MLOps: Practices, Maturity Models, Roles, Tools, and Challenges – A Systematic Literature Review

Anderson Lima[1][a], Luciano Monteiro[1][b] and Ana Paula Furtado[1,2][c]

[1]CESAR School – Centro de Estudos e Sistemas Avançados do Recife, Recife, Pernambuco, Brazil
[2]Departamento de Computação – Universidade Federal Rural de Pernambuco, Recife, Pernambuco, Brazil

Keywords: MLOps, Machine Learning, Continuous Delivery.

Abstract: Context: The development of machine learning solutions has increased significantly due to the advancement of technology based on artificial intelligence. MLOps have emerged as an approach to minimizing efforts and improving integration between those who are in the process of deploying the models in the production environment. Objective: This paper undertakes a systematic literature review in order to identify practices, standards, roles, maturity models, challenges, and tools related to MLOps. Method: The study is founded on an automatic search method of selected digital libraries that applies selection and quality criteria to identify suitable papers that underpin the research. Results: The search initially found 1,905 articles of which 30 papers were selected for analysis. This analysis led to findings that made it possible to achieve the objectives of the research. Conclusion: The results allowed us to conclude that MLOps is still in its initial stage, and to recognize that there is an opportunity to undertake further academic studies that will prompt organizations to adopt MLOps practices.

1 INTRODUCTION

Artificial intelligence solutions are developed with a view to deducing hypotheses from the knowledge built in a learning process on a mass of historical data submitted to a machine learning (ML) model. Large volumes of high-quality data increase the accuracy of the models developed (Kang *et al*., 2020).

The typical lifecycle of building a machine learning solution involves separating historical data into training data and testing data, for subsequent submission of the model to the learning process with the training data (López García *et al*., 2020). Test data is then used to assess the accuracy of the model. This process is repeated several times until a satisfactory level of results is achieved.

Data scientists are often so concerned with the steps of creating or updating, training, and evaluating a model, they neglect the phase of publishing a paper and sharing it with another team. However, this is a critical step in the process because a machine learning model can only be explored by other applications or users after it has been published (López García et al., 2020).

One of the main challenges found when adopting artificial intelligence solutions is related to the implementation process in the operational environment of machine learning models built during the development process (Treveil et al., 2020).

In this context, MLOps (Machine Learning Operations) is considered to be a set of practices and principles for operationalizing data science solutions that is used to automate the implementation of machine learning models in an operating environment (Sweenor et al., 2020).

The objective of this paper is to identify studies that address practices, patterns, roles, maturity models, challenges, and tools for automating the activities of operationalizing machine learning models, and to present the state of the art with regard to MLOps.

This article is organized as follows. In Section 2, the theoretical framework is presented so as to contextualize the research problem. In Section 3, the

[a] https://orcid.org/0000-0002-9573-7907
[b] https://orcid.org/0000-0001-9166-1776
[c] https://orcid.org/0000-0002-5439-5314

method and procedures used to conduct the systematic literature review (SLR) are presented, and this includes a detailed look at the search terms and the criteria for selecting relevant studies. In Section 4, the result of applying the search and selection protocol is presented. In Section 5, the results of evaluating the quality criteria for the studies selected and the answers to the research questions are given. In Section 6, the conclusions of the study and a summary of the work carried out are presented and suggestions are made for future lines of research.

2 THEORETICAL BACKGROUND

In this section, concepts related to the topic that will be addressed in this SLR will be introduced. Initially, the definition of machine learning will be presented and how it has influenced the development of new technological solutions in the most diverse fields of application.

Then, the concepts related to DevOps practices and how they have contributed to improving the software development and deployment process in organizations will be discussed. Finally, the concepts related to MLOps will be introduced, and a parallel comparison will be made between applying the context of machine learning models and the DevOps practices used in traditional software development.

2.1 Machine Learning

Developing and adopting machine learning solutions have been consistently expanded across different business domains and research areas (Lwakatare, Crnkovic, & Bosch, 2020). The large volume of data generated by users and the advances obtained that resulted from research on big data prompted an increase in applying artificial intelligence in various fields of activity, including face detection and voice recognition. This led to better results than those obtained from traditional software and surpassed those by people who performed activities that require human intelligence (Zhou et al., 2020).

Machine learning solutions are intended to solve problems that are not part of the traditional software development lifecycle (A. Chen et al., 2020). Developing and implementing machine learning applications become more difficult and complex than traditional applications due to peculiarities inherent in this type of solution (Zhou *et al.*, 2020).

Traditional software development involves implementing a well-defined set of requirements, while the development of machine learning solutions is based on an experimentation process, in which developers constantly need to use new data sets, models and libraries, and to make adjustments to. software and parameters in order to improve the accuracy of the model being developed and therefore the quality of the artificial intelligence solution (A. Chen *et al.*, 2020).

The process of developing machine learning solutions involves a set of activities, including data collection, data preparation, defining the machine learning model, and carrying out the training process with a view to adjusting the parameters and thus to obtaining the expected result (Lwakatare, Crnkovic, & Bosch, 2020). Unlike traditional software applications – which by their nature are deterministic, machine learning models are probabilistic, depending on the learning achieved based on the data submitted during the process of constructing the machine learning solution (Akkiraju *et al.*, 2020). Ensuring that the results obtained are accurate also requires monitoring the machine learning model after the artificial intelligence software has been implemented. This must take into account that degradation may occur over time since the model continues to be trained with the new data that are available to it (Zhou *et al.*, 2020).

2.2 DevOps

The term DevOps refers to a set of practices that help establish collaboration between the software development teams and the infrastructure (operations) team, thereby seeking to reduce the software development lifecycle and thus contributing to the constant delivery of high-quality systems (Munappy et al., 2020). This approach emphasizes that a continuous delivery (CD) mechanism must be defined to help create a reliable and repeatable process of frequently delivering software increments and modifications in a production environment. This is a key factor in software quality assurance (Cano et al., 2021).

One of the main benefits of implementing DevOps is the flow of CI and CD. This helps to deliver software functionality more frequently (Sen, 2021). CI and CD practices directly contribute to agile software development by helping to put into production more frequent changes in the software under development; user feedback is anticipated; and opportunities for constant improvement are more easily identified (Munappy *et al.*, 2020).

