Hildesheimer Informatik-Berichte

Leonhard Faubel, Klaus Schmid

Review Protocol:
A systematic literature review of MLOps

August 24, 2023

Report No. 1/2023, SSE 2/23/E
ISSN 0941-3014

Software Systems Engineering • Institut für Informatik
Universität Hildesheim •Universitätsplatz 1 • D-31134 Hildesheim

Abstract

MLOps have become an increasingly important topic in the deployment of machine learning in production. While *Machine Learning Operations* was predominantly used as a buzzword for methods in Machine Learning (ML) for the time being, since 2019, they are increasingly used in the context of deploying ML algorithms. This report is a protocol for a systematic literature review (SLR) that aims to determine the MLOps terminology and identify related activities. A further goal of the SLR is to identify where MLOps can be linked to classical software engineering. In addition, related automation techniques are considered.

The projected literature review aims to draw conclusions from papers that explicitly use the term *MLOps* or *Machine Learning Operations* with the objective to provide the necessary common baseline for future MLOps research and practice. This report thoroughly documents the SLR method, processes, and data material. We also gathered all relevant data to comprehend MLOps fully. Through our comprehensive analysis, we hope to provide valuable insights and recommendations for optimizing MLOps practices.

Keywords: MLOps, Machine Learning Operations, Machine Learning

Contents

Chapter 1

Introduction

Machine Learning Operations – MLOps for short – refers to practices, technologies, and tools to put ML into production. Here, DevOps principles are related to Machine Learning. Although MLOps have already become a frequent subject of research and industrial application, it is a topic that is still in its infancy and for which there is no clear definition. As yet, the term MLOps has not been clearly worked out and there are several definitions, each of which differs from the other. In addition, there are divergent ideas about which activities are associated with MLOps. Further, to build a system with ML components using MLOps, another question is how to combine MLOps with traditional software engineering. Lastly, it also makes sense to combine automation approaches in conjunction with MLOps in order to use ML more time efficiently.

The Objectives of the SLR are described in Chapter 2, while the review Questions are outlined in Chapter 3, and the projected Search is described in Chapter 4. Initial results are provided in Section 5. Finally, Secion 6 concludes.

Chapter 2

Objective

Still, in practice, the implementation of an ML model is often not carried out. However, if the model is deployed, it takes on average half a year. At the same time, ML is increasingly used in companies and the need for solutions is growing. The models are often not used in the real world. The reason for this is often the deployment in the real world with real data. Data Scientists know how to derive knowledge from business cases, collect data, recognize correlations in the information obtained and develop suitable ML models. Their work often ends here. Deploying these algorithms in practice in a running program requires more steps, which are often dealt with by groups of Data Scientists, Software Engineers, Operationalists, and other people. Often, such a team works together on putting the ML into production which may take months. One reason is ML models' dependency on data, models, code, and configuration. Here, MLOps approaches help to better deploy ML into running systems.

At NeurIPS 2015, Sculley et al. identified risk factors and technical debt in system development with ML and presented countermeasures to avoid them [SHG+15]. Although the measures described in this paper do not necessarily address system-level debt, this paper can be considered an initial resource that identifies the need for MLOps. Cross-dependencies between data, model, code development, and configuration are described and still are integral parts of the MLOps life-cycle which clearly depicts their dependency. If one of these changes, it may have an impact on the others.

Primarily, the benefits of MLOps occur when one needs to manage multiple models, their versions, and data sets, but also when retraining and continuous deployment of new models is needed [SHET21]. When talking about MLOps, we talk about different roles/actors, approaches, principles, technical frameworks, and tools. This SLR does not examine the roles and responsibilities of the various actors but does examine the current technical understanding of MLOps, as well as associated approaches and activities.

So far, there are no standards for interfaces, platforms, tools, and frameworks to be used with MLOps. Instead, each company builds their own platforms combining a variety of different specific tools.

Different terms exist that are not to be confused with MLOps:

DevOps for ML DevOps for ML can be sub-categorized into the terms MLOps (Machine Learning Operations), AIOps (Artificial Intelligence for IT Operations), and DataOps (Data Operations) [CDM19].

AIOps usually deals with using AI to develop and manage IT services [CDM19, LRL+21, DLH19]. However, in some cases, it is equated with MLOps [DSA22, PIP+19].

DataOps deals with the organization of data management, taking into account DevOps principles [CDM19].

The survey aims at giving an overview of existing MLOps definitions. Further, it seeks to entail the steps associated with MLOps. Another focus is the degree of automation of MLOps solutions. Since ML solutions are often bound to more extensive software systems, another aim is to examine the reference to SE activities.

Since the review does not consider roles and task distributions in teams with respect to MLOps, it lacks reference to personnel, and socio-technical aspects – it is purely technical and organizational.

Chapter 3

Review Questions

In this section, the full review questions in sentenced format are provided.

The research questions addressed by this study are:

RQ1. How are MLOps defined?

RQ2. What types of activities are available using MLOps?

RQ3. How do MLOps refer to SE activities?

RQ4. To which degree are MLOps automated?

The RQs are considering questions from different viewpoints using the *PICO framework* [K$^+$07, p.11]. The population is the application area which is MLOps. It is examined by RQ1. Here, the terminology is inspected. The intervention is the methodology, tools, technologies and procedures addressed by MLOps. RQ2 addresses those by inspecting the activities related to MLOps. A descriptive and quantitative analysis is conducted to find out about activities related to MLOps. RQ 3 addresses the comparison to general SE activities. The result of facilitating the development of ML systems with respect to the automation associated with MLOps is addressed in RQ4. For this reason, a descriptive and quantitative analysis will be conducted on automation.

Regarding question one, a concern may be that the term *Machine Learning Operations* in the sense of DevOps for ML was used only after Sculley et al. 2015 identified risk factors and technical debt in system development with ML. Nevertheless, for the sake of completeness, all MLOps mentions in the last ten years were included in the review.

Chapter 4

Method

This section describes the methods to perform the SLR including all the details to make it thorough, transparent, and reproducible. The review follows the guidelines proposed by Kitchenham [Kit04, KBB+09]. To examine the SLR for completeness, the *extended PRISMA checklist* of Page et al. is used [PMB+21].

The serarch process is described in Section 4.1. Elegibility criteria are listed in Section 4.1.1 Further, the search strategy is described in Section 4.1.2, the study selection process in Section 4.2, and the data collection process in Section 4.3. Finally, the data extraction is described in Section4.3.1.

4.1 Search Process

In an automatic search, publications in journals and conference proceedings from January 2012- May 2022 are identified using literature search engines. Initially, an iterative pilot search was conducted to figure out relevant search terms to find appropriate primary studies. The corresponding search engines are listed in Table 4.1 together with Acronyms and dates of the searches. ACM Digital Library is a full-text collection of articles published by the Association of Computing Machinery (ACM), including all magazines and conference articles. Further, it contains a bibliographic database containing publications from all major publishers of computing literature [ACM23]. IEEE Xplore digital library allows for discovery and access to journal and conference papers on computer science, electrical engineering, and electronics. It is a research database for articles published by the Institute of Electrical and Electronic Engineers (IEEE) and other publishers [IEE21]. Science Direct allows searching for scientific and medical publications. It provides access to a bibliographic database of the publisher Elsevier [Sci23]. Google Scholar is a search engine for full-text and metadata of scholarly literature including peer reviewed publications, books, theses, dissertations, pre-prints, and technical reports [Goo23]. These digital libraries were selected because they are the biggest and most common search engines for publications in software engineering.

4.1.1 Eligibility criteria

Included are peer-reviewed and published research studies, such as conference and journal papers using the term *MLOps* or *Machine Learning Operations*. We also included

Source	Acronym	Date
ACM Digital Library	ACM	2022/05/31
IEEE XPlore	IEEE	2022/05/24
Elsevier/Science Direct	ScienceDirect	2022/05/31
Google Scholar	GS	2023/01/04

Table 4.1: Sources.

articles where the terms above were not the main or only purpose of the article. All entries from January 2015 to May 2022 were considered here.

