

MLOps, LLMOps, FMOps, and Beyond

Chakkrit (Kla) Tantithamthavorn^{ID}, Monash University

Fabio Palomba, University of Salerno

Foutse Khomh^{ID}, Polytechnique Montreal

Joselito (Joey) Chua^{ID}, Transurban

THE RISE OF machine learning operations (MLOps) represents a pivotal moment in the operationalization of machine learning (ML), enabling organizations to scale AI/ML systems more effectively, ensure continuous delivery, and maintain model performance and trustworthiness in production environments. This special issue has brought together insights from a range of experts, highlighting advancements in MLOps techniques, challenges in automating the ML lifecycle, and strategies for integrating trustworthy AI principles. As AI systems become more complex and integral to critical operations, the need for robust, scalable, and ethical MLOps pipelines will only increase. The contributions in this issue underscore the importance of cross-functional collaboration, automation, and ongoing innovation in building resilient ML infrastructures. Looking ahead, further research on MLOps standards, tooling, and best practices will be essential for driving the future of AI and ensuring its responsible and sustainable deployment across industries.

ML systems are more complex than traditional software due to their dependence on data, model behavior, and the need for continuous learning. Currently, managing ML systems has become highly challenging due to their inherent complexity. ML systems rely heavily on data, model behavior, and continuous learning, making them more intricate than traditional software. The lack of ML operation practices leads to difficulties in managing data pipelines, which are crucial for ensuring the quality, reliability, and trustworthiness of models. Manual deployment and monitoring of ML models are prone to errors and inefficiencies, increasing the risk of model failure.

Additionally, ML models can degrade over time due to data drift or changes in the environment, making ongoing monitoring essential. Collaboration across teams, including data scientists, ML engineers, DevOps teams, and business analysts, becomes fragmented without a unified framework, leading to misalignments and inefficiencies. Reproducibility is also a significant challenge as models are sensitive to the specific data and conditions under which they were trained, making it difficult to replicate results. To implement trustworthy AI principles, organizations must ensure fairness, transparency, accountability, robustness, privacy, and security throughout the ML lifecycle. This involves detecting and mitigating biases, for example, using explainability tools, establishing clear accountability, and performing rigorous testing for robustness.

Privacy-preserving techniques like differential privacy and federated learning can be employed, alongside strong data and model security measures. Monitoring for fairness and performance postdeployment is often required. Ethical considerations, sustainability, and efficient resource usage should also be integral to the process, supported by end-to-end governance and cross-functional collaboration. Over time, ML systems can accumulate technical debt, resulting in higher maintenance costs and reduced agility, further complicating the management of ML operations without MLOps.¹

Sculley et al.² defined the concept of ML technical debt as the long-term maintenance challenges and inefficiencies that accumulate in ML systems over time, which is similar to technical debt in software engineering (SE). In the context of ML, technical debt can manifest in several ways.

- *Code complexity:* Over time, as ML models are iteratively improved and expanded, the underlying codebase can become more complex, harder to understand, and more difficult to maintain.
- *Model decay:* ML models trained on historical data may become less accurate or relevant as the underlying data distribution changes (data drift), necessitating regular retraining or updates.
- *Data dependencies:* ML models often rely on specific data pipelines, and changes in these pipelines or the data themselves can lead to errors or degraded performance if not properly managed.
- *Testing and validation:* Inadequate testing of ML models, particularly in production environments, can lead to undetected issues that may surface only under specific conditions, increasing the risk of failure.
- *Model overfitting:* Developing models that are overly complex or tailored to specific datasets can result in models that don't generalize well to new data, requiring ongoing tuning or replacement.
- *Operational challenges:* Deploying and maintaining ML models in production environments often requires specialized infrastructure, monitoring, and scaling capabilities, which can add to technical debt if not properly managed.

Therefore, investing in reliable infrastructure for deploying and monitoring ML models in production could help teams reduce the long-term costs and risks associated with maintaining ML systems.

MLOps

MLOps is a set of practices and tools that combines ML and SE principles to facilitate the development,

deployment, and maintenance of ML models in production environments. MLOps aims to bridge the gap between data science and operations, enabling organizations to deliver reliable, scalable, and efficient ML solutions.