In addition, CI/CD practices also contribute towards reducing risks, while the frequent implementation of new versions of the software enables users to have early contact with the application and, thus, to identify their needs and what improvements need to be incorporated into future versions (Cano *et al.*, 2021).

DevOps practices in the software deployment flow involve automating the deployment process. This includes the automatic provisioning of operating environments, and results in the development and operations teams reducing the manual process hitherto needed to deploy a new software version (Lwakatare, Crnkovic, & Bosch, 2020).

2.3 MLOps

The benefits provided by a CI and a CD solution can also be applied to the development and iterative deployment of machine learning applications (Zhou *et al.*, 2020).

In this context, based on DevOps practices, the concept of Machine Learning Operations (MLOps) arises. This aims to establish a set of practices that put tools, deployment flows, and work processes for developing machine learning models in a timely and cost-effective manner (Liu *et al.*, 2020). MLOps advocates automating and monitoring all stages of the process of developing and deploying machine learning systems (Granlund *et al.*, 2021).

In summary, MLOps practices encompass a complex orchestration of a set of software components put to work in an integrated way to perform at least five functions (Tamburri, 2020): (1) data collection; (2) data transformation; (3) continuous training of the machine learning model; (4) continuous implementation of the model; and (5) presentation of results to the end-user.

Furthermore, since machine learning applications depend on the data used in the training of the artificial intelligence model and new data is constantly submitted to the application, the performance of the solution may suffer degradation over time (Zhou *et al.*, 2020).

For this reason, monitoring machine learning solutions is one of the most relevant activities of MLOps practices, which is to ensure the efficiency and quality of the artificial intelligence solution over a long period of time (Cardoso Silva *et al.*, 2020).

3 METHODS AND PROCEDURES

Conducting an SLR has been frequently used in Software Engineering to make a comprehensive survey of available research on a particular research topic (Kitchenham *et al.*, 2015). The protocol adopted to carry out this SLR is based on the procedures proposed by Kitchenham *et al.* (2015).

This research method is used gather information to summarize the evidence related to a particular topic or, even, to identify any gaps and to suggest lines for more in-depth research in the future (Kitchenham *et al.*, 2015).

The protocol adopted to conduct the SLR included the following activities: (1) setting research questions; (2) selecting relevant studies; (3) evaluating the quality of these studies; (4) extracting data; and (5) synthesizing the data collected.

3.1 Research Questions

To carry out this study, a search on the topic was initially performed using Google Scholar, based on the terms "MLOps" and "machine learning operations" so as to obtain a preliminary set of published studies. This first automated search helped form an initial understanding of the topic and to define terms and expressions that served as the basis for the search undertaken for this SLR.

The information thus obtained, which is associated with the general objective of the research of identifying the studies that address practices, patterns, roles, challenges, and tools for automating the operationalization of machine learning models, was used to aid formulate the research questions (RQs) presented below:

- RQ1 - How are machine learning models deployed in production environments?
- RQ2 - What maturity models are used to assess the level of automation in deploying machine learning models?
- RQ3 - What roles and responsibilities are identified in the activities of operationalization of machine learning models?
- RQ4 - What tools are used in the activities for operationalizing machine learning models?
- RQ5 - What challenges are encountered with regard to deploying machine learning models in production environments?

Table 1: Result by search expression.

Search String	ACM DL	IEEE Xplore	Science Direct	Springer Link	Total
"MLOps"	5	15	75	25	120
"machine learning lifecycle"	15	5	5	2	27
"machine learning operations"	37	22	63	50	172
"machine learning" AND "deployment pipeline"	19	1	14	13	47
"machine learning" AND "continuous delivery"	80	5	80	70	235
"machine learning" AND "continuous integration"	343	34	282	157	816
"machine learning" AND "DevOps practices"	17	2	11	9	39
"machine learning" AND "deployment tools"	19	2	78	37	136
"machine learning" AND "deployment challenges"	45	4	50	21	120
"machine learning" AND "maturity model" AND "deployment"	22	0	139	32	193
Total	602	90	797	416	1.905

3.2 Search Strategy

The search strategy adopted to conduct this SLR was the automatic search in electronic research databases. These were the ACM Digital Library, IEEE Xplore, Science Direct, and Springer Link.

These databases were selected considering factors that included their coverage of the topic, how frequently they were updated, the availability of the entire content of the studies, the quality of the automatic search engine, the export feature of the results, or the integration with extensions that allowed this export and the ability to playback auto search. There was no start date set for the search. Thus, the SLR considered all articles available in these electronic databases until July 31, 2021.

3.3 Search Terms

From the preliminary search carried out to contextualize the research, key terms and expressions were identified that were later used in the form of search expressions. The selected terms were used to form the search strings presented in Table 1.

Based on these terms, some pilot searches were carried out to evaluate the most adequate combination to define the search expressions to be adopted in this systematic review.

After evaluation, it was defined that the best strategy would be to carry out a set of isolated searches and, subsequently, consolidate the articles found, at the expense of using a more elaborate search expression which could restrict the studies that address the research theme. To enable a greater number of articles to be selected, the search was carried out in all fields, including the title, abstract, key expressions, and content of the articles.

The search expressions defined were applied to the electronic search bases and 1,905 articles were returned by the search engines. The search expressions used and the detailed result by electronic search base are presented in Table 1.

3.4 Selection Criteria

Selection criteria are defined to assess the relevance of the article found for the SLR (Kitchenham *et al.*, 2015). Considering the scope and objective of the research, the inclusion criteria adopted to select the studies to be analyzed in the systematic review are listed in Table 2.

Table 2: Inclusion criteria.

IC	Criteria
1	Studies that address Machine Learning Operations (MLOps) in general
2	Studies that assess the lifecycle of machine learning solutions
3	Studies dealing with machine learning process maturity models
4	Studies that analyze the roles and responsibilities involved in the development and implementation of machine learning solutions
5	Studies that comprise tools for deploying machine learning solutions
6	Studies that identify challenges for the development and deployment of machine learning models

The exclusion criteria are used to eliminate from the analysis publications that do not contribute to collecting information that allows the RQs to be answered. Table 3 lists the exclusion criteria adopted in this systematic review.

After defining the selection criteria, they were applied to the studies found after performing the searches in the electronic search databases. Initially, 250 duplicate articles were removed, these being identified with the support of the tool used to assist in

Table 3: Exclusion criteria.

