Curriculum für

Certified Professional for Software Architecture (CPSA)®
Advanced Level

Modul
SWARC4AI

Softwarearchitektur für KI-Systeme

2024.1-RC1-DE-20240912

Inhaltsverzeichnis

Die Abkürzung "e. V." ist Teil des offiziellen Namens des iSAQB und steht für "eingetragener Verein", der seinen Status als juristische Person nach deutschem Recht beschreibt. Der Einfachheit halber wird iSAQB e. V. im Folgenden ohne die Verwendung dieser Abkürzung als iSAQB bezeichnet.

Verzeichnis der Lernziele

- LZ 2-8: Kennen die Grundaussagen des europäischen AI-Acts (insbesondere Transparenzpflichten) und kennen Strategien für deren Einhaltung sowie mögliche Herausforderungen dabei.
- LZ 2-9: Wissen, wie man Modelle und Datensätze effektiv dokumentiert, um Nachvollziehbarkeit und Transparenz zu gewährleisten.
- LZ 2-10: Kennen mögliche Fallstricke hinsichtlich Security.
- LZ 2-11: Kennen typische Angriffsarten auf ML-Modelle und Beispiele dafür, u.a.: LLM-Jailbreaks durch Prompt-Engineering, Adversarial Attacks, Data Poisoning, Model Inversion & Extraction.
- LZ 2-12: Wissen, wie man AI-Risk Minimierung Strategien entwickelt und anwendet.
- LZ 2-13: Kennen verschiedene Möglichkeiten zur Absicherung gegen solche Angriffe (AI Security) und zur Integration von Sicherheitsstandards in die Architektur.
- LZ 2-14: Kennen die Grundproblematik und die verschiedenen Facetten von AI-Safety.
- LZ 2-15: Wissen um die Probleme hinsichtlich Ethik, die KI-Systeme mit sich bringen können
- LZ 2-16: Kennen Ansätze und Möglichkeiten, mit ethischen Problemen umzugehen z.B. KI-Alignment (und dessen Grenzen) oder die Erstellung eigener KI-Richtlinien.
- LZ 2-17: Kennen wichtige Ethik-Leitlinien wie die „EU-Ethik-Leitlinien für vertrauenswürdige KI“ sowie die „Google AI Ethics Guidelines“
- LZ 2-18: Kennen die wichtigsten Dokumente zu AI Governance, um die Kernprinzipien zu AI Governance und Responsible AI für das Unternehmen auszuarbeiten.
- LZ 2-19: Erhalten einen Einblick in die Einrichtung von "Regulatory Sandboxes" zur Förderung von Innovationen und die möglichen rechtlichen Konsequenzen bei Nichteinhaltung der Vorschriften des AI-Acts.
- LZ 2-20: Verstehen die Strukturen und Prozesse, die zur Steuerung und Kontrolle von KI-Systemen notwendig sind, um ethische und gesetzliche Anforderungen zu erfüllen.
- LZ 2-21: Wissen, wie effektive Datenverwaltung die Qualität und Sicherheit von Daten in KI-Anwendungen sicherstellt.
- LZ 2-22: Verstehen die Bedeutung der Transparenzpflicht bei KI-Systemen und wissen, wie sie diese in der Praxis umsetzen können.
- LZ 3-1: Haben ein Verständnis für den Life-Cycle eines Machine-Learning- bzw. Data-Science-Projekts: Exploratory Data Analysis, Data Cleansing und Aufbereitung, Feature Engineering, Modell Training und Auswahl, POC, Deployment, Maintenance.
- LZ 3-2: Kennen typische Vorgehensmodelle für software development von KI-Systemen:
- LZ 3-3: Kennen verschiedene Arten von Daten und typische ML-Probleme bzw. -Use-Cases.
- LZ 3-4: Haben ein Verständnis für verschiedene Anforderungen an die Daten, z.B. Vorhandensein von Labels verschiedener Art.
- LZ 3-5: Verstehen die verschiedenen Machine Learning Problemstellungen (Supervised Learning, Unsupervised Learning, Reinforcement Learning) und wissen, welche Anforderungen diese haben.
- LZ 3-6: Verständnis für die Notwendigkeit von Validierung und Kenntnis der typischen Datenaufteilung in Trainings-, Validierungs- und Testdaten.
- LZ 3-7: Verständnis für Input Daten für verschiedene KI-Algorithmen (z.B. Neuronale Netze) als numerische Vektoren und Matrizen bzw. Tensoren; One-Hot-Encodings; Embeddings.
- LZ 3-8: Verstehen, wie man mit Herausforderungen wie Nicht-Determinismus, Datenqualität und Concept- und Modell-Drift umgeht.

- LZ 3-9: Kennen Transfer-Learning bzw. Fine Tuning als Möglichkeit, um die vortrainierten Basismodelle auf bestehende Use Cases zu adoptieren.
- LZ 3-10: Wissen welche Design Patterns für KI-Systeme existieren und wie man passende Patterns auswählt:
- LZ 3-11: Wissen, wie man die ein Use Case / Aufgabe eines ML-Modells definiert, z.B. die Klassifikation von Bildern oder die Erkennung von Betrug.
- LZ 3-12:Verstehen, welche Eingaben und Ausgaben für das Funktionieren eines ML-Systems erforderlich sind und können diese spezifizieren.
- LZ 3-13: Kennen verschiedene Metriken zur Messung der Performance von ML-Modellen (z.B. Precision, Recall, F1, Accuracy, etc.) und wissen, wie man Bewertungskriterien zur Leistungsbeurteilung festlegt.
- LZ 3-14: Verstehen, wie ML-Modelle in bestehende Systeme integriert werden können und kennen die Schnittstellen und Integrationspunkte.
- LZ 3-15: Wissen, wie Benutzeroberflächen gestaltet werden sollten, um effektive Interaktionen mit dem ML-System zu ermöglichen und die Benutzererfahrung zu optimieren.
- LZ 3-16: Verstehen die Bedeutung von Leistungskennzahlen wie Latenz und Durchsatz in KI-Systemen und wissen, wie diese optimiert werden können.
- LZ 3-17: Verstehen die Bedeutung der Skalierbarkeit auf erhöhte Datenmengen und wissen, wie man KI-Systeme entwickelt, die mit steigenden Datenvolumen umgehen können, ohne an Leistung zu verlieren.
- LZ 3-18: Verstehen, was Robustheit in KI-Systemen bedeutet, und können Strategien zur Erhöhung der Robustheit in verschiedenen Anwendungskontexten anwenden.
- LZ 3-19: Verstehen die Konzepte der Zuverlässigkeit und Verfügbarkeit und wissen, wie sie KI-Systeme bauen, die stabil und konstant verfügbar sind.
- LZ 3-20: Verstehen, wie wichtig es ist, dass KI-Ergebnisse reproduzierbar und prüfbar sind, und wissen, welche Methoden zur Sicherstellung dieser Eigenschaften eingesetzt werden können.
- LZ 3-21: Kennen die Anforderungen an Sicherheit, Datenschutz und Compliance und wissen, wie diese in KI-Systemen umgesetzt werden.
- LZ 3-22: Wissen, wie man KI-Modelle und -Systeme entwickelt, die ressourcenschonend arbeiten, indem sie Speicher, Rechenleistung und Speicherplatz effizient nutzen.
- LZ 3-23: Verstehen die Bedeutung von Erklärbarkeit und Interpretierbarkeit in KI-Systemen und wissen, wie man diese sicherstellen kann, um Vertrauen und Transparenz zu fördern.
- LZ 3-24: Wissen, wie Bias in Daten und Modellen erkannt und reduziert werden können, um Fairness und Gleichbehandlung in KI-Anwendungen sicherzustellen.
- LZ 3-25: Kennen die Konzepte der Fehlertoleranz und können erläutern, wie KI-Systeme trotz Fehlern oder Störungen funktionsfähig bleiben.
- LZ 4-1: Kennen verschiedene Methoden, um Daten zu akquirieren und Daten zu labeln.
- LZ 4-2: Kennen gängige Plattformen für öffentlich zugängliche Daten.
- LZ 4-3: Haben einen Überblick über relevante Werkzeuge fürs Daten-Labeln, z.B. CVAT, Amazon Mechanical Turk.
- LZ 4-4: Haben Verständnis für die Gestaltung effizienter Datenpipelines und -architekturen, Betrachtung von Datenqualität, Speicherlösungen und deren Management.