The MLOps lifecycle encompasses a series of stages that integrate ML development and operations to ensure the efficient deployment, management, and continuous improvement of ML models. Each stage of the MLOps lifecycle is described next.

Data Management

The MLOps lifecycle begins with robust data management, which is foundational for building effective ML models. This phase involves

collecting, cleaning, and preprocessing data to ensure that they are of high quality and relevant to the modeling task. Data management also includes implementing practices for version control, data lineage, and data governance, which are critical for maintaining the integrity and reproducibility of ML experiments. Proper data management facilitates seamless integration with training pipelines and supports the iterative nature of ML development, enabling teams to handle evolving datasets and maintain consistent model performance over time.

Model Development

Model development is the core of the MLOps lifecycle, where data

scientists and ML engineers create, train, and validate ML models (Figure 1). This phase involves selecting the appropriate algorithms, tuning hyperparameters, and evaluating model performance using metrics relevant to the specific problem domain. The development process also includes experimentation with different model architectures and techniques to optimize results.

Incorporating trustworthy AI practices is critical during this phase to ensure that models are not only effective but also ethical and reliable. This includes conducting fairness testing^{3,4} to identify and mitigate biases, performing robustness checks to ensure that the model performs consistently across diverse scenarios,

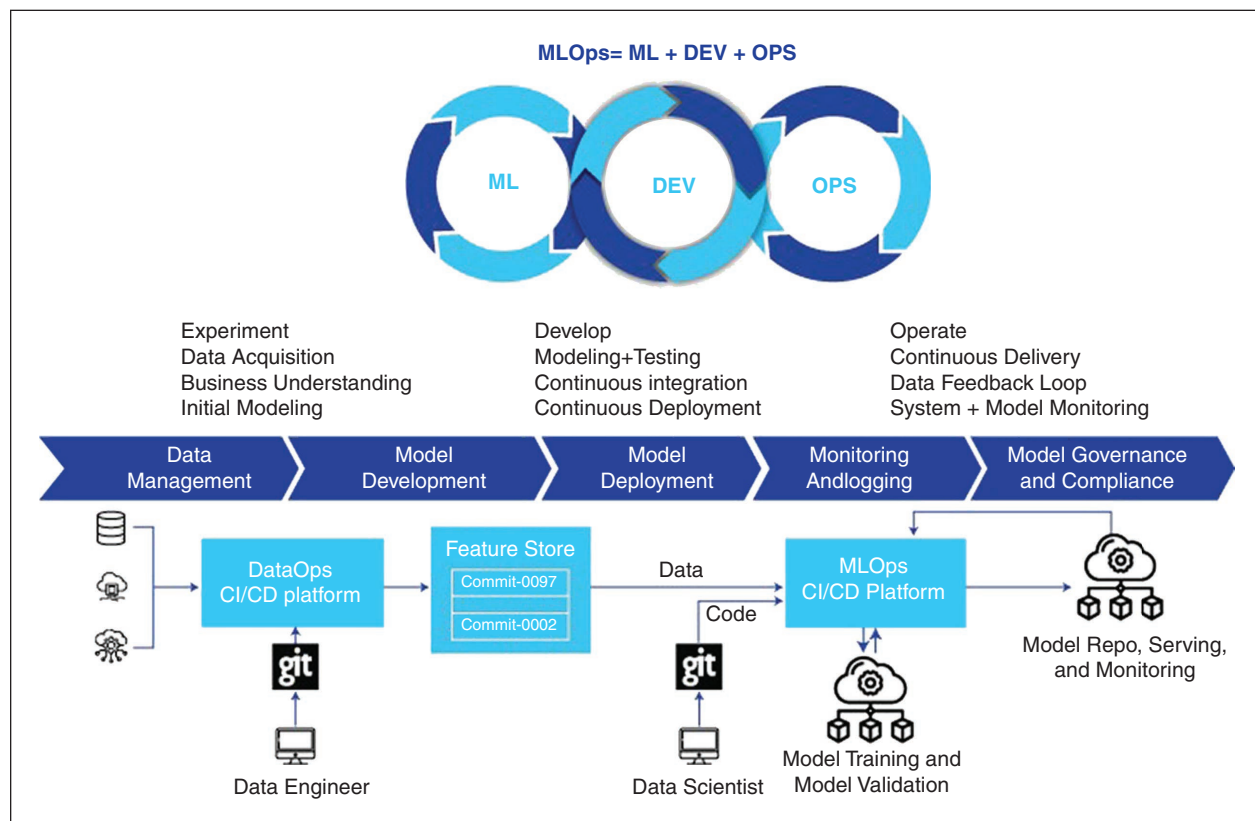


FIGURE 1. An example of the MLOps lifecycle. DataOps: data operations. (Source: <https://www.softwebsolutions.com/resources/mlops-architecture-guide.html>)

and validating the model's interpretability to make sure that its decisions can be understood and trusted by stakeholders.

The model development process is iterative and requires rigorous testing to ensure that the model generalizes well to unseen data. Effective version control and documentation of model parameters, configurations, and testing outcomes are essential to track the evolution of models, facilitate reproducibility, and ensure compliance with ethical standards.

Model Deployment

Once a model is developed and validated, it proceeds to the deployment phase, where it is integrated into production environments to deliver predictions or automate decision-making processes. Model deployment involves setting up the necessary infrastructure and ensuring that the model can scale to handle real-world workloads. This phase also includes integrating the model with existing systems and applications, managing dependencies, and ensuring compatibility with production environments. Deployment strategies may involve rolling updates, canary releases, or blue-green deployments to minimize risk and ensure a smooth transition from development to production. In the deployment phase, it is crucial to consider the sustainability of the model. This includes optimizing the model's energy consumption and carbon footprint, particularly when deployed on large-scale cloud infrastructures. Efficient resource utilization, such as selecting the appropriate hardware and optimizing data processing, can significantly reduce the environmental impact. Additionally, monitoring the model's performance over time and making adjustments to improve efficiency