EC	Criteria
1	The study was not published in English
2	Studies that address the application of machine learning models
3	Papers published as a short paper or poster
4	The study was not related to machine learning operations
5	Studies that do not allow access to its content
6	Papers that do not address the research questions

conducting the systematic review. This left 1,655 articles to be evaluated.

Then, the title of the articles was verified according to the inclusion and exclusion criteria defined. At this stage, 40 articles were directly selected; there were doubts about the relevance to the research of another 284, and the remaining 1,331 were excluded. Next, after reading the abstract of the papers that had not been selected in the previous phase based on their title, another 17 articles were selected, resulting in a total of 57 articles being selected for a full reading and evaluation of their relevance.

After doing so, 27 studies were removed, three of them because it was not possible to access their content; another three because they were invalid publication types; one study because it was duplicated, and finally another 20 articles were removed based on the exclusion criteria. Thus, 30 articles remained for data extraction, analysis, and synthesis. Figure 1 details the process described in this section for selecting the studies.

3.5 Quality Assessment

Considering that the selected studies come from the most varied types of research, the quality of the articles had to be assessed.

According to Kitchenham *et al.* (2015), the reasons for carrying out this evaluation include: (1) providing a means of evaluating the individual importance of the article when the results are being consolidated; (2) guiding the interpretation of the findings and (3) determining the degree to which their inferences corroborate; and (4) guiding future research recommendations.

The criteria for quality assessment adopted in this SLR are based on the parameters defined by Kitchenham *et al.* (2015). The quality criteria correspond to a set of factors to be analyzed in each study, to identify possible biases in the results of these studies. The criteria can be classified into three types: (1) bias; (2) internal validity; and (3) external validity.

The criteria that were applied in this paper are presented in Table 4.

The articles were evaluated under each criterion, with the following scores being assigned: 1 point if the article's answer to the criterion is "Yes" and 0 points if the answer is "No". The total score that an article can obtain, according to the quality assessment criteria, is 5 points.

It should be noted, however, that the quality criteria were not used to exclude studies selected in the previous stages but were adopted exclusively to assess the quality of the studies and their relevance for this paper.

Table 4: Quality criteria.

QC	Criteria
1	Does the study report unequivocal discoveries based on evidence and argument?
2	Did the study present a research project and not an expert opinion?
3	Did the study fully describe the context analyzed?
4	Were the objectives of the study clearly defined?
5	Have the research results been properly validated?

3.6 Data Extraction, Analysis, and Synthesis

During the data extraction phase, three main activities were undertaken: the studies were classified based on their field of research, a thematic synthesis was written (Cruzes & Dyba, 2011) and evidence of the research questions being addressed was sought (RQ1 to RQ5).

After concluding the coding phase and performing an initial analysis, the studies were sorted into the following themes: (1) Machine Learning Lifecycle; (2) Maturity Model; (3) Roles and Responsibilities; (4) MLOps Tools; and (5) Machine Learning Deployment Challenges.

3.7 Validity Threats

While conducting this evaluation, factors that could negatively influence the results obtained from the studies selected were identified. Initially, it should be clarified that the search for the studies was carried out in the selected electronic research databases. Thus, relevant studies may not have been selected because they were not selected because they were not included in these electronic research databases.

The definition of search terms considered preliminary tests to identify the best combinations to identify studies that could contribute to the research
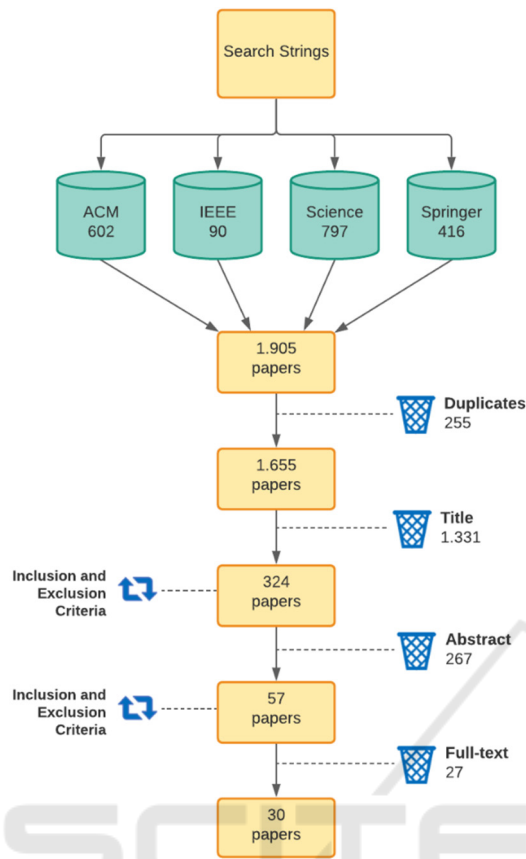
Figure 1: Study selection protocol.

synthesis procedures and data were extracted from them.

The protocol we used to select studies required us to observe the current theme of the work, depending on the year of publication of the studies. All selected articles were published between 2019 and 2021, thus highlighting the contemporaneity of the theme. Figure 2 shows a graph detailing the number of papers published per year and the research base.

The distribution of selected articles according to their contribution to the RQs was determined. 18 were about lifecycle, 4 about maturity models, 3 about roles, 6 about tools, and 8 about challenges. Some papers contributed to more than one of these themes.

The number of selected studies allows us to calculate the accuracy of this SLR, obtaining a value of 1.81%, considering the total of 1,655 articles initially evaluated, with the selection of 57 (3.44%) potentially relevant studies and 30 (1.81%) studies effectively selected for analysis.

It is plausible to observe that the study selection criteria were quite strict in filtering the articles initially found in the electronic search bases. This is justifiable considering that the terms used in the search, when applied separately as defined in this research, allowed the framing of a significant number of studies that met the requirements.

objective. Possibly, the search expressions used may have ignored relevant studies due to their not being identified by search engines in the electronic search bases.

It should also be noted that this research was conducted on peer-reviewed scientific papers, while articles published in blogs, magazines, websites, and other related articles, considered 'grey' literature, were not included. Considering that the MLOps theme has been heavily explored in the industry, relevant information may not have been analyzed in this research for this reason.

