current work provides a comprehensive snapshot of the status quo, it also points towards several areas for further investigation and development. The increasing complexity of Machine Learning models and the dynamic nature of real-world applications, necessitate a more nuanced understanding of deployment and monitoring strategies that can adapt to diverse use cases and evolving challenges.

With this, the dissertation should set a base for other empirical strategies to be applied, enriching the information gathered. A case study with a specific company could give a better understanding of how ML deployment and monitoring practices are applied in industrial settings. Consequently, with the survey questionnaire available for replication, the same study can be made with a different population for results comparison or increase the number of answers, composing a bigger view of the status quo.

Future research endeavors could also involve the development of robust and scalable deployment frameworks that accommodate a wide range of ML models and their applications, focusing on better specific infrastructure management and seamless integration with other services.

Additionally, there is a pressing need to advance methodologies for comprehensive and real-time monitoring through incisive metrics discovery and ML-ready monitoring tools, enabling stakeholders to proactively identify and address potential biases, vulnerabilities, and performance bottlenecks in Machine Learning models.

7
Bibliography

ABDELAZIZ, A. et al. A machine learning model for improving healthcare services on cloud computing environment. **Measurement**, v. 119, p. 117–128, 2018. ISSN 0263-2241. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0263224118300228>.

AL-DOGHMAN, F. et al. Ai-enabled secure microservices in edge computing: Opportunities and challenges. **IEEE Transactions on Services Computing**, v. 16, n. 2, p. 1485–1504, 2023.

ALGORITHMIA. **2020 State of Enterprise Machine Learning**. [S.l.], 2019.

ALVES, A. P. S. et al. Status quo and problems of requirements engineering for machine learning: Results from an international survey. In: SPRINGER. **International Conference on Product-Focused Software Process Improvement**. [S.l.], 2023. p. 159–174.

AMERSHI, S. et al. Software engineering for machine learning: A case study. In: IEEE. **2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice**. [S.l.], 2019. p. 291–300.

ARAUJO., G. et al. Professional insights into benefits and limitations of implementing mlops principles. In: INSTICC. **Proceedings of the 26th International Conference on Enterprise Information Systems - Volume 2: ICEIS**. [S.l.]: SciTePress, 2024. p. 305–312. ISBN 978-989-758-692-7. ISSN 2184-4992.

AWS. **Amazon SageMaker for MLOps**. 2023. Disponível em: <https://aws.amazon.com/sagemaker/mlops/?sagemaker-data-wrangler-whats-new.sort-by=item.additionalFields.postDateTime&sagemaker-data-wrangler-whats-new.sort-order=desc>.

BENTOML. **What is BentoML?** 2023. Disponível em: <https://docs.bentoml.org/en/latest/overview/what-is-bentoml.html>.

CHAHAL, D. et al. Migrating large deep learning models to serverless architecture. In: . [s.n.], 2020. p. 111 – 116. Cited by: 14. Disponível em: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85099848738&doi=10.1109%2fISSREW51248.2020.00047&partnerID=40&md5=ade5479a5291ae6c5742fdd38d779925>.

DEVOPS, A. **What is Azure DevOps?** 2022. Disponível em: <https://learn.microsoft.com/en-us/azure/devops/user-guide/what-is-azure-devops?view=azure-devops>.

EFRON, B.; TIBSHIRANI, R. J. **An Introduction to the Bootstrap**. [S.l.]: Chapman & Hall/CRC, 1993.

FERNÁNDEZ, D. M. et al. Naming the pain in requirements engineering: Contemporary problems, causes, and effects in practice. **Empirical Software Engineering**, Springer, v. 22, p. 2298–2338, 2017.

GARG, S. et al. 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)**. [S.l.: s.n.], 2021. p. 25–28.

GITLAB. **Get started with GitLab CI/CD**. 2023. Disponível em: <https://docs.gitlab.com/ee/ci/>.

HEYMANN, H. et al. Guideline for deployment of machine learning models for predictive quality in production. **Procedia CIRP**, v. 107, p. 815–820, 2022. ISSN 2212-8271. Leading manufacturing systems transformation – Proceedings of the 55th CIRP Conference on Manufacturing Systems 2022. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2212827122003523>.

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)**. [S.l.: s.n.], 2020. p. 395–404.

JOHN, M. M.; OLSSON, H. H.; BOSCH, J. Architecting ai deployment: A systematic review of state-of-the-art and state-of-practice literature. In: KLOTINS, E.; WNUK, K. (Ed.). **Software Business**. Cham: Springer International Publishing, 2021. p. 14–29. ISBN 978-3-030-67292-8.

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)**. [S.l.: s.n.], 2021. p. 1–8.

KALINOWSKI, M. et al. **Engenharia de Software para Ciência de Dados: Um guia de boas práticas com ênfase na construção de sistemas de Machine Learning em Python**. [S.l.]: Casa do Código, 2023.

KALINOWSKI, M. et al. Applying dppi: A defect causal analysis approach using bayesian networks. In: SPRINGER. **Product-Focused Software Process Improvement: 11th International Conference, PROFES 2010, Limerick, Ireland, June 21-23, 2010. Proceedings 11**. [S.l.], 2010. p. 92–106.