- LZ 4-5: Kennen Architekturmuster für Data-Engineering-Pipelines und ETL-Prozesse.
- LZ 4-6: Kennen Strategien für Datenaggregation, -bereinigung, -transformation, -anreicherung und -augmentierung.
- LZ 4-7: Kennen relevante Werkzeuge für Data Engineering Pipelines wie Apache Spark und Flink.
- LZ 4-8: Kennen verschiedene Möglichkeiten zur Speicherung der Daten sowie deren Vor- und Nachteile: (CSV-)Dateien, Spaltenorientierte Dateien, Relationale und NoSQL-Datenbanken, Data Warehouses, Data Lakes.
- LZ 5-1: Kennen die unterschiedliche (Hardware)-Anforderungen (TPU, GPU, CPU) an Training und Inferenz.
- LZ 5-2: Kennen beispielhaft Trade-Offs verschiedener Modellarchitekturen bezüglich der Qualitätsmerkmale (insbesondere für Skalierung, Effizienz und Speicherlast), z.B. die Vor- und Nachteile von RNNs und Transformern.
- LZ 5-3: Kennen Möglichkeiten, um verschiedene Qualitätsmerkmale wie Genauigkeit, Effizienz und Speicherlast eines ML-Modells abzustimmen und gegeneinander einzutauschen, z.B. Quantisierung, Pruning, Destillierung, LoRA.
- LZ 5-4: Haben ein Verständnis für Kosten, Stromverbrauch und nachhaltige Nutzung von KI (Green IT).
- LZ 5-5: Kennen den Begriff MLOps für die Automatisierung des Life-Cycles eines Data-Science-Projekts und den Zusammenhang mit DevOps
- LZ 5-6: Haben Verständnis für das Tracking von Modelltraining, Parametern, Metriken und Ergebnissen.
- LZ 5-7: Kennen Ansätze zur Evaluation von ML-Modellen und darauf aufbauenden KI-Systemen.
- LZ 5-8: Kennen verschiedene Arten von Drift, z.B. Daten-Drift und Modell-Drift, sowie mögliche Ursachen und Lösungsansätze dafür.
- LZ 5-9: Haben Verständnis für CI/CD-Pipelines, Modellmanagement und Deployment-Strategien für KI-Modelle.
- LZ 5-10: Kennen verschiedene Möglichkeiten der Zusammenarbeit und Verantwortungsaufteilung zwischen den verschiedenen Rollen, z.B. Data Engineer, ML-Engineer und Softwareentwickler:In bezogen auf die verschiedenen Phasen des Life-Cycles.
- LZ 5-11: Kennen gängige Plattformen für die Modellbereitstellung, z.B. Huggingface Hub.
- LZ 5-12: Kennen gängige Werkzeuge für das Erstellen von POCs von KI-Systemen, z.B. Gradio.
- LZ 5-13: Kennen verschiedene Deployment-Möglichkeiten: API Deployment, Embedded Deployment, Batch Prediction, Streaming, Containerization, Serverless Deployment, Cloud Services.
- LZ 5-14: Kennen die Vor- und Nachteile von SaaS und Self-Hosting und können dazwischen abwägen.
- LZ 5-15: Kennen bekannte SaaS-KI-Lösungen, z.B. Azure OpenAI Services.
- LZ 5-16: Kennen verschiedene Möglichkeiten und Standards für Embedded Deployments von ML-Modellen.
- LZ 5-17: Verstehen die Notwendigkeit für Monitoring, auch in Hinblick auf KI-spezifische Anforderungen wie das Tracking von Drift.
- LZ 5-18: Kennen relevante Metriken wie Accuracy, Precision, Recall, F1-Score, MAE, MSE, Perplexity, Latenz, Durchsatz und Ressourcenauslastung.
- LZ 5-19: Kennen Beispiel-Werkzeuge für Monitoring, sowohl allgemeine (z.B. Prometheus & Grafana) als auch ML-spezifische (z.B. MLflow).

- LZ 5-20: Verstehen den Nutzen von Nutzer-Feedback für das weitere Modelltraining.
- LZ 5-21: Kennen verschiedene Methoden und Werkzeuge zur Sammlung von Nutzer-Feedback, z.B. Auswahl zwischen mehreren Antworten und Flagging in Gradio.
- LZ 5-22: Kennen verschiedene Methoden zur Nutzung von Feedback für das Modell-Training, z.B. RLHF, RLAIF und DPO.
- LZ 5-23: Erfahren anhand eines Praxisbeispiels, wie eine MLOps-Pipeline aussehen kann und welche Einsichten diese auf die Parameter, Metriken usw. bietet.
- LZ 5-24: Können Build vs. Buy Entscheidungen für MLOps Systeme/Komponente treffen.
- LZ 5-25: Kennen bekannte MLOps-Werkzeuge und End-to-End Plattformen, bspw.:
- LZ 6-1: Kennen verschiedene Integrationsebenen von KI: Anwendungen (z.B. Coding Assistenten), AI-Engineering (z.B. Prompt Engineering), ML-Modellentwicklung (z.B. pytorch), ML-Infrastruktur (z.B. Vektor-DBs).
- LZ 6-2: Können das strategische Design von DDD (insbesondere Context Maps) einsetzen, um die Art und den Grad der Integration von KI-Systemen zu bestimmen und zu dokumentieren.
- LZ 6-3: Kennen die Qualitätsmerkmale, die für KI-Systeme besonders relevant sind: Verlässlichkeit, Skalierbarkeit, Effizienz, Sicherheit, Wartbarkeit, Interpretierbarkeit etc.
- LZ 6-4: Kennen gängige Evaluations-Frameworks, um mit Unbestimmtheit und Fehlern in KI-Systemen umzugehen z.B. LangSmith oder LangFuse.
- LZ 6-5: Kennen einige Beispiele von verbreiteten Bibliotheken, Schnittstellen und Tools zur Integration von KI-Modellen.
- LZ 6-6: Üben und diskutieren anhand eines Fallbeispiels mit einer ausgedachten Fachlichkeit, Integrationsoptionen für KI in eine bestehende Software-Landschaft abzuwägen.
- LZ 6-7: Haben ein grundlegendes Verständnis von generativer KI z.B. LLMs und Stable Diffusion.
- LZ 6-8: Haben einen Überblick über die Funktionsweise von LLMs und die zugehörige Begriffswelt: Token, Embedding, RNN, Transformer, Attention…
- LZ 6-9: Kennen bekannte Patterns bei der Nutzung von LLMs:
- LZ 6-10: Kennen typische Use-Cases für RAG wie „Talk to your documents/database/API“.
- LZ 6-11: Kennen verschiedene RAG-Techniken:
- LZ 6-12: Kennen verschiedene Arten von Prompt Engineering (z.B. Few-Shot-Learning, Chain-of-Thought, Role-Playing) und allgemeine Best Practices für das Prompting.
- LZ 6-13: Wissen, was Agentic Workflows sind: Reflexion, Werkzeug-Nutzung, Planung, Multi-Agenten-Kollaboration.
- LZ 6-14: Wissen welche Design Patterns für Generative AI-Systeme existieren:
- LZ 6-15: Kennen Techniken zur Evaluation von LLM-Anwendungen: Scoring, Human Feedback, Comparative Evaluation, Model Based Evaluation etc.
- LZ 6-16: Kennen bekannte LLMs und Auswahlkriterien: GPT, Claude, Gemini, Llama, Mistral, Luminous etc.
- LZ 6-17: Verstehen die Bedeutung von Cost Management für GenAI Applikationen.
- LZ 6-18: Kennen einige Beispiele von verbreiteten Bibliotheken, Schnittstellen und Tools im Zusammenhang mit LLM-Anwendungen: OpenAI-API, LangChain etc.

