

MLOps: Unlocking Potential of Machine Learning Operations

Rajesh Choudhary,
 Senior Director, Data Science & AI – Axtria Inc.

Business Problem

Machine learning (ML) models must be optimally designed from development to deployment. Otherwise, they tend to be ineffective in production, and data quality issues become more pronounced, resulting in accelerated deterioration and inefficient monitoring and upkeep.

Organizations wanting to expand their use of artificial intelligence (AI) and ML across domains should prioritize the establishment of standards. Focusing only on sophisticated model development and relying solely on a select group of talented data scientists and their niche techniques is not enough to set up a reliable and repeatable ML foundation.

What is “MLOps?”

MLOps, short for Machine Learning Operations, is a set of practices and tools that streamline the end-to-end ML lifecycle, from model development to deployment to maintenance. It helps organizations automate and refine the ML process, enhancing stakeholder collaboration. MLOps aids in version control, model reproducibility, and scaling, ensuring faster and more reliable model deployment. That, in turn, leads to increased productivity and reduced operational risks.

Solution Approach

To solve these significant business hurdles, we need an end-to-end enterprise MLOps solution in the commercial pharma space.

Pain Points	Solution	MLOps Concepts
<p>Pain Point #1 The customer did not have a feature store in place; thus, features had to be re-engineered for every new model, leading to duplicated efforts, inconsistent features, and lack of discoverability and reuse across teams.</p>	<p>Solution built with Databricks feature store, providing the client with a centralized repository to manage features securely, enabling teams to discover and reuse features instead of duplicating efforts.</p> <p>Results: Improved efficiency and consistency across feature engineering</p>	<p>Feature Repository: Central storage for features such as columns in tables</p> <p>Feature Pipelines: Automated Extract Transform Load (ETL) jobs to populate features</p> <p>Feature Serving: Low latency serving of features to models.</p> <p>Feature Discovery: Search, browse, and retrieve features and their lineage</p>

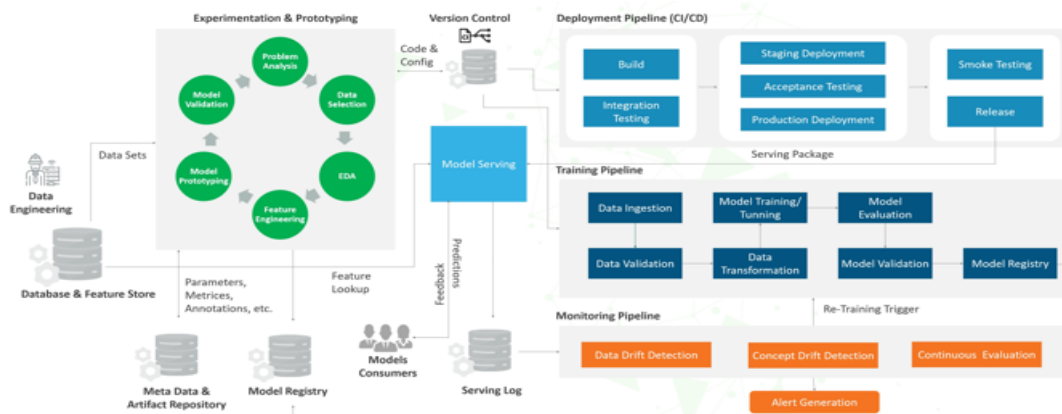
<p>Pain Point #2 The customer did not have model registry and logging. That provided limited visibility into model lineage, a lack of governance over model versions, difficulty with reproductions and model audits, and hampered collaboration.</p>	<p>Setup a MLflow model registry for the client, enabling centralized model versioning, staging, deployment, and lineage tracking. The registry streamlined collaboration through model governance, reproducibility, and auditability.</p> <p>Results: Enhanced collaboration and model governance due to centralized management</p>	<p>MLflow Tracking: Logging metrics, parameters, models, and artifacts in runs to record experiments.</p> <p>MLflow Projects: Packaging code, data, and config to reproduce runs.</p> <p>MLflow Models: Centralizing model repository for packaging and reuse</p> <p>MLflow Model Registry: Organizing, tracking, and staging models for deployment</p>
---	---	---

The Solution: Building an Integrated MLOps Workflow

Companies should include all the following steps in an MLOps pipeline. Today, most companies have limited capabilities in place. They might have Model Building and CI/CD or any other combination, but not the rest. Developed a scalable, more robust framework for our client partner that uses all these together.