can contribute to long-term sustainability. Incorporating sustainable practices into the deployment strategy not only aligns with environmental goals but also can reduce operational costs.

Model Monitoring and Maintenance

Monitoring and maintenance are critical to ensure that deployed models continue to perform effectively and reliably over time. This phase in-

practices ensure that ML models comply with legal requirements and industry regulations, such as data protection laws and algorithmic fairness guidelines. Compliance also includes maintaining documentation of model development processes, decision-making criteria, and impact assessments. Effective governance and compliance frameworks are essential for building trust in ML systems and mitigating risks associated with their



Model deployment involves setting up the necessary infrastructure and ensuring that the model can scale to handle real-world workloads.

volves tracking model performance metrics, monitoring for issues such as data drift or model degradation, and implementing automated alerting systems to detect anomalies. Regular maintenance includes retraining models with new data, updating model versions, and addressing any performance issues that arise. Continuous monitoring helps identify and mitigate risks associated with model performance, ensuring that models remain accurate and relevant as data and environments evolve.

Model Governance and Compliance

Governance and compliance are integral to the MLOps lifecycle, addressing the need for adherence to regulatory, ethical, and organizational standards. This phase involves establishing policies and procedures for data privacy, model transparency, and auditability. Governance

deployment and use. By adhering to this structured lifecycle, organizations can effectively operationalize ML systems, reduce ML technical debt, and enhance the reliability and scalability of their AI initiatives.

MLOps Maturity Models

The MLOps maturity model is a framework that helps organizations assess their current level of MLOps practices and provides a road map for evolving these practices over time. It typically consists of five stages, each representing a different level of maturity in data management, model development, model deployment, monitoring and logging, and model governance and compliance. Next are the typical stages in the MLOps maturity model.

- *Level 0—Ad hoc/manual process:* Generally, there are no

formal MLOps processes. Models are manually developed and deployed. There is also no version control for models or data with a high risk of errors and inconsistencies. Therefore, organizations tend to have difficulty reproducing results with a lack of monitoring and continuous improvement.