- LZ 6-19: Kennen Agentic AI Software Architekturen, AI Agent Architekturkomponenten, Typen vom AI Agentarchitekturen.
- LZ 7-1: Üben anhand von Fallstudien und Praxisprojekten, das erworbene Wissen in realen Szenarien anzuwenden.

Einführung: Allgemeines zum iSAQB Advanced Level

Was vermittelt ein Advanced Level Modul?

Das Modul kann unabhängig von einer CPSA-F-Zertifizierung besucht werden.

- Der iSAQB Advanced Level bietet eine modulare Ausbildung in drei Kompetenzbereichen mit flexibel gestaltbaren Ausbildungswegen. Er berücksichtigt individuelle Neigungen und Schwerpunkte.
- Die Zertifizierung erfolgt als Hausarbeit. Die Bewertung und mündliche Prüfung wird durch vom iSAQB benannte Expert:innen vorgenommen.

Was können Absolventen des Advanced Level (CPSA-A)?

CPSA-A-Absolventen können:

- eigenständig und methodisch fundiert mittlere bis große IT-Systeme entwerfen
- in IT-Systemen mittlerer bis hoher Kritikalität technische und inhaltliche Verantwortung übernehmen
- Maßnahmen zur Erreichung von Qualitätsanforderungen konzeptionieren, entwerfen und dokumentieren sowie Entwicklungsteams bei der Umsetzung dieser Maßnahmen begleiten
- architekturrelevante Kommunikation in mittleren bis großen Entwicklungsteams steuern und durchführen

Voraussetzungen zur CPSA-A-Zertifizierung

- erfolgreiche Ausbildung und Zertifizierung zum Certified Professional for Software Architecture, Foundation Level® (CPSA-F)
- mindestens drei Jahre Vollzeit-Berufserfahrung in der IT-Branche; dabei Mitarbeit an Entwurf und Entwicklung von mindestens zwei unterschiedlichen IT-Systemen
    - Ausnahmen sind auf Antrag zulässig (etwa: Mitarbeit in Open-Source-Projekten)
- Aus- und Weiterbildung im Rahmen von iSAQB-Advanced-Level-Schulungen im Umfang von mindestens 70 Credit Points aus mindestens drei unterschiedlichen Kompetenzbereichen
- erfolgreiche Bearbeitung der CPSA-A-Zertifizierungsprüfung

Grundlegendes

Was vermittelt das Modul „SWARC4AI“?

Das Modul präsentiert den Teilnehmer:innen moderne Softwarearchitektur-Konzepte für KI-Systeme als Mittel, um leistungsfähige, skalierbare und integrierbare KI- Lösungen zu gestalten. Am Ende des Moduls kennen die Teilnehmer:innen die wesentlichen Prinzipien der Softwarearchitektur für KI-Systeme und können diese bei Entwurf und Implementierung von Machine Learning und Generative KI- Systemen anwenden. Sie sind mithilfe der vermittelten kommunikativen Fähigkeiten in der Lage, eine einheitliche Sprache zwischen Data Scientists, KI-Expert:innen und Softwareentwickler:innen zu etablieren. Mit Hilfe der vermittelten Modellierungstechniken und Architekturwerkzeuge können sie KI-Komponenten nahtlos in bestehende Softwaresysteme integrieren. Die Schulung umfasst sowohl Machine Learning Systeme als auch Generative KI und vermittelt, wie diese mit klassischen Softwaresystemen kombiniert werden können. Die Teilnehmer:innen lernen, wie die Architektur für solche hybriden Systeme aussehen muss, um Skalierbarkeit, Wartbarkeit und Erweiterbarkeit zu gewährleisten.

Struktur des Lehrplans und empfohlene zeitliche Aufteilung

Inhalt	Empfohlene Mindestdauer (min)
Einführung in softwarearchitekturrelevante Konzepte für Künstliche Intelligenz	120
Compliance, Security, Alignment	120
Architektur von KI-Systemen	320
Datenmanagement und Datenverarbeitung für KI-Systeme	90
Skalierbarkeit und Leistungsoptimierung von KI-Systemen	160
Systemarchitekturen- und Plattformen für Generative KI-Systeme	160
Fallstudien und Praxisprojekte	110
Gesamt	1080

Dauer, Didaktik und weitere Details

Die Dauer einer Schulung zum Modul SWARC4AI sollte mindestens 3 Tage betragen, kann aber länger sein. Anbieter können sich durch Dauer, Didaktik, Art und Aufbau der Übungen sowie der detaillierten Kursgliederung voneinander unterscheiden. Insbesondere die Art der Beispiele und Übungen lässt der Lehrplan komplett offen. Lizenzierte Schulungen zu SWARC4AI tragen zur Zulassung zur abschließenden Advanced-Level-Zertifizierungsprüfung folgende Credit Points) bei:

Methodische Kompetenz:	10 Punkte
Technische Kompetenz:	20 Punkte
Kommunikative Kompetenz:	0 Punkte

Voraussetzungen

Teilnehmer:innen **sollten** folgende Kenntnisse und/oder Erfahrung mitbringen:

- Grundlegendes Wissen zu KI, Machine Learning und Data Science
- Machine Learning Methoden wie Supervised, Unsupervised Learning und Reinforcement Learning, z.B Klassifikation, Clustering, Regression, Dimensionality Reduction
- Grundsätzliches Wissen über die typischen Algorithmen wie: Linear/logistic regression, decision trees, random forests, support vector machines, K-means clustering, neural networks and deep learning
- Wissen über Model evaluation: Cross-validation, Metrics (accuracy, precision, recall, F1-score), Bias-variance tradeoff
- Erfahrung mit typischen Frameworks wie sk-learn, TensorFlow, PyTorch
- Erfahrung mit Jupyter Notebooks oder ähnlichen Rapid Application Development (RAD) Werkzeugen
- Wissen über die technischen Verfahren hinter KI, einschließlich: Maschinelles Lernen inkl. Deep Learning und LLMs, NLP-Ansätze (Natural Language Processing), Transformationsmodelle
- Praktische Erfahrung mit Trainieren von Künstlichen Intelligenz Modellen

Gliederung des Lehrplans

Die einzelnen Abschnitte des Lehrplans sind gemäß folgender Gliederung beschrieben:

- **Begriffe/Konzepte**: Wesentliche Kernbegriffe dieses Themas.
- **Unterrichts-/Übungszeit**: Legt die Unterrichts- und Übungszeit fest, die für dieses Thema bzw. dessen Übung in einer akkreditierten Schulung mindestens aufgewendet werden muss.
- **Lernziele**: Beschreibt die zu vermittelnden Inhalte inklusive ihrer Kernbegriffe und -konzepte.

Dieser Abschnitt skizziert damit auch die zu erwerbenden Kenntnisse in entsprechenden Schulungen.

Ergänzende Informationen, Begriffe, Übersetzungen

Soweit für das Verständnis des Lehrplans erforderlich, haben wir Fachbegriffe ins iSAQB-Glossar aufgenommen, definiert und bei Bedarf durch die Übersetzungen der Originalliteratur ergänzt.

1. Einführung in softwarearchitekturrelevante Konzepte für Künstliche Intelligenz

Dauer: 120 Min.	Übungszeit: 0 Min.

1.1. Begriffe und Konzepte

KI, Generative KI, Machine Learning, Symbolische KI, Evolutionäre Algorithmen, Statistical Learning, Deep Learning, LLMs, Halluzination, Bias, Jagged Technological Frontier, Data Preparation, Model Training, Model Evaluation, Feature Engineering, Data Science, Data Engineering

1.2. Lernziele

Die Teilnehmer:Innen …

LZ 1-1: Wissen wie man Künstliche Intelligenz definiert und die Anordnung zu Machine Learning, Data Science, Deep Learning, Generative AI.

LZ 1-2: Wissen, wie man typische allgemeine Use-Cases für KI spezifiziert, z.B. Bilderkennung & -erzeugung, Sprachverarbeitung, Vorhersagen, Personalisierung und Anomalieerkennung.