4 RESULTS

4.1 Overview of the Selected Literature

The protocol adopted in this SLR led to 30 studies being selected which were then used to attempt to answer the RQs proposed in this study. Therefore, these studies were analyzed, subjected to content
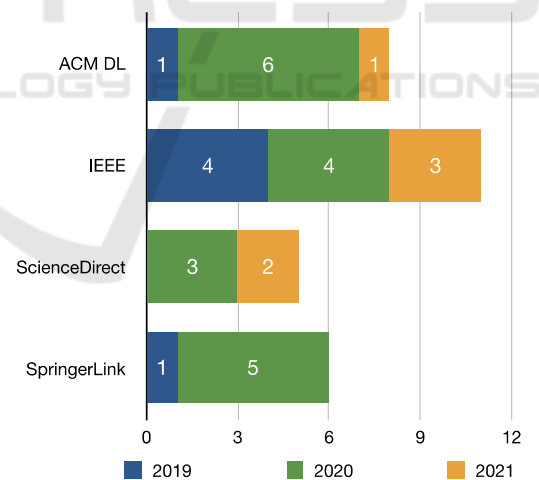

Figure 2: Distribution of papers by the digital library.

However, when analyzing the articles found based on the search string, it was found that most of them did not directly or indirectly contribute to the research questions, and therefore, they were considered to be irrelevant to this study.

5 DISCUSSION

This section provides a detailed discussion on the results of this SLR. The subsections synthesize the evaluation of the quality of the selected studies, as well as the answers to the research questions RQ1 to RQ5.

5.1 Criteria for Quality

Quality criteria are presented as a mechanism to assess how the study minimizes bias and maximizes the internal and external validity of the study performed (Kitchenham *et al.*, 2015). Table 5 shows the result of the evaluation of the selected articles against the quality criteria defined in this SLR.

5.2 RQ1 - How Are Machine Learning Models Deployed in Production Environments?

The objective of this RQ is to identify and understand the activities carried out for the publication of machine learning models in a production environment, as well as to evaluate the life cycles documented in scientific articles that record the steps for the development and implementation of solutions for machine learning.

In the selected studies, 18 papers were identified that addressed how to identify the stages or activities related to the life cycle of developing machine learning solutions.

This question was the one that received the most contributions from the selected articles in the protocol of this SLR. However, in general, the articles described different lifecycle proposals for developing machine learning solutions but did not detail the stages of implementing the models in a production environment, which would enable activities related to operationalizing machine learning models to be identified.

Serban *et al.* (2020) mention a variety of machine learning solution development lifecycles, noting that none of them emerges as a consensus in the related literature. In this context, they propose a taxonomy for group machine learning model development activities and relate the implementation activities.

One of the most established approaches to formalizing MLOps is called Continuous Delivery for Machine Learning - CD4ML (Dhanorkar *et al.*, 2021). Proposed by ThoughtWorks, it sets out to automate the entire end-to-end machine learning lifecycle. According to Granlund *et al.* (2021), the

Table 5: Result of the evaluation of quality criteria.

| Study Ref | Quality Criteria | | | | | Total |
	1	2	3	4	5	
Serban *et al.*, 2020	1	1	1	1	1	5
Amershi *et al.*, 2019	1	1	1	0	0	3
Munappy *et al.*, 2020	1	1	1	0	0	3
Karlaš *et al.*, 2020	1	1	0	1	1	4
A. Chen *et al.*, 2020	0	1	1	0	0	2
Zhang *et al.*, 2020	1	0	1	0	0	2
Ismail *et al.*, 2019	1	0	1	1	0	3
Z. Chen *et al.*, 2020	1	1	1	1	1	5
Dhanorkar *et al.*, 2021	1	1	1	0	0	3
Tamburri, 2020	0	0	1	0	0	1
Zhou *et al.*, 2020	1	1	1	0	1	4
Cardoso Silva *et al.*, 2020	1	0	1	1	1	4
Granlund *et al.*, 2021	1	0	1	1	0	3
Maskey *et al.*, 2019	1	0	1	0	0	2
Souza *et al.*, 2019	1	1	1	1	1	5
Lwakatare, Crnkovic, & Bosch, 2020	1	1	1	1	0	4
Dang *et al.*, 2019	0	0	1	0	0	1
Liu *et al.*, 2020	1	1	1	1	1	5
Giray, 2021	1	1	1	1	1	5
Martín *et al.*, 2021	1	1	1	1	1	5
Lwakatare, Raj *et al.*, 2020	1	1	1	1	1	5
Janardhanan, 2020	0	0	1	1	0	2
van den Heuvel & Tamburri, 2020	1	0	1	0	1	3
Lwakatare, Crnkovic, Rånge, *et al.*, 2020	1	1	1	1	1	5
Martínez-Fernández *et al.*, 2021	0	0	1	1	0	2
Figalist *et al.*, 2020	1	1	1	1	1	5
Akkiraju *et al.*, 2020	1	0	1	1	0	3
Bachinger & Kronberger, 2020	1	0	1	0	0	2
Ashmore *et al.*, 2021	1	1	1	1	1	5
Lwakatare *et al.*, 2019	1	1	1	1	1	5

approach produces three additional artifacts in relation to DevOps practices for traditional software: (1) different data sets used for training model and their versioning; (2) a model and its versioning; and (3) monitoring the output of the model to detect bias and other problems.

For Maskey *et al.* (2019), it is not just about finding the right algorithm and creating the model. It is an integrated approach in an end-to-end lifecycle. They present a simplified model and specifically describe the activities for deploying machine learning models, highlighting production performance requirements that must be observed: (1) metrics and baselines for an initial model, (ii) monitoring over time, (iii) since model and production software will change, we need to test model changes on historical

data and also run current production model against the baseline performance, and (iv) in production there will be new data, thus, we need to test the production model on the latest data.

In their study, Lwakatare, Crnkovic, & Bosch (2020) group the steps of a machine learning workflow into three stages and describe a proposal for integrating the ML workflow with DevOps organized into four distinct processes: (1) DM - Data management; (2) Mod - ML Modeling; (3) Dev - Development; and (4) Ops - System operation.

van den Heuvel & Tamburri (2020) point out that, like DevOps, MLOps adopts continuous integration and continuous testing cycle to produce and deploy new versions of smart enterprise applications.

The stages foreseen in the life cycle called CRISP-DM have been used as a reference for defining a workflow for developing machine learning solutions (Bachinger & Kronberger. 2020).

