MLOps: A Guide to its Adoption in the Context of Responsible AI

Beatriz M. A. Matsui
Federal University of ABC - UFABC
Santo André, SP, Brazil
beatriz.mayumi@ufabc.edu.br

Denise H. Goya
Federal University of ABC - UFABC
Santo André, SP, Brazil
denise.goya@ufabc.edu.br

ABSTRACT

DevOps practices have increasingly been applied to software development as well as the machine learning lifecycle, in a process known as MLOps. Currently, many professionals have written about this topic, but still few results can be found in the academic and scientific literature on MLOps and how to to implement it effectively. Considering aspects of responsible AI, this number is even lower, opening up a field of research with many possibilities. This article presents five steps to guide the understanding and adoption of MLOps in the context of responsible AI. The study aims to serve as a reference guide for all those who wish to learn more about the topic and intend to implement MLOps practices to develop their systems, following responsible AI principles.

CCS CONCEPTS

• **Software and its engineering** → **Software creation and management**; • **Computing methodologies** → *Machine learning*.

KEYWORDS

MLOps, DevOps, responsible AI, machine learning, model, development

ACM Reference Format:
Beatriz M. A. Matsui and Denise H. Goya. 2022. MLOps: A Guide to its Adoption in the Context of Responsible AI. In *Workshop on Software Engineering for Responsible AI (SE4RAI'22), May 19, 2022, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3526073.3527591

1 INTRODUCTION

In software development and engineering, DevOps practices have become increasingly popular. Although there is no standard definition for the concept of DevOps – a term that encompasses the union of *development* and *operations* – it can be understood as a culture of collaboration that aims to automate the continuous delivery of software, providing more speed, stability and reliability in the software delivery process, as well as value for end users [6, 15].

In the artificial intelligence (AI) world, when discussing the development of applications that involve machine learning (ML) – such as on software fault prediction [20] or recommendation systems [22] – we can also see the growing adoption of DevOps, in a practice known as MLOps [2]. Like DevOps, MLOps also does not have a formal definition; nonetheless, in [2] the authors explain it as being "an intersection between machine learning and DevOps practices", and [38] defines it as "the standardization and streamlining of machine learning lifecycle management".

When considering why research on MLOps is relevant, we can find some reasons in [32] – a 2015 study which, despite not formally introducing the term "MLOps", already discussed the reasons to reflect on good planning of the lifecycle of machine learning models. More and more organizations are thinking about this topic as it leads to a better implementation of ML models and more long-term business value, also reducing risks in AI processes [38]. While it is not a trivial task to develop an ML model to achieve a business objective (related to item classification or value prediction, for example), implementing and operating this model in production is really the biggest challenge. According to Algorithmia's *2020 State of Enterprise Machine Learning* report [1], only 22% percent of the companies surveyed are at an early stage (between 1 and 2 years) of deploying ML models into production. For mid-stage adopters (between 2 and 4 years), this rate drops to 15% and, for companies with more than five years of experience, this number is even lower: 8%. The main challenges pointed out by the report for an end-to-end deployment of ML models are version control, model reproducibility and scalability, and alignment with stakeholders.

Furthermore, an ML system has complex components not restricted to its code, which leads to the fact that developing and operating such systems is much more ingenious than it may seem [32]. In this context, responsible AI [39] appears as something essential and non-negotiable when working with artificial intelligence and machine learning. It consists of developing and deploying AI solutions with fairness, inclusiveness, human centeredness, robustness, transparency, explainability, privacy, and security [30, 33, 35]. MLOps practices then emerge, contributing to a necessary pattern of practices and processes to guide ML systems' development, deployment, and operations.

