# MLOps

Challenges and Opportunities

**Authors**

Gabriel Christensson

August Tengland

We certify that generative AI, incl. ChatGPT, has not been used to write this essay.
Using generative AI without permission is considered academic misconduct.

# 1 Introduction

The rapid adoption of DevOps (Development and Operations) principles in the world of software engineering has not gone unrecognized. These principles practically constitute a new IT paradigm spearheaded by a rising DevOps community to which organizations are progressively shifting. The reach of DevOps now spans multiple fields of computer science and companies in all lines of business are following the trend of automation. One of the fields in which automation is currently researched is Machine Learning (ML). ML systems can be difficult to maintain in production and may therefore benefit from increased automation and monitoring [1]. To operate such systems more efficiently both in production and development, MLOps (Machine Learning and Operations) was introduced as the practice of applying DevOps principles to ML systems. However, introducing new technologies and manners of working is not straightforward. It is therefore of interest to analyze and discuss the different challenges and opportunities associated with this transition.

# 2 Background

With the rise of ML-based applications such as speech recognition, product recommendations, and language translation the interest has surged for mechanisms that help make such systems more maintainable and operational. Some of the mechanisms may be classified as principles, others as tools. For this reason, terms such as MLOps can be ambiguous as what they refer to. In this section, a definition of MLOps and parts of the research, technologies, and principles related to it are presented.

## 2.1 Defining MLOps

Mark Treveil et al. defined MLOps as: "...the standardization and streamlining of machine learning life cycle management." [2, p. 4]. This definition explains the core idea while maintaining a comfortable generality. For the purpose of this essay, it is adopted here as well.

## 2.2 Demystifying MLOps

Before diving into the meaning of MLOps it is necessary to understand the principles of DevOps. Given the scope of this essay, it is assumed that the reader is familiar with them, and thus only the core DevOps concepts are revisited.

Practicing DevOps means having teams for development and operations (Dev and Ops) work together to build and deliver software products rapidly. Essentially, DevOps is all about automating the software development life cycle. Tasks that can be accomplished without human intervention such as testing, building, and deploying are automated to avoid introducing human errors and to give the Dev team more time to develop and the Ops team more time to monitor and maintain. Concurrently, the teams should collaborate and exchange feedback to improve future releases.

Practicing MLOps is practically the same as practicing DevOps but in the setting of developing/operating machine learning models. Of course, it differs somewhat from the traditional DevOps practices since there must be features specific to ML added to the mix. The idea of applying DevOps principles to ML systems emerged as these systems scaled and when a clear need for further automation was expressed.

### 2.3 The Need for MLOps

In 2014, data scientists at Google authored a publication in which they summarize multiple pitfalls related to operating ML systems [3], which may serve as a set of arguments for the transition to MLOps. One of the points made in the publication is that ML systems have a sizeable system-level complexity. The code for the ML model occupies only a small portion of the entire system and there are large components at play that make up the overall system operations. Thus, the interplay between all components is essential [4].

### 2.4 Bringing an ML Model To Production

In order to sufficiently grasp the challenges associated with MLOps one must understand the steps involved in bringing a machine learning model to production. Listed below is a road map of that process as originally described in [1].

1. Define use cases, business requirements, and success criteria.
2. **Data extraction:** Select data from relevant sources.
3. **Data preparation:** Clean the data, split data into training/validation/training sets, and transform data if necessary.
4. **Model training:** Use the prepared data to train different models and tune hyperparameters, and select the best-performing model.
5. **Model evaluation:** Evaluate the model quality by testing it on a holdout test set (data that has never been used in training).
6. **Model validation:** Validate that the model reaches established success criteria and is ample for deployment.
7. **Model serving:** Deploying the model.
8. **Model monitoring:** Continually monitor the performance of the model to eventually schedule a new iteration in the ML process.