Excluded are duplicate versions of studies, studies in other languages than English, not peer-reviewed studies, books, grey literature, studies that just mention *MLOps* or *Machine Learning Operations* without explanation or using it, and studies without reference to the research questions, talks without available information like protocols or notes, and posters.

Also, studies with a different *MLOps* or *Machine Learning Operations* context were excluded. The reason is that *MLOps* term can have different meanings, e.g., *Minimum Linear Operations* or *Multi-objective Lexicographical Optimization Problem*. Also, the term *Machine Learning Operations* is frequently used instead of ML methods, processing operations, or for supporting methods for ML. Studies are grouped by assigning them to different RQs, based on their content.

4.1.2 Search Strategy

Processes and tasks conducted to answer the research questions are depicted in Figure 4.1. Since we assess primary studies, this study can be seen as a secondary study. Nevertheless, also secondary studies are taken into account, which provide overviews of MLOps in a different context which turns this study to a partly tertiary review.

The general search process begins with the definition of the review protocol. Then, the search engines to be used are selected. Keywords are defined that can be used to find the subject of the research question. The papers found are filtered based on inclusion and exclusion criteria and then used as input for the processes concerning the validation and research questions.

In the search, specific keywords are used, which are combined to form a search query based on the scope of the literature review: *("MLOps" OR "ML Ops" OR "ML-Ops" OR "Machine Learning Operations")*. The use of the term MLOps is relatively new and was used after the publication of Sculley et al. in 2015 [SHG+15]. However, for an SLR, at least the publications of the last 10 years are considered. These begin in 2012, which precedes 2015 and thus includes a potentially different use of the term MLOps. The field often started with the terms DevOps for ML or ML deployment. These terms are not considered here as the new term MLOps has superseded them and includes more modern approaches, ideas and fundamentals. The term "AIOps" is only considered for RQ2-4 in this review as it is a different term with various definitions.

The search is conducted differently for the specific search engines: At ACM Digital Library the search term "[All:"mlops"] OR [All: "ml ops"] OR [All: "machine learning operations"] AND [Publication Date:(01/01/2012 TO 05/31/2022)]" is used. Items were searched from "The ACM Guide to Computing Literature" which includes all

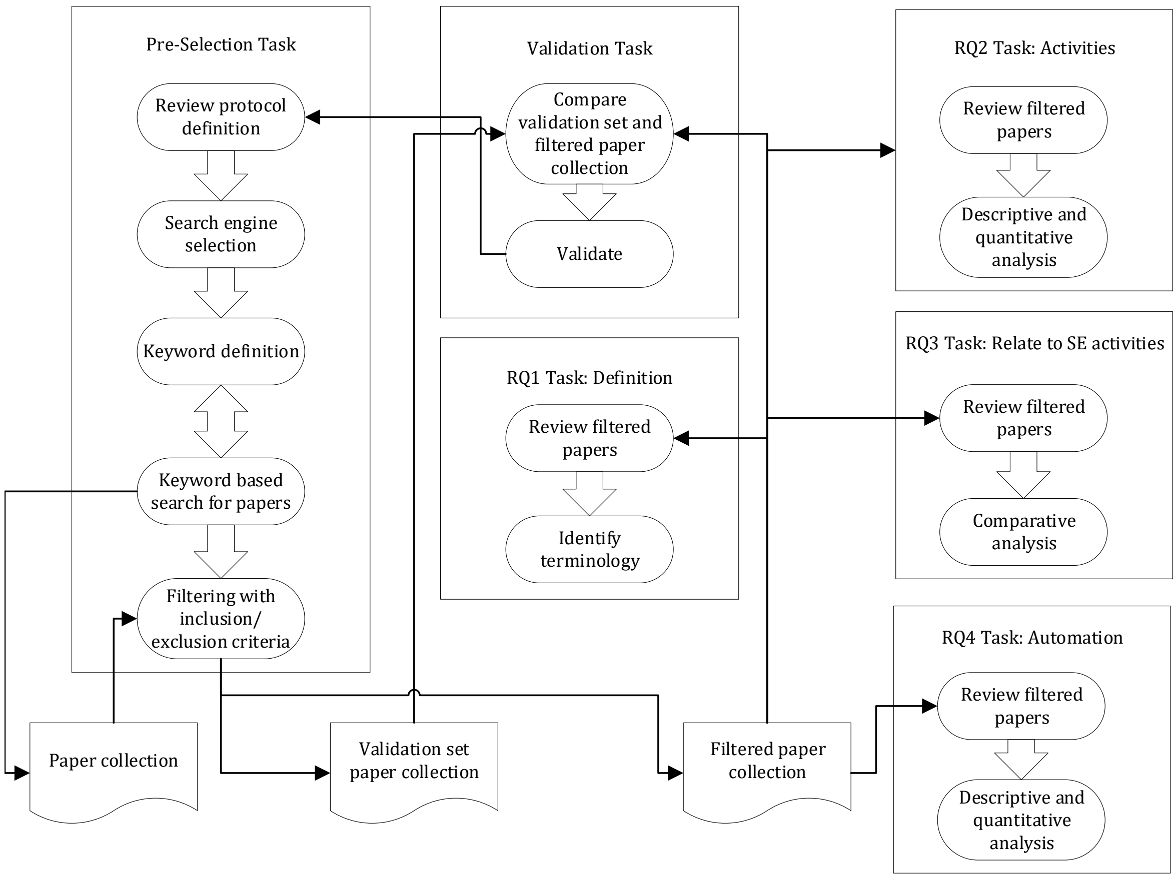

Figure 4.1: Processes.

possible databases including "The ACM Full-Text Collection". Using IEEE Xplore, All results between 01/01/1996 and 05/24/2022 that contain the search query above are included. At ScienceDirect, all results between 01/01/2012 and 05/31/2022 that contain the search query above are included.

As a research validation process, a search is performed using Google Scholar. In the search the search query above is used. The entries are sorted by relevance. Any types of entries are searched excluding patents and citations. A validation set is created by selecting the first 50 entries excluding publications that are not peer-reviewed, books, theses, technical reports, presentations, pre-prints, and papers in other languages than English. Papers from clinical journals are excluded as well. Only articles published between 01/01/2012 and 05/24/2022 or 05/31/2022, depending on the publisher (see search dates above) are selected. These articles were used to check whether most relevant studies found via Google Scholar were already covered by the previous search.

Zotero (version 6.0.18), an open-source and free of charge reference management tool was used to store the search results.

4.2 Study Selection Process

The selection process applied on these search results is performed in 5 steps.

Step 1: Remove duplicates

Step 2: Read titles, abstract, keywords

Step 3: Read introduction and conclusion

Step 4: Skim over whole text

Step 5: Read whole text

In step one, the reference management program *Zotero* is used to remove duplicates with the merge function. Further, remaining duplicates are manually excluded. Subsequently, the remaining papers are manually inspected regarding their titles, abstracts, and keywords. Inclusion and exclusion criteria are applied and eventually relevant papers are selected. In a next step, inclusion and exclusion criteria are applied to the selected papers considering the introduction and conclusion. Further, in a fourth stage, the exclusion criteria are applied on the whole text of the remaining papers. After applying the selection criteria in skimming over the whole text and diagonal reading, studies are finally included in the study.

4.3 Data Collection Process

Full-text-reading is conducted to collect the data from each study. If possible, the search results are saved as *CSV*-Files. *Zotero* is used to import the search results within the *CSV*-files. If a *CSV* export of search results is not supported by the search engine, the Browser Add-On *Zotero Connector* is used.

Data extracted from each study are:

- Item Type (Journal or conference paper).
- Publication year.
- The author(s).
- Title.
- Link to original article or DOI.
- The abstracts.
- The page references (if necessary).

4.3.1 Data Extraction

Two authors conduct a database search and review the results independently to ensure that the systematic review is thorough and accurate. They apply the inclusion and exclusion criteria to screen articles based on their title and abstract. If there is any uncertainty about whether an article meets the criteria, it will be included in the full-text review. The authors evaluate each paper that passes the initial screening to determine if it meets all the inclusion and exclusion criteria. The final decision on whether to include a paper in the systematic review will be made after the two researchers discuss each paper. If disagreements arise, a third author will be consulted to assess the article and decide independently.