Although it is a topic on the rise [37], especially for large technology companies, there are still few scientific studies on MLOps [11], since it is a recent concept. When they exist, they are mostly focused on specific aspects rather than broader characteristics of MLOps itself, as in [26], which mainly addresses CI/CD concepts in MLOps for AIoT (Artificial Intelligence of Things) applications, or [29], which focuses on MLOps tools. Our work aims to fill this gap, serving as a useful, concise, and reliable reference source for those planning to implement MLOps in their systems development. It is not intended to be a technical tutorial with steps to follow when deploying ML models into production, or even to serve as an extensive guide to responsible AI practices, but rather to provide more theoretical scientific information to guide future research and practice. It is also worth mentioning that the objective is not for the steps to be followed in a linear fashion, as in a waterfall model

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SE4RAI'22 , May 19, 2022, Pittsburgh, PA, USA
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9319-5/22/05…$15.00
https://doi.org/10.1145/3526073.3527591

Authorized licensed use limited to: University College London. Downloaded on December 07,2022 at 12:44:37 UTC from IEEE Xplore. Restrictions apply.

[25], but rather to be understood as iterative processes that are in continuous development.

The paper has the following structure: first, we introduce the theoretical background of the theme (Section 2), including some examples of related work; then, we present the five steps we consider being essential for those who want to adopt MLOps practices effectively, considering responsible AI principles (Section 3); and finally, we present our closing remarks (Section 4), bringing a reflection on the contributions and limitations of the work, as well as ideas for future work on this topic for researchers and practitioners.

2 THEORETICAL BACKGROUND

MLOps is a recent term in the AI area that stands for *machine learning operations*. It means operationalizing the machine learning lifecycle [38], which includes steps such as data collection, data processing, model training, validation, testing, production deployment, and maintenance. More than just "DevOps applied to machine learning", MLOps brings broader concepts that involve all the complexity of developing machine learning systems, such as the constant change of data and business requirements, the lack of communication and cohesive teams, and the lack of knowledge of development practices and DevOps, which can generate an overload of responsibilities to data scientists [36, 38].

The origin of MLOps goes back to the 2015 paper *Hidden Technical Debt in Machine Learning Systems* [32], in which the authors discuss the challenges inherent in developing and deploying machine learning systems, which often incur increased technical debt and high maintenance costs. Since then, research in this area has only grown, bringing studies that investigate the application of MLOps in medical software development [9], manufacturing processes [16], or clinical data management and drug discovery [36]. Other studies such as [11], [12], and [27] can also be seen, which approach the concept of MLOps through tools and platforms that aim to automate the implementation of machine learning models.

On the other hand, studies that focus on responsible AI concepts are also recent, and grow as AI develops and evolves. Responsible AI is about how to design, develop, implement, and monitor the development of artificial intelligence solutions following ethical principles [30, 34]. Also known as "trustworthy" or "ethical" AI, it consists of a framework of good practices to guide the development, application, and use of AI by society [35], adhering to ethical and legal principles [17, 24]. MLOps emerges as a set of practices that contribute to the maintenance of responsible AI, as long as aspects such as the analysis of the data and algorithms used are followed, if the development complies with data laws and regulations, if there is a policy or governance to guide ethical development, and so on.

The process of implementing ML models essentially follows the steps of data collection and processing, training, validation, and testing [2, 38]. MLOps encompasses DevOps concepts to support all stages of the ML lifecycle [36], contributing to optimizing development, deployment, and operation of ML models in production. Figure 1 shows a representation of the MLOps lifecycle and its components.

Despite the recent growth of research in this area, our literature search revealed a lack of resources focusing on how best to adopt and implement these MLOps practices. Most studies (such as [7], [27], and [29]) refer to the application of MLOps to specific scenarios without highlighting how to deal with MLOps itself, especially with cultural aspects.

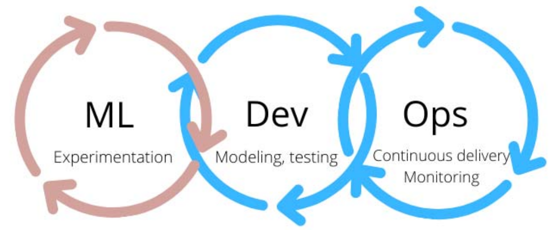

Figure 1: MLOps lifecycle. Representation of the MLOps lifecycle bringing its main components – starting with the data collection and model preparation phase (experimentation), followed by the development phase, with model training and testing (along with CI/CD practices) and, finally, the operations phase, which encompasses the continuous delivery of artifacts in production, and monitoring. The goal is to make the interconnection between ML and DevOps visually clear. Adapted from [36].