LZ 1-3: Kennen Einsatzmöglichkeiten in verschiedenen Branchen, z.B. Marketing, Medizin, Robotik und Content-Creation.

LZ 1-4: Kennen Einsatzmöglichkeiten in Endnutzeranwendungen, z.B. bSprachassistenten (Chatbots) und Empfehlungssysteme (Recommender Engine).

LZ 1-5: Wissen, wie man Risiken bei der Anwendung von KI (Halluzinationen, Bias, Fairness…) und gesellschaftlichen Risiken (Deepfakes, AI-enabled Cyberattacs, Safety Risks in Critical Systems, Social Manipulation, Intellectual Property Issues, etc.) identifiziert.

LZ 1-6: Kennen den Unterschied zu traditioneller Software:

- Datengetrieben (bei ML) - Daten-zentrierte statt Code-zentrierte Entwicklung
- Probabilistische Ergebnisse (Non-deterministic behavior)
- Statistische Validierung
- Experimentelles Design: Unterstützung für schnelle Iteration und Testen von Modellen.
- Modellkomplexität und Interpretierbarkeit: ML-Modelle, insbesondere Deep-Learning-Modelle, können äußerst komplex sein. Der „Black-Box“-Charakter erschwert die Interpretierbarkeit und Erklärbarkeit.
- Debugging und Tests sind aufgrund des Nichtdeterminismus komplexer.
- Model Decay: Continuous monitoring und Retraining sind notwendig, um die Leistung der Modelle aufrecht zu erhalten.
- AI-spezifische Regulatorik von Branchen und auf EU-Ebene müssen berücksichtigt werden.
- Interoperabilität: Nahtlose Integration in bestehende Systeme und Technologiestacks.

LZ 1-7: Können beurteilen, ob ein Problem mittels KI oder klassischer SW-Entwicklung zu lösen ist.

LZ 1-8: Kennen typische Rollen und deren Aufgaben in diesen Kontexten: Data Scientist, Data Analyst, Data Engineer, Machine Learning Engineer, MLOps Engineer, AI Architect, Data Architect, Business Intelligence (BI) Developer, Data Governance Specialist, ML-Researcher.

LZ 1-9: Wissen, wie diese Rollen im Team zusammenarbeiten könnten (Team Topologies für ML-Teams).

LZ 1-10: Wissen, wie man KI Use Cases identifiziert und priorisiert.

LZ 1-11: Kennen KI Stärken und Grenzen (Jagged Technological Frontier).

LZ 1-12: Wissen wie man Künstliche Intelligenz definiert und die Anordnung zu Machine Learning, Data Science, Deep Learning, Generative AI.

LZ 1-13: Kennen Productivity J-Curve Konzept in Verbindung mit KI-Technologie (Dieses Phänomen hilft zu verstehen, warum Unternehmen bei der Implementierung von KI zunächst einen Produktivitätsrückgang verzeichnen können, dem bei der weiteren Entwicklung aber Produktivitätsgewinne folgen können.).

1.3. Referenzen

[Roser 2022], [Burkov 2019], [Géron 2022], [Kelleher 2015], [Vaughan 2020], [Bahree 2024], [Harvard et al. 2024], [Dell'Acqua 2022], [Visengeriyeva], [Visengeriyeva], [Agrawal et al.], [Tan et al.], [Chong et al.], [Hall et al. 2023], [Huyen 2022], [Wang et al. 2024]

2. Compliance, Security, Alignment

Dauer: 120 Min.	Übungszeit: 30 Min.

2.1. Begriffe und Konzepte

EU AI Act, Datenschutz, Urheberrecht, Lizenz, Open (Source), AI Security, Jailbreak,Adversarial Attack, Data Poisoning, Model Inversion & Extraction, AI Safety, AI Ethics, AI Alignment, Model und Data Dokumentation, Transparenzpflicht, Human Oversight, Data Governance, AI Governance, AI Systems by Risk Levels (Prohibited, High-Risk, Limited Risk, Low-Risk)

2.2. Lernziele

Die Teilnehmer:Innen …

LZ 2-1: Wissen, wie Datenschutzgesetze, wie die DSGVO, die Sammlung, Verarbeitung und Speicherung von Daten durch KI-Systeme beeinflussen.

LZ 2-2: Verstehen die Ziele und Regelungen des EU AI Act und dessen Einfluss auf die Entwicklung und den Einsatz von KI-Systemen.

LZ 2-3: Verstehen die Anforderungen für EU AI Act (Trustworthy AI) und welchen Einfluss diese Anforderungen auf die Architektur des Softwaresystems hat:

- Risikomanagementsystem (Risikominimierung)
- Datenqualität und Datengovernance (Qualitätsmanagementsystem)
- Erstellung und Pflege einer umfassenden technischen Dokumentation des KI-Systems
- Automatische Aufzeichnung/Logging von Events im KI-System.
- Bereitstellung klarer und verständlicher Informationen für Nutzer
- Implementierung der Maßnahmen zur menschlichen Aufsicht
- Gewährleistung eines angemessenen Maßes an Genauigkeit/Accuracy und Robustheit
- Implementierung von Maßnahmen zur Cybersicherheit

LZ 2-4: Wissen, wie KI-Systeme nach EU AI Act Risikolevel (verboten, hochrisikoreich, begrenzt risikoreich, niedrigrisikoreich) klassifiziert werden und welche regulatorischen Anforderungen jeweils gelten.

LZ 2-5: Wissen um die Urheberrechtsproblematik für KI-generierte Inhalte und die Auswirkungen auf bestimmte Software-Lizenzmodelle sowie mögliche Umgänge damit.

LZ 2-6: Kennen verschiedene Arten bzw. Grade der Offenheit freier ML-Modelle, z.B. was die Offenlegung der Daten und der Modellparameter anbelangt.

LZ 2-7: Kennen verschiedene Arten von Lizenzen freier ML-Modelle sowie deren Auswirkungen.

LZ 2-8: Kennen die Grundaussagen des europäischen AI-Acts (insbesondere Transparenzpflichten) und kennen Strategien für deren Einhaltung sowie mögliche Herausforderungen dabei.

LZ 2-9: Wissen, wie man Modelle und Datensätze effektiv dokumentiert, um Nachvollziehbarkeit und Transparenz zu gewährleisten.

LZ 2-10: Kennen mögliche Fallstricke hinsichtlich Security.

LZ 2-11: Kennen typische Angriffsarten auf ML-Modelle und Beispiele dafür, u.a.: LLM-Jailbreaks durch Prompt-Engineering, Adversarial Attacks, Data Poisoning, Model Inversion & Extraction.

LZ 2-12: Wissen, wie man AI-Risk Minimierung Strategien entwickelt und anwendet.

LZ 2-13: Kennen verschiedene Möglichkeiten zur Absicherung gegen solche Angriffe (AI Security) und zur Integration von Sicherheitsstandards in die Architektur.

LZ 2-14: Kennen die Grundproblematik und die verschiedenen Facetten von AI-Safety.

LZ 2-15: Wissen um die Probleme hinsichtlich Ethik, die KI-Systeme mit sich bringen können

LZ 2-16: Kennen Ansätze und Möglichkeiten, mit ethischen Problemen umzugehen z.B. KI-Alignment (und dessen Grenzen) oder die Erstellung eigener KI-Richtlinien.

LZ 2-17: Kennen wichtige Ethik-Leitlinien wie die „EU-Ethik-Leitlinien für vertrauenswürdige KI“ sowie die „Google AI Ethics Guidelines“

LZ 2-18: Kennen die wichtigsten Dokumente zu AI Governance, um die Kernprinzipien zu AI Governance und Responsible AI für das Unternehmen auszuarbeiten.

- OECD AI Principles, https://oecd.ai/en/ai-principles
- The Asilomar AI Principles, https://futureoflife.org/open-letter/ai-principles/
- The IEEE Ethically Aligned Design framework, https://standards.ieee.org/wp-content/uploads/import/documents/other/ead_v2.pdf

LZ 2-19: Erhalten einen Einblick in die Einrichtung von "Regulatory Sandboxes" zur Förderung von Innovationen und die möglichen rechtlichen Konsequenzen bei Nichteinhaltung der Vorschriften des AI-Acts.