## 3 Benefits of MLOps

## 4 Current Challenges in MLOps

While MLOps has undeniably advanced the development of adaptable models, and could be considered a *de facto* standard for organizations creating ML systems, there is still room for growth. Many MLOps pipelines, if not all, still require humans to a larger extent than typical DevOps systems. This is not surprising, as the nature of machine learning poses *unique* challenges when applying DevOps principles. Some of these differences include:

- *Data Handling:* MLOps pipelines need to manage not only system code but also data. The data might need to be cleaned, transformed, validated, and logged [1] [5].

- *Automatic Testing:* Testing models requires data and can be both difficult and costly [1] [6].
- *Concept Drift:* As the data distribution a ML model is trained on changes, the model will start to decay. Re-training is often necessary to maintain performance over time [1] [6].

Tackling these issues requires DevOps-tooling that is specific to the domain of MLOps. The rest of the section dives deeper into some of the challenges currently being tackled to advance MLOps:

## 4.1 Data Management and Quality

As data is the foundation for any machine learning system, it is important that the data collection and manipulation process is well integrated into the MLOps pipeline. A frequently occurring issue in MLOps systems is managing data in accordance with the different constraints and restrictions. The pipeline might, for example, be subject to local data regulations which require that data is contained within the organization [5]. This makes DevOps practices such as automation and cloud services difficult to apply in the data management stage as they must conform to these regulations (which might differ based on where the system is deployed). As such, there is a need for more sophisticated tooling which allows managing data in a "lawful" way throughout a distributed environment.

In addition to this, the quality of a machine learning model is generally decided, in large, by the quality of data that has been supplied to it [6]. Consequently, introducing data optimization techniques into the MLOps pipeline, such as data cleaning, is a promising way to increase model performance. Cleaning is however expensive and requires a degree of manual labor, creating a desire for systems that can help optimize this process. Solutions have been proposed that can analyze how noisy data *propagates* into the final model performance, providing a decision basis on how to conduct cleaning [6] [7]. However, we have yet to see systems expanded to a general setting where they could be implemented in an existing pipeline with arbitrary ML models, giving way for further research.

## 4.2 Automatic Testing

Nämn [6] task 3: Risk of overfitting when revealing test results

## 4.3 Continuous Training

Nämn [6] task 4: Risk of overfitting when revealing test results

Nämn triggers, när ska man uppdatera? Supervised learning, when do we know that we're drifting? [8]

GDPR

# References


[1] Google, *Mlops: Continuous delivery and automation pipelines in machine learning*. [Online]. Available: https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning.

[2] M. Treveil, N. Omont, C. Stenac, *et al.*, *Introducing MLOps*. O'Reilly Media, 2020.

[3] D. Sculley, G. Holt, D. Golovin, *et al.*, "Machine learning: The high interest credit card of technical debt," in *SE4ML: Software Engineering for Machine Learning (NIPS 2014 Workshop)*, 2014.

[4] 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)*, 2021, pp. 109–112. DOI: 10.1109/WAIN52551.2021.00024.

[5] M. Steidl, M. Felderer, and R. Ramler, "The pipeline for the continuous development of artificial intelligence models—current state of research and practice," *Journal of Systems and Software*, vol. 199, p. 111 615, May 2023. DOI: 10.1016/j.jss.2023.111615. [Online]. Available: https://doi.org/10.1016%5C%2Fj.jss.2023.111615.

[6] C. Renggli, L. Rimanic, N. M. Gürel, B. Karlaš, W. Wu, and C. Zhang, *A data quality-driven view of mlops*, 2021. arXiv: 2102.07750 [cs.LG].

[7] S. Krishnan, J. Wang, E. Wu, M. J. Franklin, and K. Goldberg, "Activeclean: Interactive data cleaning for statistical modeling," *Proc. VLDB Endow.*, vol. 9, no. 12, pp. 948–959, Aug. 2016, ISSN: 2150-8097. DOI: 10.14778/2994509.2994514. [Online]. Available: https://doi.org/10.14778/2994509.2994514.

[8] 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)*, 2022, pp. 0453–0460. DOI: 10.1109/CCWC54503.2022.9720902.