3 FIVE STEPS TO EFFECTIVELY ADOPT MLOPS

In this section, we introduce the five steps that, in our understanding, can lead to a more effective way of adopting and implementing MLOps – thus contributing to adherence to responsible AI principles.

3.1 Be Open to Learning and Changing the Culture

Several aspects need to be understood when considering a cultural change. Teams need to be open to learning and willing to change – and that goes for the adoption of MLOps, DevOps, and any other practice that disrupts the way systems and business are developed and managed. It is also essential to have good communication and understanding of responsibilities – by both individuals and entire teams.

It can seem difficult or even impossible to consider changing an organization's culture, especially in very traditional companies or companies that are not used to agile development practices or DevOps. However, when discussing cultural change, we are not necessarily referring to modifying the entire organization structure, including its values and core *modus operandi*. Instead, the ideal would be to look at existing structures and implement changes to them, showing the benefits that such changes could bring. An example would be making a team of data scientists that work isolated from other parts of the company has more contact with other strategic teams, as they should also be part of business decisions. Obtaining senior management sponsorship in this endeavor would also be critical to its success [38].

Authorized licensed use limited to: University College London. Downloaded on December 07,2022 at 12:44:37 UTC from IEEE Xplore. Restrictions apply.

A clear and well-structured communication plan would also contribute much to understanding the new practices and why they should be adopted. In line with this, training professionals in MLOps practices would be necessary to be on the same page and understand their roles in this process. In [38] the authors present a template for MLOps governance that includes aspects of understanding responsibilities, ethical position, education, policies, among others. Regarding the management of responsibilities, the authors suggest the RACI (acronym that stands for *responsible*, *accountable*, *consulted*, *informed*) matrix [14], encompassing the responsibilities of all stakeholders involved in the MLOps process. That could help to embrace discipline and alignment to the process. An AI governance that employs adherence to the rules and principles of responsible AI becomes indispensable in this context [18].

Another point to consider in the cultural aspect is the importance of having a clear definition of the business objectives that one wants to achieve, with a definition of metrics that would help accomplish and measure these goals. An example of a metric in MLOps would be RMSE (*root-mean-square error*) to compare performance between models, or cost-benefit evaluation, thinking from a direct business perspective [38]. For a better definition of these metrics, OKRs (*objectives and key results*) [23] could also be implemented, helping to more effectively measure the objectives, relate them to the key expected results, and track whether they are being achieved.

3.2 Prepare the Model Development

After having implemented a cultural shift within the teams, with a well-structured communication plan, and an understanding of individual and shared responsibilities, it is time to start developing the ML model that will be deployed to production at some point. (Note that this would be the ideal scenario, having a cultural and mindset shift first, but it is not mandatory to follow this step).

First, it is essential to understand what is an ML model. In machine learning, a *model* is a set of components trained to recognize a pattern based on some instructions. An ML model comprises training data, a performance metric, an ML algorithm, hyperparameters, and an evaluation dataset [38].

When considering each of these components, it is crucial to examine some challenges before starting the development. Challenges inherent to data models are related to the type of algorithm used (which can range, for instance, from linear regression to advanced neural networks), data dependencies [32], and continuous iteration processes [19], to name a few. Each model has its characteristics and can affect the MLOps cycle differently. The best way to address these challenges is to analyze the type of data used and understand what best suits its purpose. Having such an approach would benefit reducing dependencies and improving the process as a whole. Also, a good development that complies with responsible AI takes into account the use of ethical algorithms that follow principles of privacy, fairness, and explainability [13].

Regarding version control, unlike traditional software development that essentially leads only with the code, building ML systems encompasses, in addition to the code, the data, and the model itself. The practice of versioning these artifacts is recommended to ensure more traceability, maintenance, and prevent failures, making sure that the data is synchronized and up to date. Examples of situations in which this practice can help are related to model degradation and performance loss due to some update on the data or model, which version control makes it possible to revert and restore. Another example is preemptively identifying bugs and failures, which leads to faster feedback and remediation.