1. Data and Code Versioning: Ensuring reproducibility and auditability in MLOps involves meticulous tracking of data, models, and code versions through version control systems such as Bitbucket.
2. Feature Store: Leveraging Databricks Feature Store, we have embraced a centralized repository that eases the discovery, reuse, and seamless sharing of features across teams. This approach promotes enhanced MLOps practices by effectively decoupling feature engineering from the model-building process.
3. Model Building: MLOps fosters the creation of reusable and reproducible models using workflow tools such as MLflow, with versioned model templates stored in repositories, effectively separating the modeling phase from deployment processes.
4. Model Logging and Registry: We used model registries like MLflow Model Registry, which provide versioning, model lineage, and governance for production models and to ease collaboration and automation of MLOps pipelines.
5. Continuous Integration/Continuous Deployment (CI/CD): We employed CI/CD best practices for machine learning with automated testing, continuous integration, and delivery of models and applications using tools like Jenkins.
6. Model Monitoring: Robust model monitoring through Dataiku helps keep model performance in production by tracking key metrics, data drift, and feature attribution.
7. Pipeline Orchestration: MLOps orchestrates end-to-end ML pipelines and workflows in Databricks to automate the flow of data, models, and reporting.
8. Testing Automation: MLOps requires extensive automated testing, including unit, integration, End-to-End (E2E), and concept drift tests using PyTest to keep model and data quality.

Key Tenets & Architecture of MLOps Implementation



Key Value Proposition of Building Enterprise grade MLOps framework for Data Science Teams

Without MLOps	With MLOps
Long Cycle Time – Average 6-month development cycle	Faster Time to Market – Reduction of up to 50% and cycles shortened by up to 3 months
Siloed Teams – Duplicate workflows per team hinders communication and knowledge sharing	Effective Collaboration – reduction of time spent in collaboration reduced by 50%
Complex Tools and Systems – Inefficient tools, performance bottlenecks on compute and manual code promotions cause waste of 20% efforts for each Data Science Project	Integrated Systems – Increased productivity of more than 30% with simplification of processes
Manual Steps in End-to-End Process – Multiple iterations needed due to manual steps resulting increased time to market for ML models	End to End Automation – Streamlines operations, reduces duplicate iterations, and shortens time to market
Manual Compute & Scaling – Limitations on computing for large scale datasets due to manual vertical scaling of EC2 machines	Scalable Distributed ML Compute – Databricks enabled efficient and scalable processing of ML jobs and enable horizontal auto scaling

MARKET TREND

According to a new report by MarketsandMarkets™, the MLOps market size is projected to grow from \$1.1 billion in 2022 to \$5.9 billion by 2027, at a CAGR of 41.0% across all industries.¹ Key players in the global MLOps market include Akira.AI, Alteryx, AWS, Blaize, ClearML, Cloudera, Comet, DataRobot, Datatron, and Domino Data Lab. According to a report by Deloitte, 64% of companies believe that AI enables a competitive advantage, 54% are spending four times as much as they did previously on AI initiatives, and 74% plan to integrate AI into all enterprise applications within three years.²

IMPACT

As artificial intelligence and machine learning (AI/ML) are adopted across industries at an increasing pace and scale, MLOps practices will become an essential part of any successful AI/ML program. Organizations must implement MLOps workflows and tools to operationalize models and manage them reliably throughout their lifecycle. Only through MLOps can companies sustainably scale AI/ML and achieve business impact while managing complexity, governance, and technical debt.

REFERENCES:

1. MarketsandMarkets. MLOps market size, share and global market forecast to 2027. December 2022. Accessed November 16, 2023. <https://www.marketsandmarkets.com/Market-Reports/mlops-market248805643.html>
2. Deloitte. Machine learning operations for business. May 19, 2021. Accessed November 16, 2023. <https://www2.deloitte.com/us/en/pages/consulting/articles/machine-learning-operations-forbusiness.html>