Akkiraju *et al.* (2020) highlight the critical aspect for deploying machine learning models in production, due to the decisions to be taken by the Deployment Lead in evaluating aspects involving infrastructure components.

The studies identified in this research do not go into sufficient depth to indicate activities related to the process of implementing machine learning models in a production environment, but rather they limit themselves to presenting life cycle models that encompass the complete workflow of developing solutions of machine learning. Only two papers described the activities performed in the process of deploying machine learning models.

5.3 RQ2 - What Maturity Models Are Used to Assess the Level of Automation in Deploying Machine Learning Models?

This RQ was presented to get to know the models registered in the scientific literature for assessing maturity after MLOps practices are adopted.

However, studies that directly addressed the focus of this question were not identified. Four articles were found that presented maturity models for the machine learning development process in a broader way. Some of them addressed the implementation stage of ML models, thus partially contributing to meeting the objective of this RQ.

Amershi *et al.* (2019) propose a maturity model organized into six dimensions inspired by the assessment models found in the Capability Maturity Model (CMM) and the Six Sigma methodology. The maturity evaluation criteria foreseen in the model

verify if the activity: (1) has defined goals, (2) is consistently implemented, (3) is documented, (4) is automated, (5) is measured and tracked, and (6) is continuously improved.

According to Dhanorkar *et al.* (2021), the organizations can be classified according to the degree of maturity in the development of machine learning solutions, there being three levels: (1) Data-Centric - the organization is figuring out how to manage and use data; (2) Model-centric - the organization is figuring out how to build their first model and reach production; and (3) Pipeline-centric – the organization has models in production, and they are increasingly business-critical.

Lwakatare *et al.* (2020) describe five stages of improvement in development practices: (1) a manual data science-driven process, (2) a standardized experimental-operational symmetry process, (3) an automated ML workflow process, (4) Integrated software development and ML workflow processes, and (v) an automated and fully integrated CD and ML workflow process.

Akkiraju *et al.* (2020) propose an adaptation of the Capability Maturity Model (CMM) for the context of machine learning. They define five maturity levels for each capability they present, namely: (1) initial, (2) repeatable, (3) defined, (4) managed, and (5) optimizing. The study details the characteristics of each maturity level according to the assessed capacity. The authors enumerated as capabilities of the development processes of the machine learning model the following items: (a) AI Model Goal Setting; (b) Data Pipeline Management; (c) Feature Preparation Pipeline; (d) Train Pipeline Management; (e) Model Quality, Performance and Model Management; (f) Model Error Analysis; (g) Model Fairness & Trust; (h) Model Transparency.

The maturity models identified in this SLR indirectly allow the level of improvement in the activities of developing machine learning models to be assessed. However, it is observed that aspects related to MLOps are not directly assessed. Therefore, it appears that this finding is an opportunity for improvement by developing research for this purpose.

5.4 RQ3 – What Roles and Responsibilities Are Identified in the Activities of Operationalizing Machine Learning Models?

This RQ was drawn up with the purpose of knowing the roles involved in documented MLOps processes, thus seeking to consolidate the professional profiles

responsible for carrying out the activities in the context of developing and implementing machine learning models, especially with a focus on integration between these domains.

Among the articles evaluated, three studies were identified that describe profiles and responsibilities for the activities of developing and implementing machine learning models. Specifically involving the activities of operationalizing machine learning models, only one of the studies described a profile associated with the activity of implementing models (Souza *et al*., 2019).

Karlaš *et al*. (2020) present a proposal for a continuous integration service to be used in the development of machine learning models. They present a framework for testing machine learning models based on strict theoretical limits and allowing a principle-based way to avoid overfitting the test set. In this context, they define three roles foreseen in the definition of the framework: (1) a manager, responsible for defining the test conditions, considering that he/she has a broader view of the solution architecture and all its components; (2) a data curator, in charge of providing up-to-date test data for the system, and who may perform a set of pre-processing steps before providing the test data; (3) a developer, responsible for building and improving machine learning models, and also for submitting new machine learning models for testing and further implementation of the solution.

Souza *et al*. (2019) present three roles related to developing machine learning models: (1) domain scientists, who have deep knowledge of the domain and play an important role in obtaining data and validating the results; (2) computational scientists and engineers, with great technical skills to prepare the environment for the operation of machine learning models; (3) ML scientists and engineers, in charge of designing new machine learning models, have advanced knowledge of ML statistics and algorithms. Their study also describes an additional profile, called provenance specialists, who are responsible for managing the supply of data in the life cycle of developing machine learning solutions, and who must have skills in the business domain and machine learning.

Liu *et al*. (2020) describe an open-source standards-based platform that provides a development tool and an operating environment for developing, training, evaluating, approving, delivering, and deploying models for hosting machine learning solutions. Their study details the functioning of the platform in activities performed by three distinct profiles: (1) a data scientist, who is responsible for creating and training the models; (2) a manager, in charge of evaluating the models before they are made available; (3) an end application developer, who is responsible for developing applications that the models produced will run.

Thus, it is possible to distinguish some common profiles that can be used to identify the roles and responsibilities addressed in the articles analyzed in this study, namely: (1) a domain specialist; (2) a data scientist; (3) a manager; (4) a data engineer; and (5) a developer.

5.5 RQ4 - What Tools Are Used in the Activities of Operationalizing Machine Learning Models?

The purpose of this RQ is to provide information about the tools and solutions registered in the academy for application in the development and operationalization of machine learning models.

For this purpose, four studies were identified that presented solutions that are available on the market, either under an open-source license or under a commercial license. Furthermore, two other studies proposed platforms to automate the implementation of machine learning models.

The first of them, MLflow is an open-source platform for the machine learning lifecycle – contemplating experimentation, reproducibility, and deployment – and is designed to work with any machine learning library and any programming language, according to A. Chen *et al*. (2020) and Janardhanan (2020). It consists of four components, designed to overcome the fundamental challenges in each phase of the machine learning lifecycle: (1) MLflow Tracking: this allows the running of the model to be recorded, including the code and parameters used, data input, metrics, and results. This enables the visualization, comparison and search of these models on a historical basis; (2) MLflow Models: These are in a generic model packaging format that lets it be implemented in different environments; (3) MLflow Projects: these set a format for packaging code into reusable projects, including its dependencies, code for execution and parameters for programmatic execution; (4) the MLflow Model Registry is a collaborative environment for cataloging models and managing their deployment lifecycles (A. Chen *et al*., 2020).