LZ 2-20: Verstehen die Strukturen und Prozesse, die zur Steuerung und Kontrolle von KI-Systemen notwendig sind, um ethische und gesetzliche Anforderungen zu erfüllen.

LZ 2-21: Wissen, wie effektive Datenverwaltung die Qualität und Sicherheit von Daten in KI-Anwendungen sicherstellt.

LZ 2-22: Verstehen die Bedeutung der Transparenzpflicht bei KI-Systemen und wissen, wie sie diese in der Praxis umsetzen können.

2.3. Referenzen

[Engler et al.], [Nist], [Hall et al. 2023], [Masood et al. 2023], [CSIRO et al. 2023], [Pruksachatkun et al. 2023], [Chen et al. 2022], [ATLAS], [Visengeriyeva], [EU AI Act], [Hotz], [Bhajaria 2022], [Jarmul 2023], [Molnar 2024]

3. Architektur von KI-Systemen

Dauer: 320 Min.	Übungszeit: 45 Min.

3.1. Begriffe und Konzepte

CI/CD, MLOps, CRISP-ML(Q), AI-Application, AI-Engineering, ML-Modellentwicklung, ML-Infrastruktur, Performance, Robustheit, Zuverlässigkeit, Fehlertoleranz, Model-as-Service, Model-as-Dependency, Precompute, Model-on-Demand, Hybrid-Serving, Multi-Agent System (MAS)

3.2. Lernziele

Die Teilnehmer:Innen …

LZ 3-1: Haben ein Verständnis für den Life-Cycle eines Machine-Learning- bzw. Data-Science-Projekts: Exploratory Data Analysis, Data Cleansing und Aufbereitung, Feature Engineering, Modell Training und Auswahl, POC, Deployment, Maintenance.

LZ 3-2: Kennen typische Vorgehensmodelle für software development von KI-Systemen:

- CRISP-ML(Q)
- Team Data Science Process
- GenAI Life Cycle

LZ 3-3: Kennen verschiedene Arten von Daten und typische ML-Probleme bzw. -Use-Cases.

LZ 3-4: Haben ein Verständnis für verschiedene Anforderungen an die Daten, z.B. Vorhandensein von Labels verschiedener Art.

LZ 3-5: Verstehen die verschiedenen Machine Learning Problemstellungen (Supervised Learning, Unsupervised Learning, Reinforcement Learning) und wissen, welche Anforderungen diese haben.

LZ 3-6: Verständnis für die Notwendigkeit von Validierung und Kenntnis der typischen Datenaufteilung in Trainings-, Validierungs- und Testdaten.

LZ 3-7: Verständnis für Input Daten für verschiedene KI-Algorithmen (z.B. Neuronale Netze) als numerische Vektoren und Matrizen bzw. Tensoren; One-Hot-Encodings; Embeddings.

LZ 3-8: Verstehen, wie man mit Herausforderungen wie Nicht-Determinismus, Datenqualität und Concept- und Modell-Drift umgeht.

LZ 3-9: Kennen Transfer-Learning bzw. Fine Tuning als Möglichkeit, um die vortrainierten Basismodelle auf bestehende Use Cases zu adoptieren.

LZ 3-10: Wissen welche Design Patterns für KI-Systeme existieren und wie man passende Patterns auswählt:

- ML Systems Topology Patterns
- Pipeline Architecture Patterns
- Model Training Pattern
- Model Serving Patterns
- Model Deployment Patterns

Funktionale Anforderungen

LZ 3-11: Wissen, wie man die ein Use Case / Aufgabe eines ML-Modells definiert, z.B. die Klassifikation von Bildern oder die Erkennung von Betrug.

LZ 3-12:Verstehen, welche Eingaben und Ausgaben für das Funktionieren eines ML-Systems erforderlich sind und können diese spezifizieren.

LZ 3-13: Kennen verschiedene Metriken zur Messung der Performance von ML-Modellen (z.B. Precision, Recall, F1, Accuracy, etc.) und wissen, wie man Bewertungskriterien zur Leistungsbeurteilung festlegt.

LZ 3-14: Verstehen, wie ML-Modelle in bestehende Systeme integriert werden können und kennen die Schnittstellen und Integrationspunkte.

LZ 3-15: Wissen, wie Benutzeroberflächen gestaltet werden sollten, um effektive Interaktionen mit dem ML-System zu ermöglichen und die Benutzererfahrung zu optimieren.

Nicht-funktionale Anforderungen

LZ 3-16: Verstehen die Bedeutung von Leistungskennzahlen wie Latenz und Durchsatz in KI-Systemen und wissen, wie diese optimiert werden können.

LZ 3-17: Verstehen die Bedeutung der Skalierbarkeit auf erhöhte Datenmengen und wissen, wie man KI-Systeme entwickelt, die mit steigenden Datenvolumen umgehen können, ohne an Leistung zu verlieren.

LZ 3-18: Verstehen, was Robustheit in KI-Systemen bedeutet, und können Strategien zur Erhöhung der Robustheit in verschiedenen Anwendungskontexten anwenden.

LZ 3-19: Verstehen die Konzepte der Zuverlässigkeit und Verfügbarkeit und wissen, wie sie KI-Systeme bauen, die stabil und konstant verfügbar sind.

LZ 3-20: Verstehen, wie wichtig es ist, dass KI-Ergebnisse reproduzierbar und prüfbar sind, und wissen, welche Methoden zur Sicherstellung dieser Eigenschaften eingesetzt werden können.

LZ 3-21: Kennen die Anforderungen an Sicherheit, Datenschutz und Compliance und wissen, wie diese in KI-Systemen umgesetzt werden.

LZ 3-22: Wissen, wie man KI-Modelle und -Systeme entwickelt, die ressourcenschonend arbeiten, indem sie Speicher, Rechenleistung und Speicherplatz effizient nutzen.

LZ 3-23: Verstehen die Bedeutung von Erklärbarkeit und Interpretierbarkeit in KI-Systemen und wissen, wie man diese sicherstellen kann, um Vertrauen und Transparenz zu fördern.

LZ 3-24: Wissen, wie Bias in Daten und Modellen erkannt und reduziert werden können, um Fairness und Gleichbehandlung in KI-Anwendungen sicherzustellen.

LZ 3-25: Kennen die Konzepte der Fehlertoleranz und können erläutern, wie KI-Systeme trotz Fehlern oder Störungen funktionsfähig bleiben.

3.3. Referenzen

[TU Berlin], [Bornstein et al.], [Crowe et al. 2024], [Lakshmanan et al.], [Alake], [Lakshmanan et al.], [Lakshmanan et al.], [Koc], [Cdteliot], [Visengeriyeva], [Zaharia et al.], [Savarese], [tdcox], [Studer et al.], [Hotz], [Hotz], [Saltz], [Serban], [Heiland et al. 2023], [Nahar et al.], [ML software architecture],

4. Datenmanagement und Datenverarbeitung für KI-Systeme

Dauer: 90 Min.	Übungszeit: 0 Min.

4.1. Begriffe und Konzepte

Datenakquise, Labelling, Daten-Pipeline, ETL-Prozesse, Datenaggregation, Datenbereinigung, Transformation, Augmentierung, Dateien, RDBMS, NoSQL, Data Products, Data Contracts, Data Architectures: Data Warehouse, Data Lake, Data Mesh

4.2. Lernziele

Die Teilnehmer:Innen …

LZ 4-1: Kennen verschiedene Methoden, um Daten zu akquirieren und Daten zu labeln.

LZ 4-2: Kennen gängige Plattformen für öffentlich zugängliche Daten.

LZ 4-3: Haben einen Überblick über relevante Werkzeuge fürs Daten-Labeln, z.B. CVAT, Amazon Mechanical Turk.

LZ 4-4: Haben Verständnis für die Gestaltung effizienter Datenpipelines und -architekturen, Betrachtung von Datenqualität, Speicherlösungen und deren Management.