Chapter 5

Initial Results

During the search, 296 search results were retrieved by the search engines *IEEE*, *ACM*, and *ScienceDirect*. At first, *Zotero* was used to remove 14 duplicates. Subsequently, the remaining 284 papers were manually inspected regarding their titles, abstracts, and keywords. No further duplicates were found in this step, but inclusion and exclusion criteria were applied, resulting in 121 potentially relevant papers. The inclusion and exclusion criteria were applied in the next step, considering the introduction and conclusion. Finally, in the fourth stage, the exclusion criteria were applied in skimming over the whole text and diagonal reading on the whole text of the 91 remaining papers. Finally, 69 studies were included. Figure 5.1 depicts the study selection process including progressive reduction of the number of studies throughout the process for the review. Pre-selected studies are listed in Section 5.1.1.

5.1 Pre-Selection

In an initial search, some search terms were tried. We realized that many papers use the *machine learning operations* term differently. Most papers use the term *machine learning operations* for machine learning algorithms, methods, or computing operations (51 paper). In the last three years, MLOps appeared more and more in the scientific literature. Now, most people talk about DevOps for ML when they talk about machine learning operations (see table 5.1). MLOps are not the main topic in most documents reviewed. They are rather mentioned or described as one of several relevant topics to ML platforms/tools, software systems using ML, or the usage of ML algorithms in SE. For example Srinivasan and Pruthi present a platform which includes MLOps principles [SP21]. MLOps platforms are presented by Zhou et al. (a case study of an MLOps platform) [YYB20], and Yan et al. (building an MLOps platform from open source frameworks) [LLH+20]. Oluyisola et al. propose a methodology for the design and development of smart production planning and control systems [OBSS22] and fluid software architecture is discussed by Yasser et al. [AM21]. However, there are a number of papers that deal directly with MLOps. A concise definition of MLOps functions and AI Software Sustainability is given by Tamburri [D. 20]. Further, he outlines vital challenges when dealing with MLOps. Mäkinen et al. discuss the importance of MLOps in the context of data scientists' activities based on a survey [SHET21]. Garg et al. view the ML lifecycle from a higher perspective, present differences between DevOps and MLOps, propose tools and techniques to execute a CI/CD pipeline for

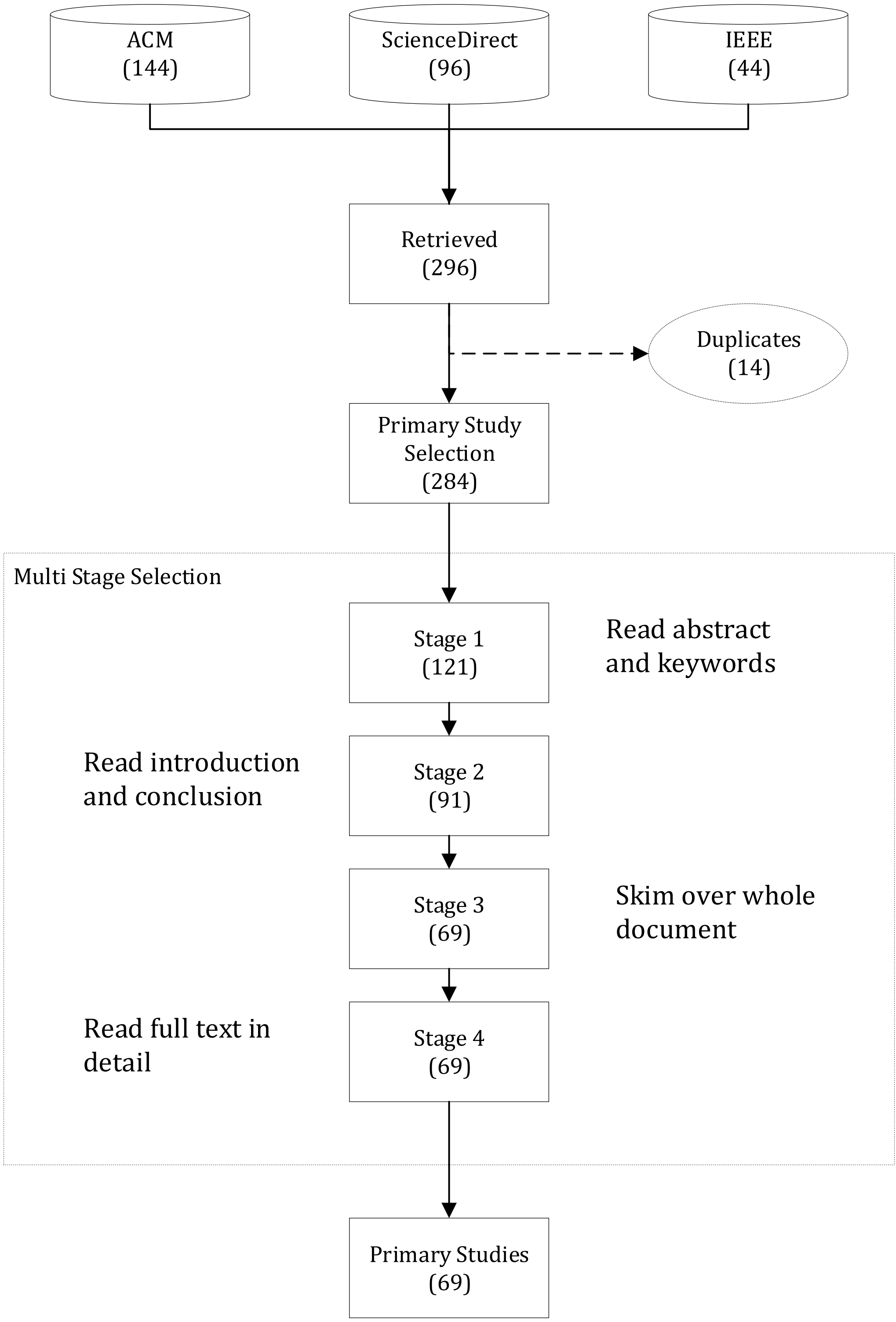

Figure 5.1: Study selection process including progressive reduction of the number of studies.

ML frameworks, and discuss challenges in this context [SPG+21]. Further MLOps definitions, tools, and challenges are discussed by Symeonidis et al. [GEAG22]. They also present an overview of the MLOps area, define the operations and components

Year	Journal	Conference	ML methods	MLOps	Total
2012	1	1	2	0	2
2014	1	1	2	0	2
2015	1	2	3	0	3
2016	3	1	4	0	4
2017	4	1	5	0	5
2018	3	1	4	0	4
2019	2	5	4	3	7
2020	7	10	13	4	17
2021	7	19	11	15	26
2022	9	5	3	11	14

Table 5.1: Distribution of the reviewed journal and conference papers with different usages of the Machine Learning Operations term (ML operations in the sense of ML methods vs. MLOps as DevOps for ML) over the publication years.

of MLOps solutions, and identify connections between MLOps and AutoML. Other papers describe MLOps in a specific context. For example, Raj et al. propose an edge MLOps framework for Artificial Intelligence of Things [EDMK21]. Granlund et al. consider MLOps Challenges in a multi-organization setup with restricted data access [TAV+21]. A method to manage the model evolution process in MLOps systems is described by Mei et al. [MLWS22]. The other 20 papers regard MLOps as background only.

In the sequel, we no longer consider work dealing with the notion of *machine learning operations* for algorithms, methods, or computational operations, since this is not concerned with putting ML models into production as is the modern MLOps notion.

5.1.1 Pre-Selected Studies

Table 5.2 presents the outcomes of the initial search procedure during the pre-selection task. Each paper is assigned to one of the four research questions, marked with an **X**. The included studies cover various topics related to MLOps, including stages and architecture, modeling and management of pipelines, challenges and maturity models, validation and deployment, and support for AI systems. They also discuss specific tools and methods for monitoring, workflow automation, edge deployment, and privacy-preserving architectures and APIs for putting ML into production. Additionally, XAI and I4.0 reference architectures are explored, along with the use of MLOps in autonomic computing and CD4ML pipelines. Overall, the papers provide a comprehensive overview of the current state of MLOps and its various applications and challenges.

