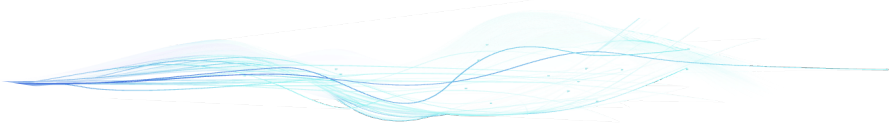
CONTECO

Workshop



Gen AI: MLOps & DevOps für KI-Projekte

Gen AI: MLOps & DevOps für KI-Projekte

Kursdauer
4 Tage

Zielgruppe
Machine Learning Engineers, DevOps-Spezialist:innen, Softwarearchitekt:innen und AI Product Owner, die skalierbare, wartbare und produktionsreife KI-Systeme bauen und betreiben möchten. Besonders relevant für alle, die den Übergang von experimentellen ML-Modellen zu robusten, kontinuierlich überwachten Services gestalten wollen.

Schulungsziel
Nach Abschluss dieser Schulung beherrschen die Teilnehmenden moderne MLOps-Techniken, um KI-Projekte effizient zu verwalten, automatisiert zu deployen und im Betrieb kontinuierlich zu überwachen. Sie kennen Best Practices zur Versionierung, Validierung und Governance von Modellen, wissen um die Bedeutung von CI/CD in der KI-Entwicklung und können Tools wie MLflow, DVC, FastAPI und Kubernetes sicher einsetzen. Ein besonderer Fokus liegt auf dem Zusammenspiel von LLM-basierten Pipelines und DevOps-Prinzipien.

Schulungsbeschreibung
Diese Schulung vermittelt, wie aus Machine-Learning-Prototypen produktionsreife KI-Systeme werden – zuverlässig, skalierbar und wiederholbar. Im Zentrum stehen CI/CD-Prinzipien für ML, automatisiertes Experiment-Tracking, Model Deployment, Monitoring sowie Data & Model Lineage. Zudem wird aufgezeigt, wie Large Language Models (LLMs) und generative KI in diesen Workflow integriert werden können – mit Fokus auf API-Serving, Governance und Observability.

Im Verlauf der vier Tage bauen die Teilnehmenden eigene End-to-End-Pipelines, deployen Modelle mit FastAPI oder Triton Inference Server, verfolgen Experimente mit MLflow und setzen Feature Stores sowie Vektordatenbanken ein. Zudem werden Strategien zur Erkennung von Concept Drift, Modellalterung und Prompt-Degradation vorgestellt.

Abgeschlossen wird die Schulung mit einem Teamprojekt, in dem eine vollständige MLOps-Pipeline entwickelt, bereitgestellt und über ein Dashboard monitorbar gemacht wird.

Wer teilnehmen sollte
Diese Schulung richtet sich an ML- und DevOps-Professionals, die produktionsreife AI-Systeme betreiben oder vorbereiten möchten. Voraussetzung sind solide Kenntnisse in Python sowie praktische Erfahrung mit ML-Workflows und APIs. Erfahrung mit Containerisierung (Docker) oder Cloud-Diensten ist von Vorteil, aber nicht zwingend notwendig.

Tag 1: Foundations of MLOps – Architektur, CI/CD & Reproduzierbarkeit

1. **MLOps-Grundlagen und Vergleich zu DevOps**
   - Unterschiede zwischen Software- und ML-Development-Lifecycle
   - Herausforderungen: Datenabhängigkeit, Drift, Evaluation
2. **Modellversionierung & Reproduzierbarkeit**
   - Git + DVC (Data Version Control)
   - Modell- und Datensatzversionierung in produktiven Pipelines
3. **Experiment-Tracking**
   - MLflow, Weights & Biases: Logging von Metriken, Parametern, Artefakten
   - Aufbau eines Model Registry Workflows
4. **Hands-on**
   - MLflow-Projekt mit Logging, Model Registry und Übergabe an Deployment-Stufe
   - Versioniertes Modelltraining mit DVC

Tag 2: Infrastructure & API Deployment für generative KI

1. **Deployment-Strategien für ML-Modelle**
   - Batch vs. Realtime Inference
   - FastAPI, Flask, Triton Inference Server
2. **Containerisierung & Orchestrierung**
   - Einführung in Docker
   - Einstieg in Kubernetes (lokal über Minikube oder remote)
3. **LLM-Serving & Prompt-Schnittstellen**
   - Prompt-Orchestrierung via API
   - Async Deployments für LLMs (OpenAI, Claude, Llama.cpp, etc.)
4. **Hands-on**
   - Erstelle eine Container-basierte API für ein Modell (z. B. HuggingFace)
   - Deployment mit Docker & FastAPI, optional Triton für LLM-Optimierung

Tag 3: Skalierung, Monitoring & Sicherheit

1. **Skalierbarkeit & Ausfallsicherheit**
   - Load Balancing, Model Caching, A/B-Testing
   - Auto-Scaling mit Kubernetes
2. **Monitoring & Observability**
   - Prometheus, Grafana, MLflow UI
   - Custom Dashboards für Modellmetriken & Systemgesundheit
3. **Concept Drift & Modellverfall**
   - Was ist Prompt Drift?
   - Integration von Drift Detection in Pipelines
4. **Security & Compliance**
   - Prompt Injection Detection
   - Role-Based Access Control (RBAC), Secrets Management
5. **Hands-on**
   - Baue ein Observability Dashboard
   - Simuliere Modell- und Daten-Drift

Tag 4: Abschlussprojekt – Von ML zu MLOps

1. **Feature Store & Datenmanagement**
    - Feast, Tecton: Zentrale Speicherung und Wiederverwendung von Features
    - Datenpipeline-Integration (Airflow, Prefect)
2. **CI/CD für KI**
    - GitHub Actions für automatisierte Modellbereitstellung
    - Testen von Modellen (Unit Tests, Performance Benchmarks)
3. **Abschlussprojekt**
    - Entwicklung einer End-to-End-MLOps-Pipeline:
        - DVC + MLflow + FastAPI + Prometheus + Deployment
        - Optional: LLM-Integration über LangChain oder Ollama
        - Aufbau eines Governance-Dashboards (Audit Trail, Modellmetrik)
4. **Ausblick & Best Practices**
    - Monitoring von LLM-Prompts
    - On-Prem vs. Cloud (SaaS, PaaS, Hybrid-Modelle)
    - Zukunft: AI-Agents, AutoMLOps, LLMOps