

SPRINTEINS

MLOPS

VON DER IDEE ZUM DEPLOYMENT

KI Summer Summit 2025
Hands-on Workshop

Kaywan Barzani / Ahmad Salah
DevOps / MLOps Engineer @ SprintEins

WORKSHOP-ÜBERBLICK

Ziel & Zielgruppe

WAS DU LERNEN WIRST:

- ML-Lebenszyklus verstehen und strukturieren
- MLflow sicher in eigene Projekte integrieren
- Unterschiede zu klassischer Softwareentwicklung erkennen

FÜR WEN:

- DevOps- & Data-Teams
- Data Scientists
- ML-Neugierige & Einsteiger im MLOps-Umfeld

Ablauf (2 Stunden)

- | | |
|--------|---|
| 15 MIN | EINFÜHRUNG MLOPS
Warum MLOps? Typische Stolpersteine |
| 30 MIN | MLFLOW BASICS
Tracking, Model Registry, Hands-on |
| 45 MIN | PRAXISBEISPIEL
Training, Logging, Auswertung, Deployment |
| 30 MIN | UND, WIE SIEHT'S BEI EUCH AUS?
Fakten statt Ausreden: Was hält euch wirklich von MLOps ab? |

Was ihr braucht

LAPTOP
mit Internet

PYTHON-UMGEBUNG
Jupyter Notebooks

PAKET-INSTALLATION
via pip

DOCKER DESKTOP
optional für Deployment

N

WER SIND WIR?

Kaywan Barzani

Lorem ipsum sit dolor

Ahmad Salah

Lorem ipsum sit dolor

WAS IST MLOPS?

MLOps ist die Brücke zwischen ML-Entwicklung und Produktion

Machine Learning Operations kombiniert bewährte Methoden aus DevOps, DataOps und Machine Learning, um den gesamten ML-Lebenszyklus zu optimieren.

MLOPS = DEVOPS + DATAOPS + MODELOPS
CI/CD + Continuous Training + Continuous Monitoring



DEVOPS

- Continuous Integration
- Continuous Deployment
- Automatisierung
- Versionskontrolle



DATAOPS

- Datenqualität
- Daten-Pipelines
- Datenversionierung
- Datenvalidierung



MODELOPS

- Modell-Training
- Modell-Deployment
- Modell-Monitoring
- Modell-Governance

WAS IST MLOPS?

HAUPTZIELE

- Schnellere Time-to-Market
- Höhere Modellqualität
- Bessere Skalierbarkeit
- Reduzierte Risiken

KERNIDEE

Struktur und Automatisierung in den ML-Lebenszyklus bringen – von der Entwicklung bis zur kontinuierlichen Überwachung in Produktion.

WARUM MLOPS?

TYPISCHE STOLPERSTEINE VON ML-PROJEKTEN

- Chaotische Notebooks
Unstrukturierter Code, keine Versionierung
- Verlorene Experimente
“Welche Parameter waren das nochmal?”
- Deployment-Probleme
“Auf meinem Laptop funktioniert es!”
- Keine Reproduzierbarkeit
Ergebnisse nicht nachvollziehbar
- Modell-Versionierung
Welches Modell läuft in Produktion?



MLOPS ALS LÖSUNG

- Strukturierte Workflows
Klare Prozesse und Standards
- Experiment Tracking
Alle Parameter und Metriken dokumentiert
- Automatisiertes Deployment
Konsistente Umgebungen
- Reproduzierbarkeit
Jedes Experiment nachvollziehbar
- Model Registry
Zentrale Modellverwaltung



WARUM MLOPS?