Another important aspect with which versioning can contribute is the reproducibility of the model. With data scientists constantly building, training, validating, testing, and iterating different versions of models [38], version control makes it possible to refer back to a previous experiment and replicate it for future needs. Arguably, there are differences between versioning for software (where everything is code-based) and versioning for ML-based applications. Nevertheless, the tools have evolved, and today it is possible to find options that help address this and other MLOps processes.

The following section delves into tools that can assist in these processes.

3.3 Choose the Right Tools

Along with good development comes choosing the right tools. This step is built into the others, particularly in model development, training, testing and validation, and production preparation. A good choice of tools is necessary to ensure that the development process is being followed in the best way, considering that data is constantly changing, and manual steps should be avoided to achieve more productivity.

The main aspects to consider when choosing tools for MLOps are related to the objectives – is the primary goal to implementing version control, focusing on CI/CD pipelines, or having a robust end-to-end platform to deal with all phases? The answers will guide the choice of the best tool according to the needs.

One example of a tool for MLOps orchestration is Kubeflow [3, 7]. Its history dates back to an internal Google project for performing ML pipelines on Kubernetes, until it became open source in 2017 [3]. It was based on TensorFlow Extended (TFX) – another platform for deployment of ML pipelines in production [8]. One downside of Kubeflow is that, as it focuses on making ML components deployable in Kuberbetes, it may not be the best choice if that is not your goal, and you are looking for broader possibilities of use.

Another example of tool is MLFlow [2]. Also open source, MLFlow is a platform designed to manage the end-to-end machine learning lifecycle [4]. It is possible to integrate it with other cloud platforms such as Databricks, Microsoft Azure, Google Cloud, and AWS SageMaker [2] for managing and deploying ML models.

By evaluating the objectives and aspects of the model, it is possible to make a correct choice of the best tool to use in MLOps, comparing the advantages and disadvantages of each one depending on the scenario. In [29], a comparison between different MLOps tools is presented, explaining their functionalities and mapping them to each phase of the MLOps workflow.

3.4 Automate the Pipelines

A critical phase of MLOps – its primary goal – is to get the ML model running in production effectively. To be able to achieve this goal, it is necessary to reflect on the automation of pipelines, which can be done through CI/CD practices [2, 28].

Authorized licensed use limited to: University College London. Downloaded on December 07,2022 at 12:44:37 UTC from IEEE Xplore. Restrictions apply.

Continuous integration (CI) is a DevOps practice that consists of having the development phase of the software fully automated, with the validation of code occurring as soon as changes are committed and merged in a version control system [21, 28]. *Continuous delivery* (CD), on the other hand, relates to the phase that comes right after: when there is a new build artifact (generated from CI), a release is triggered, and the artifact is deployed in the desired environment [21]. If the whole process is automated, then it is called *continuous deployment* [28].

In MLOps, we implement CI/CD to have all pipelines running in an automated process, guaranteeing that everything is thoroughly tested and validated [2]. From the experimental stage, pipelines can be packaged and deployed – ideally first in this test environment to ensure it is compatible and functional – and, if so, they are ready to be deployed to production [2]. This configuration promotes more agility and autonomy to the teams, allowing them to focus on other tasks such as developing or evaluating other models, knowing that the pipeline execution process is well-architected and fully automated. It also facilitates the creation of other pipelines when needed, which will continue to be created and improved.

In this context, the concept of *continuous training* (CT) [11] proves to be very important. Continuous training relates to the idea of bringing automation to the model training process so that it can be continuously retrained and prepared for redeployment. It reduces the need for manual intervention and helps to keep the ML system running up to date.

3.5 Monitor

After having implemented the ML model into production, it is vital to continue monitoring it to identify possible issues – such as the model degradation [38] – track its outcomes, prevent failures and thus avoid negative impacts on the business. It is time for the "Ops" to spring into action.

A known thing about ML models and data is that they are constantly changing, but this can happen unexpectedly and cause the model to become stale, resulting in less accurate outcomes – in a phenomenon called *data* or *concept drift* [10, 38]. One way to identify and minimize its negative impacts is through monitoring – a critical part of the MLOps lifecycle. Creating metrics and tracking changes in data, with the use of specific tools, makes it possible to identify or even isolate the root cause of the drift [5], automate and retrain the model with new data, or calibrate and rebuild it, contributing significantly to improving the model's performance in production.