- **Level 1—Automated model training:** The model training pipelines become automated with the use of basic version control for code and models. However, the model deployment is still manual with a basic monitoring of model performance.
- **Level 2—Continuous integration/continuous deployment (CI/CD) for models:** There is the use of automated CI/CD pipelines for models, version control for data, models, and code, automated testing and validation of models, and automated deployment processes.
- **Level 3—Continuous training and automated deployment:** Models are retrained automatically based on new data with the automated validation and monitoring of models in production and scalable infrastructure that supports multiple models through advanced monitoring (for example, drift detection and performance degradation).
- **Level 4—Full MLOps automation:** There is an end-to-end automation of the entire ML lifecycle that is fully integrated with business processes. Real-time data pipelines are fed into model training. Governance and compliance are automated and embedded. At this level, one goal is usually to have the MLOps pipeline be capable of autonomously retraining

models, redeploying them, and adapting to changing environments or data without human intervention. Continuous learning can be applied where models are updated in real time as new data arrive.

Introduction to the MLOps Special Issue

The purpose of the MLOps special issue is to share the latest advancements, challenges, and solutions in the field of MLOps. By integrating MLOps into SE workflows, organizations can build and manage ML systems effectively and deliver value through ML applications. In this theme issue, we received a large number of submissions from more than 30+ authors. Each article underwent a competitive review process. Finally, we accepted seven articles. Next, we provide a summary of the accepted articles.

1. “MLOps for Developing Machine Learning-Enhanced Automotive Applications: Experience at the Bosch Cross-Domain Computing Solutions Division” by André L. Ferreira and João M. Fernandes

ML-enhanced system functions are now being developed and incorporated in automobiles—for example, scratch detection on the surface of cars based on audio data, impact event detection, acoustic anomaly detection, violence detection, etc. However, ML-based development activities become expensive if they are not automated. This article^{A1} presents motivations, challenges, and lessons learned using MLOps for developing ML-enhanced automotive applications at the Cross-Domain Computing Solutions Division (XC) at Bosch Car Multimedia S.A., Portugal.

2. “Establishing Machine Learning Operations for Continual Learning in Computing Clusters: A Framework for Monitoring and Optimizing Cluster Behavior” by Diana McSpadden, Mark Jones, Ahmed Hossam Mohammed, Bryan Hess, and Malachi Schram

Continual learning (CL) refers to a model's ability to utilize data that arrive incrementally. A challenge in CL is integrating new information and learning new tasks without forgetting previously acquired information, a phenomenon known as *catastrophic forgetting*. To achieve the Level-1 MLOps maturity model (see the “MLOps Maturity Models” section), this article^{A2} proposes an MLOps CL capability to support ML research and development projects for a computing cluster at the Thomas Jefferson National Accelerator Facility (Jefferson Lab).

3. “Machine Learning Lineage for Trustworthy Machine Learning Systems: Information Framework for MLOps Pipelines” by Mikko Raatikainen, Charalampos Souris, Jukka Remes, and Vlad Stirbu

ML applications are now becoming pervasive across diverse industries and have become increasingly complex and stringent, requiring transparency and evidence for trustworthiness. While emerging MLOps practices streamline the development and operations of ML-based systems, the required information remains disconnected and insufficiently addresses diverse concerns. This article^{A3} proposes the concept of ML lineage as a framework to holistically capture and connect the required information about ML model development and operations. ML

lineage fundamentally distinguishes between the model and prediction levels, conceptually encompassing separate yet interconnected core domains for the project, experiment, model, and prediction.

4. “Toward an Open Source MLOps Architecture” by Antonio M. Burgueño-Romero, Antonio Benítez-Hidalgo, Cristóbal Barba-González, and José F. Aldana-Montes

Current MLOps solutions are available in proprietary cloud services (for example, Azure ML and Amazon Sagemaker). However, they are generally costly—black boxes and not modular—leading to issues when changing project directions. This article^{A4} introduces a Kubernetes-based open source MLOps framework designed to streamline the lifecycle management of ML models in production environments. The implementation is publicly available at <https://github.com/KhaosResearch/mlops-infra>.