LZ 4-5: Kennen Architekturmuster für Data-Engineering-Pipelines und ETL-Prozesse.

LZ 4-6: Kennen Strategien für Datenaggregation, -bereinigung, -transformation, -anreicherung und -augmentierung.

LZ 4-7: Kennen relevante Werkzeuge für Data Engineering Pipelines wie Apache Spark und Flink.

LZ 4-8: Kennen verschiedene Möglichkeiten zur Speicherung der Daten sowie deren Vor- und Nachteile: (CSV-)Dateien, Spaltenorientierte Dateien, Relationale und NoSQL-Datenbanken, Data Warehouses, Data Lakes.

4.3. Referenzen

[Sarkis], [Serra], [Dehghani], [Reis et al.], [Bornstein et al.], [Ford et al.], [Bhajaria 2022], [Sanderson et al.], [Jones]

5. Skalierbarkeit und Leistungsoptimierung von KI-Systemen

Dauer: 160 Min.	Übungszeit: 30 Min.

5.1. Begriffe und Konzepte

Monitoring, Logging, Feedback, FinOps für KI-Plattformen

5.2. Lernziele

LZ 5-1: Kennen die unterschiedliche (Hardware)-Anforderungen (TPU, GPU, CPU) an Training und Inferenz.

LZ 5-2: Kennen beispielhaft Trade-Offs verschiedener Modellarchitekturen bezüglich der Qualitätsmerkmale (insbesondere für Skalierung, Effizienz und Speicherlast), z.B. die Vor- und Nachteile von RNNs und Transformern.

LZ 5-3: Kennen Möglichkeiten, um verschiedene Qualitätsmerkmale wie Genauigkeit, Effizienz und Speicherlast eines ML-Modells abzustimmen und gegeneinander einzutauschen, z.B. Quantisierung, Pruning, Destillierung, LoRA.

LZ 5-4: Haben ein Verständnis für Kosten, Stromverbrauch und nachhaltige Nutzung von KI (Green IT).

LZ 5-5: Kennen den Begriff MLOps für die Automatisierung des Life-Cycles eines Data-Science-Projekts und den Zusammenhang mit DevOps

LZ 5-6: Haben Verständnis für das Tracking von Modelltraining, Parametern, Metriken und Ergebnissen.

LZ 5-7: Kennen Ansätze zur Evaluation von ML-Modellen und darauf aufbauenden KI-Systemen.

LZ 5-8: Kennen verschiedene Arten von Drift, z.B. Daten-Drift und Modell-Drift, sowie mögliche Ursachen und Lösungsansätze dafür.

LZ 5-9: Haben Verständnis für CI/CD-Pipelines, Modellmanagement und Deployment-Strategien für KI-Modelle.

LZ 5-10: Kennen verschiedene Möglichkeiten der Zusammenarbeit und Verantwortungsaufteilung zwischen den verschiedenen Rollen, z.B. Data Engineer, ML-Engineer und Softwareentwickler:In bezogen auf die verschiedenen Phasen des Life-Cycles.

LZ 5-11: Kennen gängige Plattformen für die Modellbereitstellung, z.B. Huggingface Hub.

LZ 5-12: Kennen gängige Werkzeuge für das Erstellen von POCs von KI-Systemen, z.B. Gradio.

LZ 5-13: Kennen verschiedene Deployment-Möglichkeiten: API Deployment, Embedded Deployment, Batch Prediction, Streaming, Containerization, Serverless Deployment, Cloud Services.

LZ 5-14: Kennen die Vor- und Nachteile von SaaS und Self-Hosting und können dazwischen abwägen.

LZ 5-15: Kennen bekannte SaaS-KI-Lösungen, z.B. Azure OpenAI Services.

LZ 5-16: Kennen verschiedene Möglichkeiten und Standards für Embedded Deployments von ML-Modellen.

LZ 5-17: Verstehen die Notwendigkeit für Monitoring, auch in Hinblick auf KI-spezifische Anforderungen wie das Tracking von Drift.

LZ 5-18: Kennen relevante Metriken wie Accuracy, Precision, Recall, F1-Score, MAE, MSE, Perplexity, Latenz, Durchsatz und Ressourcenauslastung.

LZ 5-19: Kennen Beispiel-Werkzeuge für Monitoring, sowohl allgemeine (z.B. Prometheus & Grafana) als auch ML-spezifische (z.B. MLflow).

LZ 5-20: Verstehen den Nutzen von Nutzer-Feedback für das weitere Modelltraining.

LZ 5-21: Kennen verschiedene Methoden und Werkzeuge zur Sammlung von Nutzer-Feedback, z.B. Auswahl zwischen mehreren Antworten und Flagging in Gradio.

LZ 5-22: Kennen verschiedene Methoden zur Nutzung von Feedback für das Modell-Training, z.B. RLHF, RLAIF und DPO.

LZ 5-23: Erfahren anhand eines Praxisbeispiels, wie eine MLOps-Pipeline aussehen kann und welche Einsichten diese auf die Parameter, Metriken usw. bietet.

LZ 5-24: Können Build vs. Buy Entscheidungen für MLOps Systeme/Komponente treffen.

LZ 5-25: Kennen bekannte MLOps-Werkzeuge und End-to-End Plattformen, bspw.:

- Domino Data Lab, h2o.ai, DVC, activeloop, aporia, argo, arize, bentoML, comet ML, DagsHub, Databricks MLOps Stacks, Feast, Kedro, Kubeflow, Metaflow, MLflow, MLRun, prefect, PrimeHub, Weights & Biases, WhyLabs, zenML, KNIME, RapidMiner, NVIDIA AI Enterprise, watsonx.ai
- OpenSource: MLFlow, Weights & Biases, ClearML
- PaaS: AWS SageMaker, Azure ML

5.3. Referenzen

[Chen et al. 2022], [Treveil et al. 2020], [Haviv et al. 2023], [Osipov 2022], [Tan Wei Hao et al. 2024], [Wilson 2022], [Salama et al.], [Kumara et al.]

6. Systemarchitekturen und Plattformen für Generative KI-Systeme

Dauer: 160 Min.	Übungszeit: 30 Min.

6.1. Begriffe und Konzepte

Generative KI, LLMs, MLflow, Managed MLflow, Azure Machine Learning, Metaflow, Generative KI, LLM, (Stable) Diffusion, Vektor-DB, Embedding, RNN, Transformer, RAG, Agentic Workflows etc.

6.2. Lernziele

LZ 6-1: Kennen verschiedene Integrationsebenen von KI: Anwendungen (z.B. Coding Assistenten), AI-Engineering (z.B. Prompt Engineering), ML-Modellentwicklung (z.B. pytorch), ML-Infrastruktur (z.B. Vektor-DBs).

LZ 6-2: Können das strategische Design von DDD (insbesondere Context Maps) einsetzen, um die Art und den Grad der Integration von KI-Systemen zu bestimmen und zu dokumentieren.

LZ 6-3: Kennen die Qualitätsmerkmale, die für KI-Systeme besonders relevant sind: Verlässlichkeit, Skalierbarkeit, Effizienz, Sicherheit, Wartbarkeit, Interpretierbarkeit etc.

LZ 6-4: Kennen gängige Evaluations-Frameworks, um mit Unbestimmtheit und Fehlern in KI-Systemen umzugehen z.B. LangSmith oder LangFuse.

LZ 6-5: Kennen einige Beispiele von verbreiteten Bibliotheken, Schnittstellen und Tools zur Integration von KI-Modellen.

LZ 6-6: Üben und diskutieren anhand eines Fallbeispiels mit einer ausgedachten Fachlichkeit, Integrationsoptionen für KI in eine bestehende Software-Landschaft abzuwägen.

LZ 6-7: Haben ein grundlegendes Verständnis von generativer KI z.B. LLMs und Stable Diffusion.

LZ 6-8: Haben einen Überblick über die Funktionsweise von LLMs und die zugehörige Begriffswelt: Token, Embedding, RNN, Transformer, Attention…

LZ 6-9: Kennen bekannte Patterns bei der Nutzung von LLMs:

- RAG und Retrieval-Strategien
- Function Calling
- Finetuning
- Assistenten
- Agenten, etc.