Two other open-source tools were identified in the studies analyzed: Polyaxon and Kubeflow. Polyaxon is an open-source machine learning model lifecycle management tool, providing a platform for the reproducibility and scalability of machine learning

and artificial intelligence applications (Janardhanan, 2020). Kubeflow is an end-to-end machine learning model lifecycle management platform that enables a workflow for deploying models in any production environment, on-premises or in the cloud, using clusters based on Kubernetes, which promotes, therefore, the simplification and portability of models in different infrastructures (Zhou *et al*., 2020).

Under the commercial licensing format, the Comet.ml platform presents itself as an automatic versioning solution for machine learning models. It tracks and organizes development efforts at all stages, contemplating the provision of an automated mechanism for optimizing hyper-parameters (Janardhanan, 2020).

Finally, two studies proposed new tools based on solutions available on the market. Martín *et al*. (2021) present the Kafka-ML platform, which consists of an open-source framework that allows the pipeline management of machine learning and artificial intelligence applications using the data flow architecture. It is a platform developed based on the Apache Kafka solution – unified, high-capacity, and low-latency for real-time data treatment – with support for the TensorFlow library for data flow integration with machine learning models. In another study, the open-source platform called MLModelCI is presented as a solution for optimizing, managing, and deploying machine learning models as a service (MLaaS) (Zhang *et al*., 2020). This allows models to be converted automatically into optimized formats, and thus configures them for different scenarios and classifies the models as cloud services using container technology. According to the authors, the tool enables the development cycle of machine learning solutions to be reduced from weeks and days to hours and even minutes (Zhang *et al*., 2020).

These solutions, however, despite being based on tools that are widely used in the market, are presented as innovations and tool proposals and, therefore, are still observed in a restricted context of research.

5.6 RQ5 - What Challenges Are Encountered for Deploying Machine Learning Models in Production Environments?

The purpose of this RQ is to raise the difficulties and challenges reported in operationalizing machine learning models. In all, eight articles that directly or indirectly addressed this RQ were selected.

Most of the studies analyzed report challenges related to constructing LM models in specific contexts. For example, Lwakatare, Raj, *et al*. (2020)

present a review of challenges and solutions for the development, maintenance, and implementation of machine learning solutions in large-scale ML systems, in which the main categories of Software Engineering challenges were organized into four themes: (1) adaptability; (2) scalability; (3) privacy; and (4) safety. From this classification, the challenges are associated and detailed in their study. In the same sense, Amershi *et al.* (2019) describe the challenges for building large-scale ML applications, based on interviews and surveys among Microsoft project participants. Martínez-Fernández *et al.* (2021) address challenges related to the context of autonomous systems that make use of ML, for which they highlight the difficulty of deploying and versioning AI models in context-dependent autonomous systems.

In the context of adopting DevOps practices when developing machine learning solutions, addressed by MLOps, Lwakatare, Crnkovic, & Bosch (2020) present the challenges in integrating these practices with developing artificial intelligence applications. Their study highlights that integrating software development with the ML workflow is still not well defined, and they describe the challenges inherent in this integration. In a similar approach, Dang *et al.* (2019) state that adopting AIOps (equivalent to MLOps but in the broader aspect) is still at an early stage, and is proving to be especially challenging in organizations. Their study presents some challenges and research proposals for innovations in this theme.

Giray (2021) seeks to present the state of the art in software engineering for ML systems. He argues that researchers and professionals in the areas of SE and AI/ML have a holistic view of ML systems engineering. Therefore, he presents a series of SE challenges related to implementing ML models, among which some were identified as pertinent to the stage of implementing machine learning models.

In the framework proposed by Figalist et *al.* (2020), activities are categorized into domains identified from a literature review they carried out. Regarding the implementation of machine learning models, their study highlights that it is essential, for the delivery of the desired benefits with ML models, to go beyond the analysis of prototypes of the models in controlled environments and to implement the models in the environments in which they will be used.

Lwakatare *et al.* (2019) set out to identify and classify challenges for developing and implementing ML systems in a market context. They define five stages of the evolution of using ML components in software systems. The stages have activities that were

organized into four groups: (1) assemble dataset; (2) create model; (3) train and evaluate model; (4) deploy the model. As for the activities of implementing ML models, the authors present the challenges according to the stage of evolution they are in.

6 CONCLUSIONS

This study undertook a systematic review of the literature to identify practices, standards, roles, maturity models, challenges, and tools adopted to automate the activities for operationalizing machine learning models. The aim was to present the state of the art of MLOps practices.

An automatic search was conducted in the selected electronic databases, and initially this resulted in finding 1,905 articles that satisfied the search strings proposed. A protocol was then drawn up and applied. This led to 30 articles being selected from which data were extracted and used to contribute to the answers to the research questions.

The analysis of the 30 articles enabled us to conclude that there is not yet a lifecycle model of machine learning solutions established as a standard in the scientific literature. The activities related to the development, training, testing, implementation, and operation of machine learning models are defined and organized in phases using various proposals for approaches, but without a consensus having formed among researchers and professionals who tackle the development of ML solutions. Hence, there is a significant gap in the detailing of activities related to operationalizing machine learning models, which characterize the MLOps practices.

This situation contributed to the difficulty in consolidating a maturity model that can assess the level of adoption and mastery of MLOps practices in organizations. During the research, some studies were found to have proposed maturity models based on the Capability Maturity Model – CMM. However, these models sought a broader evaluation of the machine learning solution development process, and do not address in sufficient depth the practices related to MLOps.

The roles and responsibilities mentioned in the articles analyzed allow us to distinguish some common profiles in the studies, namely: (1) a domain specialist; (2) a data scientist; (3) a manager; (4) a data engineer; and (5) a developer.

Analysis of the articles identified some tools used for managing the lifecycle of machine learning models that addressed the activities of deployment and operationalization of ML models. Tools mentioned in the articles include MLflow, Kuberflow, Polyaxon, and Comet.ml. Furthermore, two other tools, according to the articles analyzed, were proposed based on the use of open-source solutions available on the market: Kafka-ML and MLModelCI.