KALINOWSKI, M.; MENDES, E.; TRAVASSOS, G. H. Automating and evaluating probabilistic cause-effect diagrams to improve defect causal analysis. In: SPRINGER. **Product-Focused Software Process Improvement: 12th International Conference, PROFES 2011, Torre Canne, Italy, June 20-22, 2011. Proceedings 12**. [S.l.], 2011. p. 232–246.

KALINOWSKI, M. et al. **Naming the Pain in Machine Learning-Enabled Systems Engineering**. 2024.

KIM, M. et al. Data scientists in software teams: State of the art and challenges. **IEEE Transactions on Software Engineering**, IEEE, v. 44, n. 11, p. 1024–1038, 2017.

KOUROUKLIDIS, P. et al. 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)**. [S.l.: s.n.], 2021. p. 160–164.

LEI, S.; SMITH, M. Evaluation of several nonparametric bootstrap methods to estimate confidence intervals for software metrics. **IEEE Transactions on Software Engineering**, v. 29, n. 11, p. 996–1004, 2003.

LEWIS, G. A.; OZKAYA, I.; XU, X. Software architecture challenges for ml systems. In: **2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)**. [S.l.: s.n.], 2021. p. 634–638.

LINAKER, J. et al. Guidelines for conducting surveys in software engineering v. 1.1. **Lund University**, v. 50, 2015.

LUNNEBORG, C. E. Bootstrap inference for local populations. **Therapeutic Innovation & Regulatory Science**, v. 35, n. 4, p. 1327–1342, 2001.

MLFLOW. **What is MLflow?** 2023. Disponível em: <https://mlflow.org/docs/latest/what-is-mlflow.html>.

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**, v. 537, p. 132 – 147, 2020. Cited by: 72; All Open Access, Hybrid Gold Open Access. Disponível em: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85086395609&doi=10.1016%2fj.ins.2020.05.070&partnerID=40&md5=30781f86c3056851136ce7fb7678a97e>.

MäKINEN, S. et al. Who needs mlops: What data scientists seek to accomplish and how can mlops help? In: **2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN)**. [S.l.: s.n.], 2021. p. 109–112.

NAHAR, N. et al. 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, 2023. p. 171–183. Disponível em: <https://doi.ieeecomputersociety.org/10.1109/CAIN58948.2023.00034>.

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)**. [S.l.: s.n.], 2019. p. 183–186.

PALEYES, A.; URMA, R.-G.; LAWRENCE, N. D. Challenges in deploying machine learning: A survey of case studies. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 55, n. 6, dec 2022. ISSN 0360-0300. Disponível em: <https://doi.org/10.1145/3533378>.

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**, p. 1–1, 2023.

RUF, P. et al. Demystifying mlops and presenting a recipe for the selection of open-source tools. **Applied Sciences**, v. 11, n. 19, 2021. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/11/19/8861>.

SCHRÖER, C.; KRUSE, F.; GÓMEZ, J. M. A systematic literature review on applying crisp-dm process model. **Procedia Computer Science**, Elsevier, v. 181, p. 526–534, 2021.

SCHRöDER, T.; SCHULZ, M. Monitoring machine learning models: a categorization of challenges and methods. **Data Science and Management**, v. 5, n. 3, p. 105–116, 2022. ISSN 2666-7649. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666764922000303>.

SIEGEL, E. **Models Are Rarely Deployed: An Industry-wide Failure in Machine Learning Leadership**. 2022. Disponível em: <https://www.kdnuggets.com/2022/01/models-rarely-deployed-industrywide-failure-machine-learning-leadership.html>.

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**. [S.l.: s.n.], 2016. p. 120–131.

SYAFRUDIN, M. et al. Performance analysis of iot-based sensor, big data processing, and machine learning model for real-time monitoring system in automotive manufacturing. **Sensors**, v. 18, n. 9, 2018. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/18/9/2946>.

UP, S. **Machine Learning History: The Complete Timeline**. 2022. Disponível em: <https://startechup.com/blog/machine-learning-history/>.

WAGNER, S. et al. Status quo in requirements engineering: A theory and a global family of surveys. **ACM Trans. Softw. Eng. Methodol.**, Association for Computing Machinery, New York, NY, USA, v. 28, n. 2, 2019. ISSN 1049-331X.

WAGNER, S. et al. Challenges in survey research. **Contemporary Empirical Methods in Software Engineering**, Springer, p. 93–125, 2020.

WEINER, J. **Why AI/Data Science Projects Fail: How to avoid project pitfalls**. [S.l.]: Morgan; Claypool Publishers, 2021.

ZAHARIA, M. et al. Apache spark: A unified engine for big data processing. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 59, n. 11, p. 56–65, oct 2016. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/2934664>.

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)**. [S.l.: s.n.], 2020. p. 494–500.

ZIMELEWICZ, E. et al. **ML-Enabled Systems Model Deployment and Monitoring: Status Quo and Problems**. 2024.

ZIMELEWICZ, E. et al. MI-enabled systems model deployment and monitoring: Status quo and problems. In: BLUDAU, P. et al. (Ed.). **Software Quality as a Foundation for Security**. Cham: Springer Nature Switzerland, 2024. p. 112–131. ISBN 978-3-031-56281-5.