LZ 6-10: Kennen typische Use-Cases für RAG wie „Talk to your documents/database/API“.

LZ 6-11: Kennen verschiedene RAG-Techniken:

- Simple RAG
- Context Enrichment Techniques
- Multi-faceted Filtering
- Fusion Retrieval
- Intelligent Reranking
- Query Transformations

- Hierarchical Indices
- Hypothetical Questions
- Dynamic Chunk Sizing
- Semantic Chunking
- Contextual Compression
- Explainable Retrieval
- Retrieval with Feedback Loops
- Adaptive Retrieval
- Iterative Retrieval
- Ensemble Retrieval
- Knowledge Graph Integration
- Multi-modal Retrieval
- RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval

LZ 6-12: Kennen verschiedene Arten von Prompt Engineering (z.B. Few-Shot-Learning, Chain-of-Thought, Role-Playing) und allgemeine Best Practices für das Prompting.

LZ 6-13: Wissen, was Agentic Workflows sind: Reflexion, Werkzeug-Nutzung, Planung, Multi-Agenten-Kollaboration.

LZ 6-14: Wissen welche Design Patterns für Generative AI-Systeme existieren:

- AI Query Router [Simple Router; Ranking-based Router; Learning-based Router
- Layered Caching Strategy Leading to Fine-Tuning
- Multiplexing AI Agents
- Fine-Tuning LLMs for Multiple Tasks
- Blending Rules-Based and Generative Approaches
- Utilizing Knowledge Graphs with LLMs
- Swarm of Generative AI Agents
- Modular Monolith LLM Approach with Composability
- Memory Cognition for LLMs
- Red and Blue Team Dual-Model Evaluation

LZ 6-15: Kennen Techniken zur Evaluation von LLM-Anwendungen: Scoring, Human Feedback, Comparative Evaluation, Model Based Evaluation etc.

LZ 6-16: Kennen bekannte LLMs und Auswahlkriterien: GPT, Claude, Gemini, Llama, Mistral, Luminous etc.

LZ 6-17: Verstehen die Bedeutung von Cost Management für GenAI Applikationen.

LZ 6-18: Kennen einige Beispiele von verbreiteten Bibliotheken, Schnittstellen und Tools im Zusammenhang mit LLM-Anwendungen: OpenAI-API, LangChain etc.

LZ 6-19: Kennen Agentic AI Software Architekturen, AI Agent Architekturkomponenten, Typen vom AI Agentarchitekturen.

6.3. Referenzen

[Koc], [Dibia 2025], [Gradient Flow], [bornstein-radovanic], [Bahree 2024], [Spirin et al.], [Foster 2023], [Parnin]

7. Fallstudien und Praxisprojekte

Dauer: 110 Min.	Übungszeit: 110 Min.

7.1. Lernziele

Die Teilnehmer:Innen …

LZ 7-1: Üben anhand von Fallstudien und Praxisprojekten, das erworbene Wissen in realen Szenarien anzuwenden.

Referenzen

Dieser Abschnitt enthält Quellenangaben, die ganz oder teilweise im Curriculum referenziert werden.

A

- [Agrawal et al.] A. Agrawal, J. Gans, A. Goldfarb: Prediction Machines: The Simple Economics of Artificial Intelligence https://www.predictionmachines.ai/
- [Alake] R. Alake: ML Pipeline Architecture Design Patterns (With 10 Real-World Examples) https://neptune.ai/blog/ml-pipeline-architecture-design-patterns
- [ATLAS] ATLAS - Adversarial Threat Landscape for Artificial-Intelligence Systems. https://github.com/mitre/advmlthreatmatrix

B

- [Bahree 2024] Bahree, A.: Generative AI in Action https://www.manning.com/books/generative-ai-in-action
- [Bhajaria 2022] N. Bhajaria: Data Privacy - A runbook for engineers https://www.manning.com/books/data-privacy
- [Bornstein et al.] M. Bornstein, J. Li, M. Casado: Emerging Architectures for Modern Data Infrastructure https://a16z.com/emerging-architectures-for-modern-data-infrastructure/
- [bornstein-radovanic] M. Bornstein and R. Radovanovic: Emerging Architectures for LLM Applications https://a16z.com/emerging-architectures-for-llm-applications/
- [Burkov 2019] Burkov, A.: The Hundred-Page Machine Learning Book https://themlbook.com/

C

- [Cdteliot] AI Agents: Understanding Their Impact and Functions https://www.perplexity.ai/page/ai-agents-understanding-their-bL1Mg8FeStyUB4o9u3HT5Q
- [Chen et al. 2022] C. Chen, N. R. Murphy, K. Parisa, D. Sculley, T. Underwood: Reliable Machine Learning https://www.oreilly.com/library/view/reliable-machine-learning/9781098106218/
- [Chong et al.] J. Chong, Y. C. Chang: How to Lead in Data Science https://www.manning.com/books/how-to-lead-in-data-science
- [Crowe et al. 2024] R. Crowe, H. Hapke, E. Caveness, D. Zhu: Machine Learning Production Systems https://learning.oreilly.com/library/view/machine-learning-production/9781098156008/
- [CSIRO et al. 2023] CSIRO, Q. Lu, J. Wittle, X. Xu, L. Xhu: Responsible AI: Best Practices for Creating Trustworthy AI Systems https://www.oreilly.com/library/view/responsible-ai-best/9780138073947/

D

- [Dehghani] Z. Dehghani: Data Mesh https://learning.oreilly.com/library/view/data-mesh/9781492092384/
- [Dell'Acqua 2022] Fabrizio Dell'Acqua et al.: Paper: "Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality" https://www.hbs.edu/ris/Publication%20Files/24-013_d9b45b68-9e74-42d6-a1c6-c72fb70c7282.pdf
- [Dibia 2025] V. Dibia with C. Wang: Multi-Agent Systems with AutoGen https://www.manning.com/

books/multi-agent-systems-with-autogen

E

- [Engler et al.] M. Engler, N. Dhamani: Generative AI. Misuse and Adversarial Attacks. https://learning.oreilly.com/library/view/introduction-to-generative/9781633437197/OEBPS/Text/05.html
- [EU AI Act] EU AI Act https://artificialintelligenceact.eu/de/ai-act-explorer/

F

- [Ford et al.] N. Ford, M. Richards, P. Sadalage, Z. Dehghani Software Architecture: The Hard Parts. https://learning.oreilly.com/library/view/software-architecture-the/9781492086888/
- [Foster 2023] D. Foster: Generative Deep Learning, 2nd Edition https://www.oreilly.com/library/view/generative-deep-learning/9781098134174/

G

- [Géron 2022] Aurélien Géron: Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow https://learning.oreilly.com/library/view/hands-on-machine-learning/9781098125967/
- [Gradient Flow] LLM Routers Unpacked https://gradientflow.com/llm-routers-unpacked/

H

- [Hall et al. 2023] P. Hall, J. Curtis, P. Pandey: Machine Learning for High-Risk Applications https://www.oreilly.com/library/view/machine-learning-for/9781098102425/
- [Hall et al. 2023] P. Hall, J. Curtis, P. Pandey: Machine Learning for High-Risk Applications (Chapter 1 Chapter 1. Contemporary Machine Learning Risk Management https://www.oreilly.com/library/view/machine-learning-for/9781098102425/ch01.html#:-:text=Contemporary%20Machine%20Lear,ning%20Risk%20Management)
- [Harvard et al. 2024] Harvard Business Review, E. Mollick, D. De Cremer, T. Neeley, P. Sinha: Generative AI: The Insights You Need. (Generative AI Use Cases) https://learning.oreilly.com/library/view/generative-ai-the/9781647826406/
- [Haviv et al. 2023] Y. Haviv, N. Gift: Implementing MLOps in the Enterprise https://www.oreilly.com/library/view/implementing-mlops-in/9781098136574/
- [Heiland et al. 2023] L. Heiland, M. Hauser, J. Bogner: Design Patterns for AI-based Systems: A Multivocal Literature Review and Pattern Repository. 2023 IEEE/ACM 2nd International Conference on AI Engineering–Software Engineering for AI (CAIN). IEEE, 2023.
- [Hotz] N. Hotz: 15 Data Science Documentation Best Practices https://www.datascience-pm.com/documentation-best-practices/
- [Hotz] N. Hotz: What is a Data Science Life Cycle? https://www.datascience-pm.com/data-science-life-cycle/
- [Hotz] N. Hotz: What is TDSP https://www.datascience-pm.com/tdsp/
- [Huyen 2022] C. Huyen: Designing Machine Learning Systems https://www.oreilly.com/library/view/designing-machine-learning/9781098107956/