Regarding the challenges, the studies analyzed presented a set of obstacles and aspects to be observed in the process of developing and implementing machine learning solutions under different contexts and applications.

The results consolidated in this research could contribute to the conduct of new studies to examine in greater depth the areas addressed in this article. In addition, it is hoped that this paper will be useful to professionals who work on developing machine learning solutions when conducting the process of implementing MLOps in organizations.

As a proposal for future work, further research is suggested into standardizing activities related to MLOps practices, to include defining roles and responsibilities, and into developing a maturity model that assesses the level of adoption of MLOps practices in an organization.

The conclusion can be drawn that research on MLOps is still in its initial stage of development. Therefore, it would be highly opportune for academic work to be carried out to fill this gap and thus to promote the adoption of MLOps practices in organizations.

REFERENCES

Akkiraju, R., Sinha, V., Xu, A., Mahmud, J., Gundecha, P., Liu, Z., Liu, X., & Schumacher, J. (2020). Characterizing Machine Learning Processes: A Maturity Framework. In D. Fahland, C. Ghidini, J. Becker, & M. Dumas (Orgs.), Business Process Management (p. 17–31). Springer International Publishing. https://doi.org/10.1007/978-3-030-58666-9_2

Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., & Zimmermann, T. (2019). Software engineering for machine learning: A case study. Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Practice, 291–300. https://doi.org/10.1109/ICSE-SEIP.2019.00042

Ashmore, R., Calinescu, R., & Paterson, C. (2021). Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges. ACM Computing Surveys, 54(5), 1–39. https://doi.org/10.1145/3453444

Bachinger, F., & Kronberger, G. (2020). Concept for a Technical Infrastructure for Management of Predictive Models in Industrial Applications. In R. Moreno-Díaz, F. Pichler, & A. Quesada-Arencibia (Orgs.), Computer

Aided Systems Theory – EUROCAST 2019 (p. 263–270). Springer International Publishing. https://doi.org/10.1007/978-3-030-45093-9_32

Cano, P. O., Mejia, A. M., De Gyves Avila, S., Dominguez, G. E. Z., Moreno, I. S., & Lepe, A. N. (2021). A Taxonomy on Continuous Integration and Deployment Tools and Frameworks. In J. Mejia, M. Muñoz, Á. Rocha, & Y. Quiñonez (Orgs.), New Perspectives in Software Engineering (p. 323–336). Springer International Publishing. https://doi.org/10.1007/978-3-030-63329-5_22

Cardoso Silva, L., Rezende Zagatti, F., Silva Sette, B., Nildaimon dos Santos Silva, L., Lucrédio, D., Furtado Silva, D., & de Medeiros Caseli, H. (2020). Benchmarking Machine Learning Solutions in Production. 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), 626–633. https://doi.org/10.1109/ICMLA51294.2020.00104

Chen, A., Chow, A., Davidson, A., DCunha, A., Ghodsi, A., Hong, S. A., Konwinski, A., Mewald, C., Murching, S., Nykodym, T., Ogilvie, P., Parkhe, M., Singh, A., Xie, F., Zaharia, M., Zang, R., Zheng, J., & Zumar, C. (2020). Developments in MLflow: A System to Accelerate the Machine Learning Lifecycle. Proceedings of the Fourth International Workshop on Data Management for End-to-End Machine Learning. https://doi.org/10.1145/3399579.3399867

Chen, Z., Cao, Y., Liu, Y., Wang, H., Xie, T., & Liu, X. (2020). A Comprehensive Study on Challenges in Deploying Deep Learning Based Software. Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 750–762. https://doi.org/10.1145/3368089.3409759

Cruzes, D. S., & Dyba, T. (2011). Recommended Steps for Thematic Synthesis in Software Engineering. 2011 International Symposium on Empirical Software Engineering and Measurement, 275–284. https://doi.org/10.1109/ESEM.2011.36

Dang, Y., Lin, Q., & Huang, P. (2019). AIOps: Real-World Challenges and Research Innovations. 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), 4–5. https://doi.org/10.1109/ICSE-Companion.2019.00023

Dhanorkar, S., Wolf, C. T., Qian, K., Xu, A., Popa, L., & Li, Y. (2021). Who Needs to Know What, When?: Broadening the Explainable AI (XAI) Design Space by Looking at Explanations Across the AI Lifecycle. Designing Interactive Systems Conference 2021, 1591–1602. https://doi.org/10.1145/3461778.3462131

Figalist, I., Elsner, C., Bosch, J., & Olsson, H. H. (2020). An End-to-End Framework for Productive Use of Machine Learning in Software Analytics and Business Intelligence Solutions. In M. Morisio, M. Torchiano, & A. Jedlitschka (Orgs.), Product-Focused Software Process Improvement (p. 217–233). Springer International Publishing. https://doi.org/10.1007/978-3-030-64148-1_14

Giray, G. (2021). A software engineering perspective on engineering machine learning systems: State of the art and challenges. Journal of Systems and Software, 180, 111031. https://doi.org/10.1016/j.jss.2021.111031

Granlund, T., Kopponen, A., Stirbu, V., Myllyaho, L., & Mikkonen, T. (2021). MLOps Challenges in Multi-Organization Setup: Experiences from Two Real-World Cases. 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN), 82–88. https://doi.org/10.1109/WAIN52551.2021.00019

Ismail, B. I., Khalid, M. F., Kandan, R., & Hoe, O. H. (2019). On-Premise AI Platform: From DC to Edge. Proceedings of the 2019 2nd International Conference on Robot Systems and Applications, 40–45. https://doi.org/10.1145/3378891.3378899

Janardhanan, P. S. (2020). Project repositories for machine learning with TensorFlow. Procedia Computer Science, 171,188–196.https://doi.org/10.1016/j.procs.2020.04.020

Kang, Z., Catal, C., & Tekinerdogan, B. (2020). Machine learning applications in production lines: A systematic literature review. Computers & Industrial Engineering, 149, 106773. https://doi.org/10.1016/j.cie.2020.106773

Karlaš, B., Interlandi, M., Renggli, C., Wu, W., Zhang, C., Mukunthu Iyappan Babu, D., Edwards, J., Lauren, C., Xu, A., & Weimer, M. (2020). Building Continuous Integration Services for Machine Learning. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (p. 2407–2415). Association for Computing Machinery. https://doi.org/10.1145/3394486.3403290

Kitchenham, B. (2004). Procedures for Performing Systematic Reviews. 33.