Table 5.2: Pre-Selected Studies.

Title	RQ1	RQ2	RQ3	RQ4
Smart Cities Traffic Congestion Monitoring and Control System [OBT20]			X	
Designing and Developing Smart Production Planning and Control Systems in the Industry 4.0 Era: A Methodology and Case Study [OBSS22]	X		X	
Enabling Robot Selective Trained Deep Neural Networks for Object Detection Through Intelligent Infrastructure [PIP+19]	X	X		X
FinRL-Podracer: High Performance and Scalable Deep Reinforcement Learning for Quantitative Finance [LLZ+21]	X	X		
JIZHI: A Fast and Cost-Effective Model-As-A-Service System for Web-Scale Online Inference at Baidu [LGL+21]		X		
From ML Models to Intelligent Applications: The Rise of MLOps [Var21]	X	X		
You Do Not Need a Bigger Boat: Recommendations at Reasonable Scale in a (Mostly) Serverless and Open Stack [Tag21]		X		
Building a Reciprocal Recommendation System at Scale from Scratch: Learnings from One of Japan's Prominent Dating Applications [RSP20]			X	
End-to-End Machine Learning Using Kubeflow [GS22]	X	X		X
Enhancing Performance of Operationalized Machine Learning Models by Analyzing User Feedback [JR22]		X		
What Do People Really Want When They Say They Want "Explainable AI?" We Asked 60 Stakeholders. [Bre20]	X		X	
Scaling Enterprise Recommender Systems for Decentralization [vdG21]	X	X	X	
Model Provenance Management in MLOps Pipeline [MLWS22]	X	X	X	
Scarecrow - Intelligent Annotation Platform for Engine Health Management [SP21]		X	X	
A Report on the First Workshop on Changing Phases of Software Engineering with the Generations [Ms22]		X		
Machine Learning application lifecycle augmented with explanation and security [DS21]		X		
Deep Learning for B5G Open Radio Access Network: Evolution, Survey, Case Studies, and Challenges [BKA22]	X	X		X
MSR4ML: Reconstructing Artifact Traceability in Machine Learning Repositories [NAAF21]			X	

Continued on next page

Table 5.2 – continued from previous page

Title	RQ1	RQ2	RQ3	RQ4
Automating Tiny ML Intelligent Sensors DevOPS Using Microsoft Azure [VIC+20]	X	X		X
An Agile Software Development Life Cycle Model for Machine Learning Application Development [RA21]	X	X		
Benchmarking Machine Learning Solutions in Production [LFB+20]	X	X		
MLOps Challenges in Multi-Organization Setup: Experiences from Two Real-World Cases [TAV+21]	X	X		
Test Automation with Grad-CAM Heatmaps - A Future Pipe Segment in MLOps for Vision AI? [MRS+21]	X	X		
DeepEpil: Towards an Epileptologist-Friendly AI Enabled Seizure Classification Cloud System based on Deep Learning Analysis of 3D videos [TAC+21]	X	X		
Towards MLOps: A Case Study of ML Pipeline Platform [YYB20]	X	X	X	X
Edge MLOps: An Automation Framework for AIoT Applications [EDMK21]	X	X	X	
MLOps - Definitions, Tools and Challenges [GEAG22]	X	X		X
On Continuous Integration / Continuous Delivery for Automated Deployment of Machine Learning Models using MLOps [SPG+21]	X		X	
Who Needs MLOps: What Data Scientists Seek to Accomplish and How Can MLOps Help? [SHET21]	X	X		
Sustainable MLOps: Trends and Challenges [D. 20]	X	X		
Towards MLOps: A Framework and Maturity Model [MHJ21]	X			X
AI Lifecycle Models Need to Be Revised: An Exploratory Study in Fintech [HCHvD21]	X	X		
On-Premise AI Platform: From DC to Edge [IKKH19]	X	X	X	
Semantic Data Integration with DevOps to Support Engineering Process of Intelligent Building Automation Systems [MRM21]	X			
Identifying Roles, Requirements and Responsibilitiesin Trustworthy AI Systems [BA21]	X	X		
Characterizing Practices, Limitations, and Opportunities Related to Text Information Extraction Workflows: A Human-in-the-Loop Perspective [RK22]	X	X		
Apache submarine: a unified machine learning platform made simple [CSC+22]			X	
Recognition and Visualization of Facial Expression and Emotion in Healthcare [HRB+21]			X	

Continued on next page

Table 5.2 – continued from previous page

Title	RQ1	RQ2	RQ3	RQ4
A Classification and Review of Tools for Developing and Interacting with Machine Learning Systems [MRPARBB22]	X		X	
Towards Understanding End-to-End Learning in the Context of Data: Machine Learning Dancing over Semirings & Codd's Table [WZ21]	X	X		
Engineering Big Data Solutions [Moc14]		X		
Impact Factors and Best Practices to Improve Effort Estimation Strategies and Practices in DevOps [MT21]	X	X		
Try before You Buy: Privacy-Preserving Data Evaluation on Cloud-Based Machine Learning Data Marketplace [SCS+21]			X	
Dynamic Offloading of Web Application Execution Using Snapshot [JJM20]	X	X		
AMS: Generating AutoML Search Spaces from Weak Specifications [CCR20]		X		X
Adoption and Effects of Software Engineering Best Practices in Machine Learning [SvdBHV20]		X		
Who Needs to Know What, When?: Broadening the Explainable AI (XAI) Design Space by Looking at Explanations Across the AI Lifecycle [DWQ+21]		X		
A Review of Earth Artificial Intelligence [SSCO+22]	X	X		X
A Review on TinyML: State-of-the-Art and Prospects [Ray22]			X	
Automated Evolutionary Approach for the Design of Composite Machine Learning Pipelines [NVS+22]				X
An artificial intelligence life cycle: From conception to production [DSA22]	X	X		X
Intrusion and anomaly detection for the next-generation of industrial automation and control systems [RCF+21]			X	
Data Science Methodologies: Current Challenges and Future Approaches [MVGO21]		X		
Viral outbreaks detection and surveillance using wastewater-based epidemiology, viral air sampling, and machine learning techniques: A comprehensive review and outlook [ADH+22]	X		X	
A scalable deep learning system for monitoring and forecasting pollutant concentration levels on UK highways [AOB+22]	X		X	
FabOS: Towards an open, distributed, real-time-capable, and secure operating system for production [LSC21]	X	X	X	

Continued on next page

Table 5.2 – continued from previous page

Title	RQ1	RQ2	RQ3	RQ4
A machine learning approach on the relationship among solar and wind energy production, coal consumption, GDP, and CO2 emissions [MMS21]			X	
Building A Platform for Machine Learning Operations from Open Source Frameworks [LLH+20]	X	X		X
MLHarness: A scalable benchmarking system for MLCommons [CPHX21]	X		X	
Towards Fluid Software Architectures: Bidirectional Human-AI Interaction [AM21]	X		X	
GELAB – The Cutting Edge of Grammatical Evolution [KMA+22]	X			
Regression Test Selection Tool for Python in Continuous Integration Process [EJTM21]	X			X
Evidence-driven Requirements Engineering for Uncertainty of Machine Learning-based Systems [IM20]			X	
Drift Lens: Real-time unsupervised Concept Drift detection by evaluating per-label embedding distributions [GC21]	X			X
Monitoring the Emotional Response to the COVID-19 Pandemic Using Sentiment Analysis: A Case Study in Mexico [LSZBS+22]		X	X	
Data-centric Engineering: integrating simulation, machine learning and statistics. Challenges and opportunities [PMM22]				X
Smart mask – Wearable IoT solution for improved protection and personal health [HDH+22]			X	
AI for next generation computing: Emerging trends and future directions [GXO+22]		X		X

5.2 Data related to the Research Questions

In this section, the relevant data to answer the research question is summarized. Section 5.2.1 summarizes statements and definitions about MLOps. Activities are listed in Section 5.2.2, Section 5.2.3 shows how MLOps refer to SE activities. Finally, automation related data is summarized in Section 5.2.4.