J

- [Jarmul 2023] K. Jarmul: Practical Data Privacy https://www.oreilly.com/library/view/practical-data-privacy/9781098129453/
- [Jones] A. Jones: Driving Data Quality with Data Contracts https://learning.oreilly.com/library/view/driving-data-quality/9781837635009/

K

- [Kelleher 2015] John D. Kelleher, Brian Mac Namee, and Aoife D'Arcy: Fundamentals of Machine Learning for Predictive Data Analytics https://mitpress.mit.edu/9780262029445/fundamentals-of-machine-learning-for-predictive-data-analytics
- [Koc] V. Koc: Generative AI Design Patterns: A Comprehensive Guide https://towardsdatascience.com/generative-ai-design-patterns-a-comprehensive-guide-41425a40d7d0
- [Kumara et al.] I. Kumara, R., D. Di Nucci, W. J. Van Den Heuvel, D. A. Tamburri: Requirements and Reference Architecture for MLOps:Insights from Industry https://www.techrxiv.org/doi/full/10.36227/techrxiv.21397413.v1

L

- [Lakshmanan et al.] V. Lakshmanan, S Robinson, M. Munn: Machine Learning Design Patterns https://learning.oreilly.com/library/view/machine-learning-design/9781098115777/
- [Lakshmanan et al.] V. Lakshmanan, S Robinson, M. Munn: Machine Learning Design Patterns Chapter 4 https://www.oreilly.com/library/view/machine-learning-design/9781098115777/ch04.html
- [Lakshmanan et al.] V. Lakshmanan, S Robinson, M. Munn: Machine Learning Design Patterns Chapter 5 https://www.oreilly.com/library/view/machine-learning-design/9781098115777/ch05.html

M

- [Masood et al. 2023] A. Masood, H. Dawe: Responsible AI in the Enterprise https://www.oreilly.com/library/view/responsible-ai-in/9781803230528/
- [ML software architecture] ML software architecture https://appliedaiinitiative.notion.site/ML-software-architecture-790b9f5fcfcf408884287acb82f4d75e
- [Molnar 2024] C. Molnar: Interpretable Machine Learning, 2nd ed. https://christophm.github.io/interpretable-ml-book/

N

- [Nahar et al.] N. Nahar, et al.: A meta-summary of challenges in building products with ml components–collecting experiences from 4758+ practitioners. 2023 IEEE/ACM 2nd International Conference on AI Engineering–Software Engineering for AI (CAIN). IEEE, 2023.
- [NirDiamant] RAG Techniques https://github.com/NirDiamant/RAG_Techniques
- [Nist] NIST AI Risk Management Framework. https://www.nist.gov/itl/ai-risk-management-framework

O

- [Osipov 2022] C. Osipov: MLOps Engineering at Scale https://www.manning.com/books/mlops-engineering-at-scale

P

- [Parnin] Building Your Own Product Copilot: Challenges, Opportunities, and Needs https://arxiv.org/pdf/2312.14231v1
- [Pruksachatkun et al. 2023] Y. Pruksachatkun, M. Mcateer, S. Majudmar: Practicing Trustworthy Machine Learning https://www.oreilly.com/library/view/practicing-trustworthy-machine/9781098120269/

R

- [Reis et al.] J. Reis, M. Housley: Fundamentals of Data Engineering https://learning.oreilly.com/library/view/fundamentals-of-data/9781098108298/
- [Roser 2022] Roser, Max: Brief History of AI: https://ourworldindata.org/brief-history-of-ai

S

- [Salama et al.] K. Salama, J. Kazmierczak, D. Schut: Practitioners guide to MLOps: A framework for continuous delivery and automation of machine learning. https://services.google.com/fh/files/misc/practitioners_guide_to_mlops_whitepaper.pdf
- [Saltz] J. Saltz: The GenAI Life Cycle https://www.datascience-pm.com/the-genai-life-cycle/
- [Sanderson et al.] C. Sanderson, M. Freeman: Data Contracts https://learning.oreilly.com/library/view/data-contracts/9781098157623/
- [Sarkis] A. Sarkis: Training Data for Machine Learning https://learning.oreilly.com/library/view/training-data-for/9781492094517/
- [Savarese] S. Savarese: How AI Agents Will Revolutionize the AI Enterprise https://blog.salesforceairesearch.com/how-ai-agents-will-revolutionize-the-ai-enterprise/
- [Serban] A. Serban, J. Visser: "Adapting software architectures to machine learning challenges." 2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER). IEEE, 2022.
- [Serra] J. Serra: Deciphering Data Architectures https://learning.oreilly.com/library/view/deciphering-data-architectures/9781098150754/
- [Spirin et al.] N. Spirin, M. Balint: Mastering LLM Techniques: LLMOps https://developer.nvidia.com/blog/mastering-llm-techniques-llmops/
- [Studer et al.] S. Studer et al.: Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology https://arxiv.org/abs/2003.05155

T

- [Tan et al.] D. Tan, A. Leung, D. Colls: Effective Machine Learning Teams https://learning.oreilly.com/library/view/effective-machine-learning/9781098144623/
- [Tan Wei Hao et al. 2024] B. Tan Wei Hao, S. Padmanabhan, V. Mallya: Design a Machine Learning System (From Scratch) https://www.manning.com/books/design-a-machine-learning-system-design-from-scratch
- [tdcox] MLOps Roadmap 2024 - DRAFT https://github.com/cdfoundation/sig-mlops/blob/main/roadmap/2024/MLOpsRoadmap2024.md
- [Treveil et al. 2020] M. Treveil, N. Omont, C. Stenac, K. Lefevre, D. Phan, J. Zentici, A. Lavoillotte, M.

Miyazaki, L. Heidmann: Introducing MLOps https://www.oreilly.com/library/view/introducing-mlops/9781492083283/

- [TU Berlin] Architecture of Machine Learning Systems (TU Berlin, SS 2024): https://mboehm7.github.io/teaching/ss24_amls/index.htm

V

- [Vaughan 2020] Vaughan, D.: Analytical Skills for AI and Data Science (AI Use Cases) https://learning.oreilly.com/library/view/analytical-skills-for/9781492060932/
- [Visengeriyeva] Visengeriyeva, L.: Defining Jagged Technological Frontier:https://www.perplexity.ai/page/defining-jagged-technological-iF8sDPVFQEKSdd2oyytztA
- [Visengeriyeva] Visengeriyeva, L.: The Productivity J-Curve of AI: https://www.perplexity
- [Visengeriyeva] Visengeriyeva, L.: AI Agents vs. Traditional Models https://www.perplexity.ai/page/ai-agents-vs-traditional-model-JFf4gKT0RySW_Ehvbxho2g
- [Visengeriyeva] Model Governance, Ethics, Responsible AI (Linksammlung) https://github.com/visenger/Awesome-ML-Model-Governance

W

- [Wang et al. 2024] C. Wang et al.: Quality Assurance for Artificial Intelligence: A Study of Industrial Concerns, Challenges and Best Practices https://arxiv.org/pdf/2402.16391
- [Wilson 2022] B. Wilson: Machine Learning Engineering in Action https://www.manning.com/books/machine-learning-engineering-in-action

Z

- [Zaharia et al.] M. Zaharia et al.: The Shift from Models to Compound AI Systems https://bair.berkeley.edu/blog/2024/02/18/compound-ai-systems/