5. “MLOps for Cyberphysical Production Systems: Challenges and Solutions” by Leonhard Faubel, Thomas Woudsma, Benjamin Kloepper, Holger Eichelberger, Fabian Buelow, Klaus Schmid, Amir Ghorbani Ghezleghem-eidan, Leila Methnani, Andreas Theodorou, and Magnus Bång

Cyberphysical production systems (CPPSs) are the integration of computation with physical processes. In cyberphysical systems, physical and software components are deeply intertwined, are able to operate on different spatial and temporal scales, exhibit multiple and distinct behavioral modalities, and interact with each other in ways that change with context. Recently, ML has been widely used in

CPPSs—for example, automated visual inspection for electronics manufacturing and metal production and anomaly detection in chemical plants. However, such systems are not deployed on the cloud production environment, limiting the applicability of MLOps in CPPSs. This article^{A5} discusses the challenges and solutions of deploying MLOps for CPPSs.

6. “A State-of-the-Practice Release-Readiness Checklist for Generative AI-Based Software Products: A Gray Literature Survey” by Harsh Patel, Dominique Boucher, Emad Fallahzadeh, Ahmed E. Hassan, and Bram Adams

Generative AI, especially large language models (LLMs), is increasingly being integrated into software products, marking a significant shift in how companies approach product development and release. However, determining the release readiness of these products is increasingly complex. This article^{A6} investigates the complexities of integrating LLMs into software products (called *LLM-Ops*) with a focus on the challenges encountered in determining their readiness for release. This article identifies common challenges in deploying LLMs, ranging from pretraining and fine-tuning to user experience considerations. This article also introduces a comprehensive checklist designed to guide practitioners in evaluating key release readiness aspects such as performance, monitoring, and deployment strategies, aiming to enhance the reliability and effectiveness of LLM-based applications in real-world settings.

7. “Addressing a New Paradigm Shift in Data Science: An Empirical Study on Novel Project

Characteristics for Foundation Model Projects” by Sven Giesselbach, Dennis Wegener, Lennard Helmer, Claudio Martens, and Stefan Rüping

Foundation models are large-scale ML models, like GPT-4, Llama, and Gemini, that serve as the base model for various applications and can be fine-tuned or adapted for specific natural language understanding tasks. This article^{A7} presents the insights from 29 real-world applications of foundation model operations (FMOPs) across different industries and scopes, offering a holistic insight into the lists of success factors, challenges, and requirements that particularly influenced the project execution and outcome.

In this special issue, we provide a brief background knowledge of MLOps, LLMOPs, and FMOPs. Then, we introduce seven peer-reviewed articles, aiming to share the latest advancements, challenges, and solutions in the field of MLOps. While MLOps has shown a lot of promise in many modern organizations, there are various emerging challenges that remain largely unexplored. For example: How can we ensure the safety of LLM systems before and after deployment? How can we ensure that the LLM systems are not hallucinated (i.e., not generate a factually error response)? How can we integrate responsible AI principles throughout the LLMOPs pipeline? We look forward to seeing more publications addressing these important research questions in the future. 📄

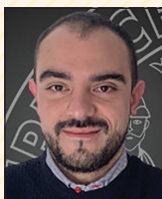
References

1. S. Amershi et al., “Software engineering for machine learning: A case

ABOUT THE AUTHORS



CHAKKRIT (KLA) TANTITHAMTHAVORN is a senior lecturer in software engineering at the Faculty of Information Technology, Monash University, Melbourne 3800, Australia. He is a Member of IEEE. Contact him at chakkrit@monash.edu



FABIO PALOMBA is an assistant professor at the Software Engineering Lab of the University of Salerno, 84084 Fisciano, Italy. Contact him at fpalomba@unisa.it.



FOUTSE KHOMH is a full professor of software engineering at Polytechnique Montréal, Montréal, QC H3T 1J4, Canada. He is a Senior Member of IEEE, a Canada Research Chair Tier 1 on Trustworthy Intelligent Software Systems, a Canada CIFAR AI Chair on Trustworthy Machine Learning Software Systems, an NSERC Arthur B. McDonald Fellow, an Honoris Genius Prize Laureate, and an FRQ-IVADO Research Chair on Software Quality Assurance for Machine Learning Applications. Contact him at foutse.khomh@polymtl.ca.