An important aspect to consider is *continuous monitoring* (CM) [31], a concept that comes from the DevOps umbrella and allows the identification of risks and maintenance of the model in production, aligned with the business metrics. It may seem obvious that monitoring needs to be continuous, but the term helps to remember that development processes – and machine learning – are becoming increasingly agile and dynamic, leading to new monitoring and tracking changes approaches.

4 CONCLUSIONS, LIMITATIONS, AND FUTURE WORK

This work presented five steps to guide researchers and practitioners in understanding and adopting MLOps practices, in adherence and helping to ensure the practice of responsible AI principles. Starting from the theoretical context and background, we introduced the five steps: 1) be open to learning and changing the culture; 2) prepare the development of the model; 3) choose the right tools; 4) automate the pipelines, and 5) monitor, as a concise guide to effectively implement MLOps. We understand these steps as essential for those who want to start working with MLOps and those who already know machine learning but may be confused by this new concept.

Our work sought to be concise, compiling and bringing essential steps into an original format that was not as large as a book but not so small that it could not contain valuable information. We understand cultural aspects as the most important for the successful implementation of MLOps, and we tried to make that explicit here. Some limitations of the study are related to the fact that we did not evaluate the entire MLOps lifecycle, something that we could call an "MLOps governance". In this paper, our objective was to encapsulate the key concepts involved in the MLOps process and describe them as a guide to essential steps, without going into the scope of full lifecycle governance and how to assess whether these steps are being implemented effectively, which may be a topic for future work. We also did not delve too deeply into the concepts covered within responsable AI, this being a possibility for future work.

We believe that this subject will be increasingly researched in the coming years, considering the evolution of AI and machine learning systems. Research that considers a comparison between the adoption of MLOps by large and small companies, for example, is also very welcome. Our following projects will likely go in this direction, also analyzing MLOps research and practice status by organizations in our country.

REFERENCES

[1] Algorithmia. 2020. *2020 state of enterprise machine learning.* Retrieved January 22, 2022 from https://info.algorithmia.com/hubfs/2019/Whitepapers/The-State-of-Enterprise-ML-2020/Algorithmia_2020_State_of_Enterprise_ML.pdf.

[2] Sridhar Alla and Suman Kalyan Adari. 2021. What Is MLOps? In *Beginning MLOps with MLFlow.* Springer, Berkeley, CA, USA, 79–124.

[3] Ekaba Bisong. 2019. Kubeflow and kubeflow pipelines. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform.* Springer, 671–685.

[4] Andrew Chen, Andy Chow, Aaron Davidson, Arjun DCunha, Ali Ghodsi, Sue Ann Hong, Andy Konwinski, Clemens Mewald, Siddharth Murching, Tomas Nykodym, Paul Ogilvie, Mani Parkhe, Avesh Singh, Fen Xie, Matei Zaharia, Richard Zang, Juntai Zheng, and Corey Zumar. 2020. Developments in MLflow: A System to Accelerate the Machine Learning Lifecycle. In *Proceedings of the Fourth International Workshop on Data Management for End-to-End Machine Learning* (Portland, OR, USA) *(DEEM'20).* Association for Computing Machinery, New York, NY, USA, Article 5, 4 pages. https://doi.org/10.1145/3399579.3399867

[5] Microsoft Docs. 2021. *Detect data drift (preview) on datasets.* Retrieved January 25, 2022 from https://docs.microsoft.com/en-us/azure/machine-learning/how-to-monitor-datasets?tabs=python.

[6] Andrej Dyck, Ralf Penners, and Horst Lichter. 2015. Towards Definitions for Release Engineering and DevOps. In *2015 IEEE/ACM 3rd International Workshop on Release Engineering.* IEEE, 3–3. https://doi.org/10.1109/RELENG.2015.10

[7] Johnu George and Amit Saha. 2022. 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).* 336–338.