Der Unterschied zu klassischer Softwareentwicklung



DATENABHÄNGIGKEIT

Code + Daten + Parameter
= Model



EXPERIMENTELLER CHARAKTER

- Trial & Error
- Hyperparameter-Tuning



PERFORMANCE-DRIFT

Modelle verschlechtern
sich über die Zeit

MLOPS tOOL-LANDSCHAFT

Die richtige Auswahl für jeden Anwendungsfall



VENDOR END-TO-END

- AWS SageMaker
Vollständige ML-Plattform von Amazon
- Azure ML
Microsoft's Cloud ML-Service
- Google Vertex AI
Google's einheitliche ML-Plattform

✓ Vollständig integriert



OPEN SOURCE END-TO-END

- Kubeflow
ML-Workflows auf Kubernetes
- ClearML
Experiment Management & AutoML
- ZenML
Portable ML-Pipeline Framework

✓ Flexibel & kostenlos



SPEZIALISIERTE TOOLS

- Mlflow
Experiment Tracking & Model Registry
- DVC
Data Version Control
- Weights & Biases
Experiment Tracking & Visualization

✓ Best-of-Breed

WIE WÄHLE ICH DIE RICHTIGEN TOOLS?

TEAMGRÖSSE & EXPERTISE

- Kleine Teams → spezialisierte Tools
- Große Teams → Plattformen

KOMPLEXITÄT

- Einfache Projekte vs. Enterprise-Anforderungen

BUDGET

- Open Source vs. kommerzielle Lösungen

COMPLIANCE

- Datenschutz und Sicherheitsanforderungen

MLFLOW ALS PRAKTISCHE LÖSUNG

Open-Source-Plattform für den gesamten ML-Lebenszyklus

MLFLOW

Das Toolkit für reproduzierbare, skalierbare und kollaborative ML-Projekte

TRACKING

- Parameter loggen
- Metriken verfolgen
- Artefakte speichern
- Experimente vergleichen

MODELS

- Einheitliches Format
- Multi-Framework
- Deployment-ready
- Reproduzierbar

REGISTRY

- Zentrale Verwaltung
- Versionierung
- Staging/Production
- Governance

DEPLOYMENT

- REST APIs
- Docker Container
- Cloud Platforms
- Batch Inference

MLFLOW ALS PRAKTISCHE LÖSUNG

VORTEILE

- Open Source:
Kostenlos und erweiterbar
- Framework-agnostisch:
Funktioniert mit allem
- Einfache Integration:
Wenige Zeilen Code
- Skalierbar:
Von Prototyp bis Enterprise



PERFEKT FÜR HEUTE

MLflow löst genau die Probleme, die wir in unserem Hands-on erleben werden: Experimente nachverfolgen, Modelle verwalten und bereitstellen.

WIR WERDEN
ALLE VIER
KOMPONENTEN
PRAKTISCH
KENNENLERNEN!

HANDS-ON: WAS WIR GEMEINSAM BAUEN

Unser Praxisbeispiel: Iris-Datensatz

Iris Flower Classification

150 Datenpunkte • 4 Features • 3 Klassen

Warum Iris?

- Klassischer ML-Datensatz – jeder kennt ihn
- Klein und überschaubar
- Perfekt für MLOps-Konzepte
- Schnelle Trainingszeiten



SCIKIT-LEARN
Random Forest



MATPLOTLIB
Visualisierungen

Unser Workflow

- 1 Datenexploration
Iris-Datensatz laden und verstehen
- 2 MLflow Setup
Tracking-Server starten, erste Runs
- 3 Modelltraining
Random Forest mit verschiedenen Parametern
- 4 Experiment Tracking
Parameter, Metriken, Plots loggen
- 5 Model Registry
Bestes Modell registrieren und versionieren
- 6 Deployment
REST-API erstellen und testen

WAS WIR AM ENDE HABEN WERDEN



Strukturiertes
Notebook



Experiment
History



Registriertes
Modell



Deploybare
API

N

Los geht's!

Zeit für praktische Übungen



SCHNELLE CHECKLISTE

- ✓ Laptop bereit & Internet verbunden
- ✓ Python-Umgebung funktioniert
- ✓ Jupyter Notebook kann gestartet werden
- ✓ pip install bereit für MLflow & Co.

UNSER WEG

- 1 Jupyter Notebook öffnen
- 2 MLflow installieren & starten
- 3 Iris-Datensatz erkunden
- 4 Erstes Modell trainieren



JUPYTER
NOTEBOOKS
STARTEN!

Fragen? Probleme? Einfach melden!