5.2.1 RQ1 - How are MLOps defined?

The following statements and definitions about MLOps have been made in the literature:

- MLOps, which stands for Machine Learning Operations, is a specialized set of principles that combine ML methods with DevOps [CDM19, PIP+19, YYB20, vdG21, LLZ+21, RA21, MT21, WZ21, SHET21, SPG+21, EDMK21, MLWS22, OBSS22, BKA22, MRPARBB22, DSA22, GEAG22].

- Due to their cross-dependencies, this requires a holistic view of data, models, and code [SP21, MT21, MLWS22, OBSS22, BKA22, AOB+22, DSA22].

- MLOps is a collection of techniques, tools, practices, or processes for ML deployment in production [PIP+19, GEAG22, LLH+20, BA21, SP21, RA21, MT21, AM21, TAC+21, RK22, GS22, KMA+22, SSCO+22]. The deployment process can be done manually or automatically [SPG+21], but MLOps aims to increase automation [vdG21, MT21, EDMK21, BKA22, MRPARBB22, AOB+22, DSA22, GEAG22].

- One important property associated with MLOps is the reproducibility and repeatability of the models [IKKH19, vdG21], which creates the need to version data, models, code, and configurations [YYB20, vdG21, OBSS22, DSA22].

- MLOps is designed to lower the time-to-delivery significantly [LLH+20, vdG21, SHET21, SPG+21, KMA+22] and includes CI, CD [vdG21, GS22], and CT [RA21, TAC+21, TAV+21, MRPARBB22] as fixed parts [YYB20, MT21, SHET21, SPG+21, MLWS22, BKA22, AOB+22, GEAG22].

- In CI, changes made by developers are continuously integrated into a repository [CDM19]. A project build is then automatically executed, and the changes are integrated into the code via CD and published to the production environment.

5.2.2 RQ2 - What types of activities are available using MLOps?

The core activities commonly associated with MLOps involve a sequence of steps:

- Data collection [PIP+19, LFB+20, D. 20, YYB20, vdG21, LLZ+21, EDMK21, MRS+21, TAV+21, RK22, BKA22, SSCO+22, MRPARBB22, DSA22, GEAG22]

- Data analysis [PIP+19, LFB+20, D. 20, YYB20, RK22, BKA22, MRPARBB22, GEAG22]

- Data preparation [PIP+19, LFB+20, D. 20, YYB20, RK22, BKA22, MRPARBB22, DSA22, GEAG22]

- Model building [PIP+19, LFB+20, YYB20, LLH+20, D. 20, vdG21, LLZ+21, RA21, SHET21, SPG+21, EDMK21, MRPARBB22, BKA22, DSA22, GEAG22]

- Model training [PIP+19, LFB+20, LLH+20, D. 20, vdG21, LLZ+21, SPG+21, EDMK21, BKA22, SSCO+22, MRPARBB22, DSA22, GEAG22]

- Model evaluation [LFB+20, LLH+20, D. 20, vdG21, LLZ+21, RA21, SHET21, SPG+21, EDMK21, TAV+21, RK22, BKA22, SSCO+22, MRPARBB22, DSA22, GEAG22]

- Model selection [vdG21, LLZ+21, RA21, EDMK21, TAV+21, BKA22, SSCO+22, MRPARBB22, DSA22, GEAG22]

- Model packaging [PIP+19, LFB+20, LLH+20, D. 20, vdG21, SPG+21, EDMK21, TAV+21, BKA22, MRPARBB22, DSA22, GEAG22]
- Model deployment [PIP+19, LFB+20, D. 20, YYB20, vdG21, LLZ+21, RA21, SHET21, EDMK21, MRPARBB22, DSA22, GEAG22]
- Model monitoring [PIP+19, LFB+20, vdG21, EDMK21, TAV+21, RK22, BKA22, MRPARBB22, DSA22, GEAG22]

Furthermore, sporadic mentions in the literature include the following terms:

- Planning/analysis and design [RA21].
- Requirements engineering [RA21].
- Data cleaning [YYB20].
- Feature engineering, seen across multiple sources [LFB+20, LLZ+21, RK22].
- Division of data into training, testing, and cross-validation sets [SHET21, TAV+21].
- Hyperparameter tuning/optimization [SHET21, TAV+21, vdG21, RK22].
- Model registering [EDMK21].
- Algorithm configuration [LFB+20, vdG21].
- (Code) testing [vdG21, SHET21, TAV+21].
- (System) integration [TAC+21, TAV+21].
- Release processes [TAV+21].
- Infrastructure management [TAV+21].
- Output production/operation/inference [D. 20, LLZ+21, vdG21, BKA22].
- Versioning [DSA22, OBSS22, vdG21, YYB20].

5.2.3 RQ3 - How do MLOps refer to SE activities?

In modern software systems, machine learning (ML) components are often used to enable intelligent functionalities [SHET21]. However, these ML components are usually only a tiny part of the overall system. Therefore, the ML methods must interact seamlessly with the rest of the software. Cross-relationships between MLOps and software development activities, especially in DevOps, are critical in ensuring smooth integration and collaboration [RA21].

MLOps refers to integrating ML systems into the DevOps approach to enable continuous experimentation through a CI/CD pipeline [LLH+20]. Agile and lean software development technologies and tools from the DevOps context can be used here [LLH+20]. The link between MLOps and DevOps is through data and model flows and the combination of model publishing and deployment with the DevOps process. A significant point is the integration of models into the application repository, which provides a close link to software engineering.

It is worth mentioning that errors are not only caused by code but also by the experimental nature of model training [SPG+21]. This requires additional expenses in data preparation, algorithm selection, and feature engineering. High-performance hardware for parallel jobs leads to increased complexity, possible real-time requirements, and the need to transfer data for training and testing purposes. One solution is containerizing ML components to isolate them from the execution environment. Orchestration, in turn, enables automation of deployment and networking processes.

The integration of MLOps and DevOps is evident in several areas, including the testing process [YYB20]. While DevOps includes classic unit and integration testing, MLOps extends this approach to include data and model validation. The continuous testing (CT) aspect of MLOps requires triggers for model training or performance improvements based on new data or a decrease in model performance.

MLOps life cycles with different loops have similarities and differences to the DevOps lifecycle. In [LLH+20] the MLOps cycle is be linked to the DevOps cycle to ensure a CI/CD workflow for intelligent applications. This interaction supports the seamless integration of ML components into software systems and promotes these components' continuous development, deployment, and improvement. Integrating ML components into software systems requires a close connection between MLOps and DevOps to ensure a smooth development, deployment, and operations process. This requires adjustments in traditional DevOps practices to account for the experimental nature of ML methods and the specific requirements of model training. An overview of different perspectives and approaches to MLOps lifecycles and pipelines presented in the referenced studies is provided below:

- Rahman characterizes the Machine Learning (ML) lifecycle without MLOps [RK22].
- Symeonidis introduces an MLOps cycle with 3 loops [GEAG22].
- Tamburri presents an MLOps cycle with 4 loops [D. 20].
- Ranawana describes an MLOps lifecycle similar to Microsoft's [RA21].
- De Silva discusses a kind of MLOps lifecycle [DSA22].
- Granlund presents an MLOps pipeline similar to the life-cycle approach, including Continuous Deployment for Machine Learning (CD4ML) [TAV+21].

5.2.4 RQ4 - To which degree are MLOps automated?

ML is characterized by various technologies, algorithms, tools, and libraries used in different phases of product development. The product pipeline extends from hardware to software, raw data storage to information processing, and web services to endpoint software. This complexity makes it unrealistic to manage all of these pieces manually [SSCO+22].

To address this challenge, the concept of MLOps was developed. MLOps aims to apply automation techniques to address the complexity of ML development and deployment [GEAG22]. To this end, principles and approaches from the DevOps domain are applied to ML activities [GEAG22, DSA22]. This includes continuous monitoring of data profiles and performance of deployed models, as well as automation of all steps in building deep learning systems, including data preparation, model training, evaluation, and validation [BKA22].