[8] Aurélien Géron. 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly

Authorized licensed use limited to: University College London. Downloaded on December 07,2022 at 12:44:37 UTC from IEEE Xplore. Restrictions apply.

Media.
[9] Tuomas Granlund, Vlad Stirbu, and Tommi Mikkonen. 2021. Towards Regulatory-Compliant MLOps: Orazivio's Journey from a Machine Learning Experiment to a Deployed Certified Medical Product. *SN Computer Science* 2, 5 (2021), 1–14.
[10] Syed Sajjad Hussain, Manzoor Hashmani, Vali Uddin, Tayyab Ansari, and Muslim Jameel. 2021. A Novel Approach to Detect Concept Drift Using Machine Learning. In *2021 International Conference on Computer Information Sciences (ICCOINS)*. 136–141. https://doi.org/10.1109/ICCOINS49721.2021.9497232
[11] Meenu Mary John, Helena Holmström Olsson, and Jan Bosch. 2021. Towards MLOps: A Framework and Maturity Model. In *2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. 1–8. https://doi.org/10.1109/SEAA53835.2021.00050
[12] Ioannis Karamitsos, Saeed Albarhami, and Charalampos Apostolopoulos. 2020. Applying DevOps practices of continuous automation for machine learning. *Information* 11, 7 (2020), 363.
[13] Michael Kearns and Aaron Roth. 2020. Ethical Algorithm Design. *SIGecom Exch.* 18, 1 (dec 2020), 31–36. https://doi.org/10.1145/3440959.3440966
[14] P.M. Khan and Kaleem A. Quraishi. 2014. Impact of RACI on Delivery and Outcome of Software Development Projects. In *2014 Fourth International Conference on Advanced Computing Communication Technologies*. 177–184. https://doi.org/10.1109/ACCT.2014.66
[15] Leonardo Leite, Carla Rocha, Fabio Kon, Dejan Milojicic, and Paulo Meirelles. 2019. A Survey of DevOps Concepts and Challenges. *ACM Comput. Surv.* 52, 6, Article 127 (nov 2019), 35 pages. https://doi.org/10.1145/3359981
[16] Junsung Lim, Hoejoo Lee, Youngmin Won, and Hunje Yeon. 2019. MLOp Lifecycle Scheme for Vision-based Inspection Process in Manufacturing. In *2019 USENIX Conference on Operational Machine Learning (OpML 19)*. USENIX Association, Santa Clara, CA, 9–11. https://www.usenix.org/conference/opml19/presentation/lim
[17] Qinghua Lu, Liming Zhu, Xiwei Xu, Jon Whittle, David Douglas, and Conrad Sanderson. 2021. Software engineering for responsible AI: An empirical study and operationalised patterns. *arXiv preprint arXiv:2111.09478* (2021).
[18] Qinghua Lu, Liming Zhu, Xiwei Xu, Jon Whittle, and Zhenchang Xing. 2022. Towards a Roadmap on Software Engineering for Responsible AI. *arXiv preprint arXiv:2203.08594* (2022).
[19] Lucy Ellen Lwakatare, Ivica Crnkovic, and Jan Bosch. 2020. DevOps for AI – Challenges in Development of AI-enabled Applications. In *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. 1–6. https://doi.org/10.23919/SoftCOM50211.2020.9238323
[20] Ruchika Malhotra. 2015. A systematic review of machine learning techniques for software fault prediction. *Applied Soft Computing* 27 (2015), 504–518. https://doi.org/10.1016/j.asoc.2014.11.023
[21] Beatriz M. A. Matsui and Denise H. Goya. 2020. Application of DevOps in the improvement of machine learning processes. In *IV Workshop @NUVEM*. Zenodo. https://doi.org/10.5281/zenodo.4318113
[22] Agata Nawrocka, Andrzej Kot, and Marcin Nawrocki. 2018. Application of machine learning in recommendation systems. In *2018 19th International Carpathian Control Conference (ICCC)*. IEEE, 328–331. https://doi.org/10.1109/CarpathianCC.2018.8399650