Kitchenham, B.A., Budgen, D., Brereton, P. (2015). Evidence-Based Software Engineering and Systematic Reviews, vol. 4. CRC press.

Liu, Y., Ling, Z., Huo, B., Wang, B., Chen, T., & Mouine, E. (2020). Building A Platform for Machine Learning Operations from Open Source Frameworks. IFAC-PapersOnLine, 53(5), 704–709. https://doi.org/10.1016/j.ifacol.2021.04.161

López García, Á., De Lucas, J. M., Antonacci, M., Zu Castell, W., David, M., Hardt, M., Lloret Iglesias, L., Moltó, G., Plociennik, M., Tran, V., Alic, A. S., Caballer, M., Plasencia, I. C., Costantini, A., Dlugolinsky, S., Duma, D. C., Donvito, G., Gomes, J., Heredia Cacha, I., … Wolniewicz, P. (2020). A Cloud-Based Framework for Machine Learning Workloads and Applications. IEEE Access, 8, 18681–18692. https://doi.org/10.1109/ACCESS.2020.2964386

Lwakatare, L. E., Crnkovic, I., & Bosch, J. (2020). DevOps for AI – Challenges in Development of AI-enabled Applications. 2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 1–6. https://doi.org/10.23919/SoftCOM50211.2020.9238323

Lwakatare, L. E., Crnkovic, I., Rånge, E., & Bosch, J. (2020). From a Data Science Driven Process to a Continuous Delivery Process for Machine Learning Systems. In M. Morisio, M. Torchiano, & A. Jedlitschka (Orgs.), Product-Focused Software Process

Improvement (p. 185–201). Springer International Publishing. https://doi.org/10.1007/978-3-030-64148-1_12

Lwakatare, L. E., Raj, A., Bosch, J., Olsson, H. H., & Crnkovic, I. (2019). A Taxonomy of Software Engineering Challenges for Machine Learning Systems: An Empirical Investigation. In P. Kruchten, S. Fraser, & F. Coallier (Orgs.), Agile Processes in Software Engineering and Extreme Programming (Vol. 355, p. 227–243). Springer International Publishing. https://doi.org/10.1007/978-3-030-19034-7_14

Lwakatare, L. E., Raj, A., Crnkovic, I., Bosch, J., & Olsson, H. H. (2020). Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions. Information and Software Technology, 127, 106368. https://doi.org/10.1016/j.infsof.2020.106368

Martín, C., Langendoerfer, P., Zarrin, P. S., Díaz, M., & Rubio, B. (2021). Kafka-ML: Connecting the data stream with ML/AI frameworks. Future Generation Computer Systems, 126, 15–33. https://doi.org/10.1016/j.future.2021.07.037

Martínez-Fernández, S., Franch, X., Jedlitschka, A., Oriol, M., & Trendowicz, A. (2021). Developing and Operating Artificial Intelligence Models in Trustworthy Autonomous Systems. In S. Cherfi, A. Perini, & S. Nurcan (Orgs.), Research Challenges in Information Science (p. 221–229). Springer International Publishing. https://doi.org/10.1007/978-3-030-75018-3_14

Maskey, M., Ramachandran, R., Gurung, I., Freitag, B., Miller, J. J., Ramasubramanian, M., Bollinger, D., Mestre, R., Cecil, D., Molthan, A., & Hain, C. (2019). Machine Learning Lifecycle for Earth Science Application: A Practical Insight into Production Deployment. IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 10043–10046. https://doi.org/10.1109/IGARSS.2019.8899031

Munappy, A. R., Mattos, D. I., Bosch, J., Olsson, H. H., & Dakkak, A. (2020). From Ad-Hoc Data Analytics to DataOps. Proceedings of the International Conference on Software and System Processes, 165–174. https://doi.org/10.1145/3379177.3388909

Sen, A. (2021). DevOps, DevSecOps, AIOPS- Paradigms to IT Operations. In P. K. Singh, A. Noor, M. H. Kolekar, S. Tanwar, R. K. Bhatnagar, & S. Khanna (Orgs.), Evolving Technologies for Computing, Communication and Smart World (p. 211–221). Springer. https://doi.org/10.1007/978-981-15-7804-5_16

Serban, A., van der Blom, K., Hoos, H., & Visser, J. (2020). Adoption and Effects of Software Engineering Best Practices in Machine Learning. Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). https://doi.org/10.1145/3382494.3410681

Souza, R., Azevedo, L., Lourenço, V., Soares, E., Thiago, R., Brandão, R., Civitarese, D., Brazil, E., Moreno, M., Valduriez, P., Mattoso, M., Cerqueira, R., & Netto, M. A. S. (2019). Provenance Data in the Machine Learning Lifecycle in Computational Science and Engineering. 2019 IEEE/ACM Workflows in Support of Large-Scale

Science (WORKS), 1–10. https://doi.org/10.1109/WORKS49585.2019.00006

Sweenor, D., Hillion, S., Rope, D., Kannabiran, D., & Hill, T. (2020). ML Ops: Operationalizing Data Science. O'Reilly Media, Inc.

Tamburri, D. A. (2020). Sustainable MLOps: Trends and Challenges. 2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 17–23. https://doi.org/10.1109/SYNASC51798.2020.00015

Treveil, M., Omont, N., Stenac, C., Lefevre, K., Phan, D., & Zentici, J. (2020). Introducing MLOps. O'Reilly Media, Inc.

van den Heuvel, W.-J., & Tamburri, D. A. (2020). Model-Driven ML-Ops for Intelligent Enterprise Applications: Vision, Approaches and Challenges. In B. Shishkov (Org.), Business Modeling and Software Design (p. 169–181). Springer International Publishing. https://doi.org/10.1007/978-3-030-52306-0_11

Zhang, H., Li, Y., Huang, Y., Wen, Y., Yin, J., & Guan, K. (2020). MLModelCI: An Automatic Cloud Platform for Efficient MLaaS. Proceedings of the 28th ACM International Conference on Multimedia, 4453–4456. https://doi.org/10.1145/3394171.3414535

Zhou, Y., Yu, Y., & Ding, B. (2020). Towards MLOps: A Case Study of ML Pipeline Platform. 2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE), 494–500. https://doi.org/10.1109/ICAICE51518.2020.00102