Automation of ML activities in MLOps can take several forms:

- **End-to-end pipeline automation:** where manual steps in the process chain are replaced by automation. This includes automated pipeline connecting services, for example, through continuous integration and continuous delivery (CI/CD) [vdG21].

- **Automation of individual steps:** This includes automating specific process steps, such as automated features, labeling, and model training [RA21].

- **Infrastructure automation:** the automation of infrastructure management that enables rapid provisioning of resources and environments [OBSS22].

Various triggers can trigger automation. For example, automatic retraining of models occurs when validation results are below a certain threshold [BKA22]. Monitoring models and data and reaching thresholds in model performance can also trigger automation [EDMK21, PIP+19]. Customer feedback can also serve as a trigger [AM21].

Automation can reach different levels, depending on the degree of implementation. These range from manual processes to full automation of model training, deployment, and monitoring [GEAG22, YYB20]. Google and Microsoft have each developed models for these levels of automation [GEAG22]. Automation includes technical aspects and considers legal requirements such as the General Data Protection Regulation (GDPR) [D. 20].

Various approaches and platforms are available for implementing automation. AutoML tools enable automatic tuning of hyperparameters and model creation [OBSS22]. Platforms such as *Google Cloud AutoML*, *Microsoft Azure AutoML*, and *Amazon SageMaker Autopilot* provide supporting capabilities for training, experimentation, hyperparameter optimization, and model deployment [GEAG22, LLH+20, LLZ+21].

Overall, MLOps aims to address the complexity of ML development and deployment through automation by applying proven DevOps principles and approaches. This enables organizations to become more efficient, scalable, and agile in delivering their ML projects.

Chapter 6

Conclusion

In conclusion, we have thoroughly documented the SLR method, processes, and data material and gathered all relevant data to understand MLOps comprehensively. However, we have yet to conduct a detailed evaluation as it is essential for us to thoroughly analyze the data before drawing conclusions. Once we have completed the review, we will be able to provide more insight and recommendations. Our commitment to a comprehensive analysis ensures that decisions or recommendations are well-informed and grounded in accurate information.

Acknowledgments

This work is partially supported by the project EXPLAIN, funded by the Federal Ministry of Education under grant 01|S22030E. Any opinions expressed herein are solely by the authors and not of the Federal Ministry of Education.

Bibliography

[ACM23] About ACM DL, 2023. [Online; accessed 2. Jan. 2023].

[ADH+22] Omar M. Abdeldayem, Areeg M. Dabbish, Mahmoud M. Habashy, Mohamed K. Mostafa, Mohamed Elhefnawy, Lobna Amin, Eslam G. Al-Sakkari, Ahmed Ragab, and Eldon R. Rene. Viral outbreaks detection and surveillance using wastewater-based epidemiology, viral air sampling, and machine learning techniques: A comprehensive review and outlook. *Sci. Total Environ.*, 803:149834, 2022.

[AM21] A. Yasser and M. Abu-Elkhier. Towards fluid software architectures: Bidirectional human-AI interaction. In *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 1368–1372, 2021.

[AOB+22] Taofeek D. Akinosho, Lukumon O. Oyedele, Muhammad Bilal, Ari Y. Barrera-Animas, Abdul-Quayyum Gbadamosi, and Oladimeji A. Olawale. A scalable deep learning system for monitoring and forecasting pollutant concentration levels on UK highways. *Ecological Informatics*, 69:101609, 2022.

[BA21] Iain Barclay and Will Abramson. Identifying roles, requirements and responsibilitiesin trustworthy AI systems. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers*, pages 264–271. Association for Computing Machinery, 2021.

[BKA22] B. Brik, K. Boutiba, and A. Ksentini. Deep learning for b5g open radio access network: Evolution, survey, case studies, and challenges. *IEEE Open Journal of the Communications Society*, 3:228–250, 2022.

[Bre20] Andrea Brennen. What Do People Really Want When They Say They Want \. In *CHI EA '20: Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–7. Association for Computing Machinery, New York, NY, USA, 2020.

[CCR20] José P. Cambronero, Jürgen Cito, and Martin C. Rinard. AMS: generating AutoML search spaces from weak specifications. In *ESEC/FSE 2020: Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of*

Software Engineering, pages 763–774. Association for Computing Machinery, New York, NY, USA, 2020.

[CDM19] Antonio Capizzi, Salvatore Distefano, and Manuel Mazzara. From DevOps to DevDataOps: Data management in DevOps processes. In *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment: Second International Workshop, DEVOPS 2019, Château de Villebrumier, France, May 6–8, 2019, Revised Selected Papers*, pages 52–62. Springer-Verlag, 2019.

[CPHX21] Yen-Hsiang Chang, Jianhao Pu, Wen-mei Hwu, and Jinjun Xiong. MLHarness: A scalable benchmarking system for MLCommons. *BenchCouncil Transactions on Benchmarks, Standards and Evaluations*, 1(1):100002, 2021.

[CSC+22] Kai-Hsun Chen, Huan-Ping Su, Wei-Chiu Chuang, Hung-Chang Hsiao, Wangda Tan, Zhankun Tang, Xun Liu, Yanbo Liang, Wen-Chih Lo, Wanqiang Ji, Byron Hsu, Keqiu Hu, HuiYang Jian, Quan Zhou, and Chien-Min Wang. Apache submarine: a unified machine learning platform made simple. In *EuroMLSys '22: Proceedings of the 2nd European Workshop on Machine Learning and Systems*, pages 101–108. Association for Computing Machinery, New York, NY, USA, 2022.

[D. 20] D. A. Tamburri. Sustainable MLOps: Trends and challenges. In *2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 17–23, 2020.

[DLH19] Yingnong Dang, Qingwei Lin, and Peng Huang. AIOps: real-world challenges and research innovations. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, pages 4–5. IEEE, 2019.

[DS21] Saikat Das and Sajjan Shiva. Machine Learning application lifecycle augmented with explanation and security. In *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pages 0171–0177. IEEE, 2021.

[DSA22] Daswin De Silva and Damminda Alahakoon. An artificial intelligence life cycle: From conception to production. *Patterns*, page 100489, 2022.

[DWQ+21] Shipi Dhanorkar, Christine T. Wolf, Kun Qian, Anbang Xu, Lucian Popa, and Yunyao Li. Who needs to know what, when?: Broadening the Explainable AI (XAI) Design Space by Looking at Explanations Across the AI Lifecycle. In *DIS '21: Proceedings of the 2021 ACM Designing Interactive Systems Conference*, pages 1591–1602. Association for Computing Machinery, New York, NY, USA, 2021.

[EDMK21] E. Raj, D. Buffoni, M. Westerlund, and K. Ahola. Edge MLOps: An automation framework for AIoT applications. In *2021 IEEE International Conference on Cloud Engineering (IC2E)*, 2021.

[EJTM21] E. Kauhanen, J. K. Nurminen, T. Mikkonen, and M. Pashkovskiy. Regression test selection tool for python in continuous integration process. In *2021 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 618–621, 2021.

[GC21] Salvatore Greco and Tania Cerquitelli. Drift Lens: Real-time unsupervised Concept Drift detection by evaluating per-label embedding distributions. In *2021 International Conference on Data Mining Workshops (ICDMW)*, pages 341–349. IEEE, 2021.

[GEAG22] G. Symeonidis, E. Nerantzis, A. Kazakis, and G. A. Papakostas. MLOps - definitions, tools and challenges. In *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0453–0460, 2022.

[Goo23] About Google Scholar, 2023. [Online; accessed 4. Jan. 2023].

[GS22] Johnu George and Amit Saha. End-to-end machine learning using kubeflow. In *5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD)*, CODS-COMAD 2022, pages 336–338. Association for Computing Machinery, 2022.

[GXO+22] Sukhpal Singh Gill, Minxian Xu, Carlo Ottaviani, Panos Patros, Rami Bahsoon, Arash Shaghaghi, Muhammed Golec, Vlado Stankovski, Huaming Wu, Ajith Abraham, Manmeet Singh, Harshit Mehta, Soumya K. Ghosh, Thar Baker, Ajith Kumar Parlikad, Hanan Lutfiyya, Salil S. Kanhere, Rizos Sakellariou, Schahram Dustdar, Omer Rana, Ivona Brandic, and Steve Uhlig. AI for next generation computing: Emerging trends and future directions. *Internet of Things*, 19:100514, 2022.