[23] Paul R Niven and Ben Lamorte. 2016. *Objectives and key results: Driving focus, alignment, and engagement with OKRs.* John Wiley & Sons.
[24] Dorian Peters, Karina Vold, Diana Robinson, and Rafael A. Calvo. 2020. Responsible AI—Two Frameworks for Ethical Design Practice. *IEEE Transactions on Technology and Society* 1, 1 (2020), 34–47. https://doi.org/10.1109/TTS.2020.2974991
[25] Kai Petersen, Claes Wohlin, and Dejan Baca. 2009. The waterfall model in large-scale development. In *International Conference on Product-Focused Software Process Improvement*. Springer, 386–400.
[26] Emmanuel Raj, David Buffoni, Magnus Westerlund, and Kimmo Ahola. 2021. Edge MLOps: An Automation Framework for AIoT Applications. In *2021 IEEE International Conference on Cloud Engineering (IC2E)*. 191–200. https://doi.org/10.1109/IC2E52221.2021.00034
[27] Cedric Renggli, Bojan Karlaš, Bolin Ding, Feng Liu, Kevin Schawinski, Wentao Wu, and Ce Zhang. 2019. Continuous Integration of Machine Learning Models with ease.ml/ci: Towards a Rigorous Yet Practical Treatment. In *Proceedings of Machine Learning and Systems 2019*. 322–333.
[28] Sander Rossel. 2017. *Continuous Integration, Delivery, and Deployment: Reliable and faster software releases with automating builds, tests, and deployment.* Packt Publishing Ltd.
[29] Philipp Ruf, Manav Madan, Christoph Reich, and Djaffar Ould-Abdeslam. 2021. Demystifying MLOps and Presenting a Recipe for the Selection of Open-Source Tools. *Applied Sciences* 11, 19 (Sep 2021), 8861. https://doi.org/10.3390/app11198861
[30] Nithya Sambasivan and Jess Holbrook. 2018. Toward Responsible AI for the next Billion Users. *Interactions* 26, 1 (dec 2018), 68–71. https://doi.org/10.1145/3298735
[31] Theo Schlossnagle. 2018. Monitoring in a DevOps world. *Commun. ACM* 61, 3 (2018), 58–61.
[32] David Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. 2015. Hidden technical debt in machine learning systems. *Advances in neural information processing systems* 28 (2015), 2503–2511.
[33] Ben Shneiderman. 2020. Bridging the Gap Between Ethics and Practice: Guidelines for Reliable, Safe, and Trustworthy Human-Centered AI Systems. 10, 4, Article 26 (oct 2020), 31 pages. https://doi.org/10.1145/3419764
[34] Ben Shneiderman. 2021. Responsible AI: Bridging from Ethics to Practice. *Commun. ACM* 64, 8 (jul 2021), 32–35. https://doi.org/10.1145/3445973
[35] Ryan Soklaski, Justin Goodwin, Olivia Brown, Michael Yee, and Jason Matterer. 2022. Tools and Practices for Responsible AI Engineering. *arXiv preprint arXiv:2201.05647* (2022).
[36] Ola Spjuth, Jens Frid, and Andreas Hellander. 2021. The machine learning life cycle and the cloud: implications for drug discovery. *Expert Opinion on Drug Discovery* (2021), 1–9.
[37] Damian A. Tamburri. 2020. Sustainable MLOps: Trends and Challenges. In *2020 22nd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*. 17–23. https://doi.org/10.1109/SYNASC51798.2020.00015
[38] Mark Treveil, Nicolas Omont, Clément Stenac, Kenji Lefevre, Du Phan, Joachim Zentici, Adrien Lavoillotte, Makoto Miyazaki, and Lynn Heidmann. 2020. *Introducing MLOps.* " O'Reilly Media, Inc.".
[39] Liming Zhu, Xiwei Xu, Qinghua Lu, Guido Governatori, and Jon Whittle. 2022. AI and Ethics—Operationalizing Responsible AI. In *Humanity Driven AI*. Springer, 15–33.

Authorized licensed use limited to: University College London. Downloaded on December 07,2022 at 12:44:37 UTC from IEEE Xplore. Restrictions apply.