JOSELITO (JOEY) CHUA is currently the senior manager for advanced analytics at Transurban, 2008 Melbourne, Australia, where he leads the machine learning operations (MLOps) function. He is a Member of IEEE. Contact him at jchua@transurban.com.

study,” in *Proc. IEEE/ACM 41st Int. Conf. Softw. Eng.: Softw. Eng. Pract. (ICSE-SEIP)*, Piscataway, NJ, USA: IEEE Press, 2019, pp. 291–300, doi: [10.1109/ICSE-SEIP.2019.00042](https://doi.org/10.1109/ICSE-SEIP.2019.00042).

2. D. Sculley et al., “Hidden technical debt in machine learning systems,” in *Proc. 28th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2015, vol. 2, pp. 2503–2511.
3. A. Perera et al., “Search-based fairness testing for regression-based machine learning systems,” *Empirical Softw. Eng.*, vol. 27, no. 3, 2022, Art. no. 79, doi: [10.1007/s10664-022-10116-7](https://doi.org/10.1007/s10664-022-10116-7).

4. S. Galhotra, Y. Brun, and A. Meiliou, “Fairness testing: Testing software for discrimination,” in *Proc. 11th Joint Meeting Found. Softw. Eng.*, 2017, pp. 498–510, doi: [10.1145/3106237.3106277](https://doi.org/10.1145/3106237.3106277).

Appendix: Related Articles

- A1. A. L. Ferreira and J. M. Fernandes, “MLOps for developing machine-learning-enhanced automotive applications: Experience at the Bosch cross-domain computing solutions division,” *IEEE Softw.*, vol. 42, no. 1, pp. 34–41,

Jan./Feb. 2025, doi: [10.1109/MS.2024.3415750](https://doi.org/10.1109/MS.2024.3415750).

- A2. D. McSpadden, M. Jones, A. H. Mohammed, B. Hess, and M. Schram, “Establishing machine learning operations for continual learning in computing clusters: A framework for monitoring and optimizing cluster behavior,” *IEEE Softw.*, vol. 42, no. 1, pp. 42–50, Jan./Feb. 2025, doi: [10.1109/MS.2024.3424256](https://doi.org/10.1109/MS.2024.3424256).
- A3. M. Raatikainen, C. Souris, J. Remes, and V. Stirbu, “Machine learning lineage for trustworthy machine learning systems: Information framework for MLOps pipelines,” *IEEE Softw.*, vol. 42, no. 1, pp. 51–58, Jan./Feb. 2025, doi: [10.1109/MS.2024.3414317](https://doi.org/10.1109/MS.2024.3414317).
- A4. A. M. Burgueño-Romero, A. Benítez-Hidalgo, C. Barba-González, and J. F. Aldana-Montes, “Toward an open source MLOps architecture,” *IEEE Softw.*, vol. 42, no. 1, pp. 59–64, Jan./Feb. 2025, doi: [10.1109/MS.2024.3421675](https://doi.org/10.1109/MS.2024.3421675).
- A5. L. Faubel et al., “MLOps for cyberphysical production systems: Challenges and solutions,” *IEEE Softw.*, vol. 42, no. 1, pp. 65–73, Jan./Feb. 2025, doi: [10.1109/MS.2024.3441101](https://doi.org/10.1109/MS.2024.3441101).
- A6. H. Patel, D. Boucher, E. Fallahzadeh, A. E. Hassan, and B. Adams, “A state-of-the-practice release-readiness checklist for generative AI-based software products: A gray literature survey,” *IEEE Softw.*, vol. 42, no. 1, pp. 74–83, Jan./Feb. 2025, doi: [10.1109/MS.2024.3440190](https://doi.org/10.1109/MS.2024.3440190).
- A7. S. Giesselbach, D. Wegener, L. Helmer, C. Martens, and S. Rüping, “Addressing a new paradigm shift in data science: An empirical study on novel project characteristics for foundation model projects,” *IEEE Softw.*, vol. 42, no. 1, pp. 84–92, Jan./Feb. 2025, doi: [10.1109/MS.2024.3435024](https://doi.org/10.1109/MS.2024.3435024).