[HCHvD21] Mark Haakman, Luís Cruz, Hennie Huijgens, and Arie van Deursen. AI lifecycle models need to be revised: An exploratory study in Fintech. *Empirical Softw. Engg.*, 26(5), 2021.

[HDH+22] Jarkko Hyysalo, Sandun Dasanayake, Jari Hannu, Christian Schuss, Mikko Rajanen, Teemu Leppänen, David Doermann, and Jaakko Sauvola. Smart mask – Wearable IoT solution for improved protection and personal health. *Internet of Things*, 18:100511, 2022.

[HRB+21] Hayette Hadjar, Thoralf Reis, Marco X. Bornschlegl, Felix C. Engel, Paul Mc Kevitt, and Matthias L. Hemmje. Recognition and Visualization of Facial Expression and Emotion in Healthcare. In *Advanced Visual Interfaces. Supporting Artificial Intelligence and Big Data Applications*, pages 109–124. Springer, Cham, Switzerland, 2021.

[IEE21] About Content in IEEE Xplore, 2021. [Online; accessed 2. Jan. 2023].

[IKKH19] Bukhary Ikhwan Ismail, Mohammad Fairus Khalid, Rajendar Kandan, and Ong Hong Hoe. On-premise AI platform: From DC to edge. In *Proceedings of the 2019 2nd International Conference on Robot Systems and Applications*, ICRSA 2019, pages 40–45. Association for Computing Machinery, 2019.

[IM20] Fuyuki Ishikawa and Yutaka Matsuno. Evidence-driven Requirements Engineering for Uncertainty of Machine Learning-based Systems. In *28th International Requirements Engineering Conference (RE)*, pages 346–351. IEEE, 2020.

[JJM20] Hyuk-Jin Jeong, Inchang Jeong, and Soo-Mook Moon. Dynamic Offloading of Web Application Execution Using Snapshot. *ACM Trans. Web*, 14(4):1–24, 2020.

[JR22] Hemadri Jayalath and Lakshmish Ramaswamy. Enhancing Performance of Operationalized Machine Learning Models by Analyzing User Feedback. In *IVSP '22: Proceedings of the 2022 4th International Conference on Image, Video and Signal Processing*, pages 197–203. Association for Computing Machinery, New York, NY, USA, 2022.

[K+07] Staffs Keele et al. Guidelines for performing systematic literature reviews in software engineering, 2007.

[KBB+09] Barbara Kitchenham, O. Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. Systematic literature reviews in software engineering–a systematic literature review. *Information and software technology*, 51(1):7–15, 2009. ISBN: 0950-5849 Publisher: Elsevier.

[Kit04] Barbara Kitchenham. Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004):1–26, 2004.

[KMA+22] K. K. Gupt, M. A. Raja, A. Murphy, A. Youssef, and C. Ryan. GELAB – the cutting edge of grammatical evolution. *IEEE Access*, 10:38694–38708, 2022.

[LFB+20] L. Cardoso Silva, F. Rezende Zagatti, B. Silva Sette, L. Nildaimon dos Santos Silva, D. Lucrédio, D. Furtado Silva, and H. de Medeiros Caseli. Benchmarking machine learning solutions in production. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 626–633, 2020.

[LGL+21] Hao Liu, Qian Gao, Jiang Li, Xiaochao Liao, Hao Xiong, Guangxing Chen, Wenlin Wang, Guobao Yang, Zhiwei Zha, Daxiang Dong, Dejing Dou, and Haoyi Xiong. JIZHI: A Fast and Cost-Effective Model-As-A-Service System for Web-Scale Online Inference at Baidu. *arXiv*, 2021.

[LLH+20] Yan Liu, Zhijing Ling, Boyu Huo, Boqian Wang, Tianen Chen, and Esma Mouine. Building a platform for machine learning operations from open source frameworks. *3rd IFAC Workshop on Cyber-Physical & Human Systems CPHS 2020*, 53(5):704–709, 2020.

[LLZ+21] Zechu Li, Xiao-Yang Liu, Jiahao Zheng, Zhaoran Wang, Anwar Walid, and Jian Guo. FinRL-podracer: High performance and scalable deep reinforcement learning for quantitative finance. In *Proceedings of the Second ACM International Conference on AI in Finance*, ICAIF '21, pages 1–9. Association for Computing Machinery, 2021.

[LRL+21] Yingzhe Lyu, Gopi Krishnan Rajbahadur, Dayi Lin, Boyuan Chen, and Zhen Ming (Jack) Jiang. Towards a consistent interpretation of AIOps models. *ACM Trans. Softw. Eng. Methodol.*, 31(1), 2021.

[LSC21] Martin Lukas, Daniel Stock, and Akos Csiszar. FabOS: Towards an open, distributed, real-time-capable, and secure operating system for production. *Procedia CIRP*, 104:962–967, 2021.

[LSZBS+22] Edgar León-Sandoval, Mahdi Zareei, Liliana Ibeth Barbosa-Santillán, Luis Eduardo Falcón Morales, Antonio Pareja Lora, and Gilberto Ochoa Ruiz. Monitoring the Emotional Response to the COVID-19 Pandemic Using Sentiment Analysis: A Case Study in Mexico. *Comput. Intell. Neurosci.*, 2022:4914665., 2022.

[MHJ21] M. M. John, H. H. Olsson, and J. Bosch. Towards MLOps: A framework and maturity model. In *2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 1–8, 2021.

[MLWS22] Songzhu Mei, Cong Liu, Qinglin Wang, and Huayou Su. Model provenance management in MLOps pipeline. In *2022 The 8th International Conference on Computing and Data Engineering*, ICCDE 2022, pages 45–50. Association for Computing Machinery, 2022.

[MMS21] Cosimo Magazzino, Marco Mele, and Nicolas Schneider. A machine learning approach on the relationship among solar and wind energy production, coal consumption, GDP, and CO2 emissions. *Renewable Energy*, 167:99–115, 2021.

[Moc14] Audris Mockus. Engineering big data solutions. In *FOSE 2014: Future of Software Engineering Proceedings*, pages 85–99. Association for Computing Machinery, New York, NY, USA, 2014.

[MRM21] Iori Mitzutani, Ganesh Ramanathan, and Simon Mayer. Semantic data integration with DevOps to support engineering process of intelligent building automation systems. In *BuildSys '21: Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pages 294–297. Association for Computing Machinery, New York, NY, USA, 2021.

[MRPARBB22] Eduardo Mosqueira-Rey, Elena Hernández Pereira, David Alonso-Ríos, and José Bobes-Bascarán. A classification and review of tools for developing and interacting with machine learning systems. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, SAC '22, pages 1092–1101. Association for Computing Machinery, 2022.

[MRS+21] M. Borg, R. Jabangwe, S. Åberg, A. Ekblom, L. Hedlund, and A. Lidfeldt. Test automation with grad-CAM heatmaps - a future pipe segment in MLOps for vision AI? In *2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 175–181, 2021.

[Ms22] Srinivasan Ms. A Report on the First Workshop on Changing Phases of Software Engineering with the Generations. In *ISEC 2022: 15th Innovations in Software Engineering Conference*, pages 1–2. Association for Computing Machinery, New York, NY, USA, 2022.

[MT21] Dulani Meedeniya and Harshanika Thennakoon. Impact factors and best practices to improve effort estimation strategies and practices in DevOps. In *2021 The 11th International Conference on Information Communication and Management*, pages 11–17. Association for Computing Machinery, 2021.

[MVGO21] Iñigo Martinez, Elisabeth Viles, and Igor G. Olaizola. Data Science Methodologies: Current Challenges and Future Approaches. *Big Data Res.*, 24:100183, 2021.

[NAAF21] Aquilas Tchanjou Njomou, Alexandra Johanne Bifona Africa, Bram Adams, and Marios Fokaefs. MSR4ML: Reconstructing Artifact Traceability in Machine Learning Repositories. In *2021 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 536–540. IEEE, 2021.

[NVS+22] Nikolay O. Nikitin, Pavel Vychuzhanin, Mikhail Sarafanov, Iana S. Polonskaia, Ilia Revin, Irina V. Barabanova, Gleb Maximov, Anna V. Kalyuzhnaya, and Alexander Boukhanovsky. Automated evolutionary approach for the design of composite machine learning pipelines. *Future Gener. Comput. Syst.*, 127:109–125, 2022.

[OBSS22] Olumide Emmanuel Oluyisola, Swapnil Bhalla, Fabio Sgarbossa, and Jan Ola Strandhagen. Designing and developing smart production planning and control systems in the industry 4.0 era: A methodology and case study. *J. Intell. Manuf.*, 33(1):311–332, 2022.

[OBT20] Tamer Omar, Daniel Bovard, and Huy Tran. Smart Cities Traffic Congestion Monitoring and Control System. In *ACM SE '20: Proceedings of the 2020 ACM Southeast Conference*, pages 115–121. Association for Computing Machinery, New York, NY, USA, 2020.

[PIP+19] Christian Poss, Thomas Irrenhauser, Marco Prueglmeier, Daniel Goehring, Firas Zoghlami, Vahid Salehi, and Olimjon Ibragimov. Enabling robot selective trained deep neural networks for object detection through intelligent infrastructure. In *Proceedings of the 2019 4th International Conference on Automation, Control and Robotics Engineering*, CACRE2019, pages 1–6. Association for Computing Machinery, 2019.

[PMB+21] Matthew J. Page, David Moher, Patrick M. Bossuyt, Isabelle Boutron, Tammy C. Hoffmann, Cynthia D. Mulrow, Larissa Shamseer, Jennifer M. Tetzlaff, Elie A. Akl, Sue E. Brennan, Roger Chou, Julie Glanville, Jeremy M. Grimshaw, Asbjørn Hróbjartsson, Manoj M. Lalu, Tianjing Li, Elizabeth W. Loder, Evan Mayo-Wilson, Steve McDonald, Luke A. McGuinness, Lesley A. Stewart, James Thomas, Andrea C. Tricco, Vivian A. Welch, Penny Whiting, and Joanne E. McKenzie. PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *BMJ*, 372:n160, 2021.

[PMM22] Indranil Pan, Lachlan R. Mason, and Omar K. Matar. Data-centric Engineering: integrating simulation, machine learning and statistics. Challenges and opportunities. *Chem. Eng. Sci.*, 249:117271, 2022.

[RA21] R. Ranawana and A. S. Karunananda. An agile software development life cycle model for machine learning application development. In *2021 5th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI)*, 2021.

[Ray22] Partha Pratim Ray. A review on TinyML: State-of-the-art and prospects. *Journal of King Saud University - Computer and Information Sciences*, 34(4):1595–1623, 2022.

[RCF+21] Luis Rosa, Tiago Cruz, Miguel Borges de Freitas, Pedro Quitério, João Henriques, Filipe Caldeira, Edmundo Monteiro, and Paulo Simões. Intrusion and anomaly detection for the next-generation of industrial automation and control systems. *Future Gener. Comput. Syst.*, 119:50–67, 2021.

[RK22] Sajjadur Rahman and Eser Kandogan. Characterizing practices, limitations, and opportunities related to text information extraction workflows: A human-in-the-loop perspective. In *CHI Conference on Human Factors in Computing Systems*, CHI '22, pages 1–15. Association for Computing Machinery, 2022.

[RSP20] R. Ramanathan, Nicolas K. Shinada, and Sucheendra K. Palaniappan. Building a reciprocal recommendation system at scale from scratch: Learnings from one of Japan's prominent dating applications. In *RecSys '20: Proceedings of the 14th ACM Conference on Recommender Systems*, pages 566–567. Association for Computing Machinery, New York, NY, USA, 2020.

[Sci23] ScienceDirect.com | Science, health and medical journals, full text articles and books., 2023. [Online; accessed 2. Jan. 2023].

[SCS+21] Qiyang Song, Jiahao Cao, Kun Sun, Qi Li, and Ke Xu. Try before You Buy: Privacy-preserving Data Evaluation on Cloud-based Machine Learning Data Marketplace. In *ACSAC '21: Annual Computer Security Applications Conference*, pages 260–272. Association for Computing Machinery, New York, NY, USA, 2021.

[SHET21] S. Mäkinen, H. Skogström, E. Laaksonen, and T. Mikkonen. 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)*, pages 109–112, 2021.

[SHG+15] D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. Hidden technical debt in machine learning systems. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'15, pages 2503–2511. MIT Press, 2015.

[SP21] Deepak S. Srinivasan and Rohit Pruthi. Scarecrow - intelligent annotation platform for engine health management. In *The First International Conference on AI-ML-Systems*, AIMLSystems 2021, pages 1–4. Association for Computing Machinery, 2021.

[SPG+21] S. Garg, P. Pundir, G. Rathee, P. K. Gupta, S. Garg, and S. Ahlawat. 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)*, pages 25–28, 2021.

[SSCO+22] Ziheng Sun, Laura Sandoval, Robert Crystal-Ornelas, S. Mostafa Mousavi, Jinbo Wang, Cindy Lin, Nicoleta Cristea, Daniel Tong, Wendy Hawley Carande, Xiaogang Ma, Yuhan Rao, James A. Bednar, Amanda Tan, Jianwu Wang, Sanjay Purushotham, Thomas E. Gill, Julien Chastang, Daniel Howard, Benjamin Holt, Chandana Gangodagamage, Peisheng Zhao, Pablo Rivas, Zachary Chester, Javier Orduz, and Aji John. A review of earth artificial intelligence. *Comput. Geosci.*, 159, 2022.

[SvdBHV20] Alex Serban, Koen van der Blom, Holger Hoos, and Joost Visser. Adoption and Effects of Software Engineering Best Practices in Machine Learning. In *ESEM '20: Proceedings of the 14th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, pages 1–12. Association for Computing Machinery, New York, NY, USA, 2020.

[TAC+21] T. Karácsony, A. M. Loesch-Biffar, C. Vollmar, S. Noachtar, and J. P. S. Cunha. DeepEpil: Towards an epileptologist-friendly AI enabled seizure classification cloud system based on deep learning analysis of 3d videos. In *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 1–5, 2021.

[Tag21] Jacopo Tagliabue. You Do Not Need a Bigger Boat: Recommendations at Reasonable Scale in a (Mostly) Serverless and Open Stack. In *RecSys '21: Proceedings of the 15th ACM Conference on Recommender Systems*, pages 598–600. Association for Computing Machinery, New York, NY, USA, 2021.

[TAV+21] T. Granlund, A. Kopponen, V. Stirbu, L. Myllyaho, and T. Mikkonen. MLOps challenges in multi-organization setup: Experiences from two real-world cases. In *2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN)*, pages 82–88, 2021.

[Var21] Manasi Vartak. From ML models to intelligent applications: the rise of MLOps. *Proc. VLDB Endow.*, 14(13):3419, 2021.

[vdG21] Maurits van der Goes. Scaling enterprise recommender systems for decentralization. In *Fifteenth ACM Conference on Recommender Systems*, pages 592–594. Association for Computing Machinery, 2021.

[VIC+20] Chandrasekar Vuppalapati, Anitha Ilapakurti, Karthik Chillara, Sharat Kedari, and Vanaja Mamidi. Automating Tiny ML Intelligent Sensors DevOPS Using Microsoft Azure. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 2375–2384. IEEE, 2020.

[WZ21] Wentao Wu and Ce Zhang. Towards understanding end-to-end learning in the context of data: Machine learning dancing over semirings & codd's table. In *Proceedings of the Fifth Workshop on Data Management for End-To-End Machine Learning*, pages 1–4. Association for Computing Machinery, 2021.

[YYB20] Y. Zhou, Y. Yu, and B. Ding. Towards MLOps: A case study of ML pipeline platform. In *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, pages 494–500, 2020.