

Bachelorarbeit

KI-Integration in Cloud Native Platform Engineering: Eine systematische Analyse aktueller Lösungsansätze und deren praktische Anwendung

von

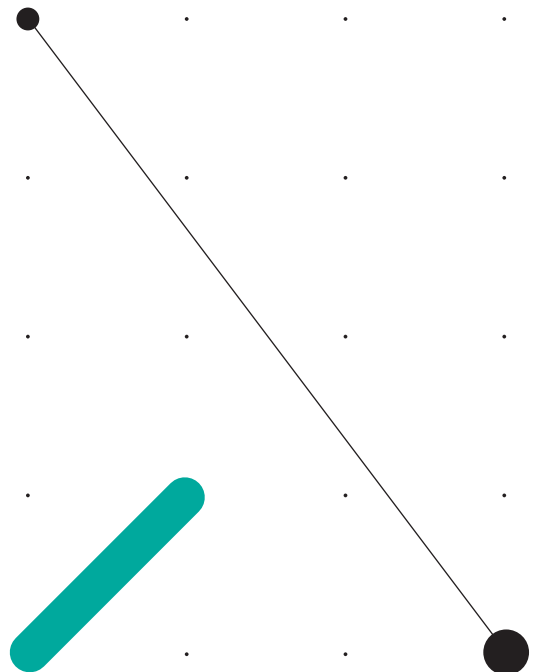
Nils Arnold

zur Erlangung des akademischen Grades

Bachelor of Science

im Studiengang Wirtschaftsinformatik

an der Hochschule Konstanz Technik, Wirtschaft und Gestaltung und der Robert Bosch GmbH



Matrikelnummer: 307179

Abgabedatum: 31.01.2026

Erstbetreuer: Prof. Dr. Johannes Schneider

Zweitbetreuer: Lukas Grodmeier

Abstract

Abstract. . .

Ehrenwörtliche Erklärung

Hiermit erkläre ich, Nils Arnold, geboren am 18. Januar 2003 in Balingen,

(1) dass ich meine Bachelorarbeit mit dem Titel:

„KI-Integration in Cloud Native Platform Engineering: Eine systematische Analyse aktueller Lösungsansätze und deren praktische Anwendung“

bei der Robert Bosch GmbH unter Anleitung von Prof. Dr. Johannes Schneider und Lukas Grodmeier selbstständig und ohne fremde Hilfe angefertigt habe und keine anderen als die angeführten Hilfen benutzt habe;

(2) dass ich die Übernahme wörtlicher Zitate, von Tabellen, Zeichnungen, Bildern und Programmen aus der Literatur oder anderen Quellen (Internet) sowie die Verwendung der Gedanken anderer Autoren an den entsprechenden Stellen innerhalb der Arbeit gekennzeichnet habe.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Ort, Datum

Unterschrift

Inhaltsverzeichnis

Abstract	i
Ehrenwörtliche Erklärung	ii
Inhaltsverzeichnis	iii
1 Einleitung	1
1.1 Problemstellung	1
1.2 Zielsetzung der Arbeit.	2
1.3 Aufbau der Arbeit	2
2 Grundlagen und verwandte Arbeiten	3
2.1 Grundlagen	3
2.1.1 Cloud Native Technologien und Platform Engineering	3
2.1.2 DevOps, CI/CD und Plattformbetrieb	3
2.1.3 Künstliche Intelligenz im Software-Engineering	3
2.1.4 AIOps und verwandte Konzepte	4
2.2 Verwandte Arbeiten	4
3 Methodisches Vorgehen	5
3.1 Vorgehen der Mapping Study	5
3.2 Forschungsfragen	6
3.3 Literatur Analyse Prozess	7
3.3.1 Suchstrategie	7
3.3.2 Auswahlkriterien.	8
3.3.3 Schneeballmethode	8
3.3.4 Datenextraktion	8
3.3.5 Datensynthese	9
4 Ergebnisse	10
4.1 Quantitativ Analyse	10
4.2 Mapping Study	10
4.3 Matching/ Framework	10
4.3.1 Überblick und Kategorisierung der KI-Anwendungen	11

5	Theoretisches Konzept	12
6	Analyse der Bosch Digital Manufacturing Platform	13
7	Diskussion	14
7.1	Beantwortung der Forschungsfragen	14
7.2	Praxisrückschluss	14
7.3	Limitationen.	14
8	Handlungsempfehlung	15
9	Zusammenfassung und Ausblick	16
9.1	Zusammenfassung	16
9.2	Ausblick	16
	Tabellenverzeichnis	17
	Literaturverzeichnis	18

1

Einleitung

Mit der zunehmenden Verbreitung von Cloud-Native-Technologien und Plattform-Engineering Ansätzen, stehen Unternehmen vor der Herausforderung, ihre Software-Entwicklung und den Plattformbetrieb effizient, skalierbar und resilient zu gestalten. Laut der CNF-Studie geben 89 Prozent der befragten Organisation an, Cloud Native-Technologien zu nutzen [1]. Die Komplexität von Infrastrukturen, Plattformen und Entwicklungsumgebung wächst in diesem Umfeld stark. Diese Entwicklung eröffnet zwar enorme Potentiale für Agilität und Innovation, stellt Plattform-Teams aber gleichermaßen vor neue Herausforderungen. Der Wunsch nach Automatisierung durch den Einsatz von Künstlicher Intelligenz wird immer größer.

1.1. Problemstellung

Trotz der flächendeckenden Verbreitung von Cloud-Native Architekturen und dem klaren Fokus vieler Unternehmen auf Automatisierung und Effizienzsteigerung, bleibt die Frage offen, wie Plattform-Teams konkret von fortgeschrittenen Automatisierungs- und KI-gestützten Ansätzen profitieren. Während zahlreiche Unternehmen bereits erste KI-gestützte Tools in ihren Cloud-Native Umgebungen einsetzen, existiert bislang keine systematische und evidenzbasierte Analyse, welche Lösungen tatsächlich Mehrwert für Plattform-Teams schaffen. Die Forschung zu AIOps und KI im Software-Engineering ist zwar umfangreich, doch deren Bezug zu spezifischen Domänen des Plattform Engineerings wie CI/CD-Automatisierung, Infrastruktur-Management und Monitoring bleibt

häufig unscharf. Zudem fehlen praxisnahe Untersuchungen, die auf realen Plattform-Stacks aufbauen und konkrete Pain Points der beteiligten Teams berücksichtigen. Daraus ergibt sich die Notwendigkeit, den aktuellen Stand der Forschung systematisch zu erfassen, mit industriellen Anforderungen abzugleichen und daraus Handlungsempfehlungen für die Integration von KI in bestehenden Plattformlandschaften abzuleiten.

1.2. Zielsetzung der Arbeit

Ziel dieser Arbeit ist es, eine wissenschaftlich fundierte Grundlage für die Integration von Künstlicher Intelligenz in Cloud-Native-Plattform-Engineering-Umgebung zu schaffen. Hierzu wird zunächst im Rahmen einer systematischen Mapping Study untersucht, welche aktuellen KI-Ansätze im Plattform Engineering existieren und wo genau sie Anwendung finden. Darauf aufbauend erfolgt eine Analyse der Bosch Digital Manufacturing Plattform, um bestehende Herausforderungen zu identifizieren und potenzielle Einsatzfelder von KI-Technologien zu evaluieren. Das Ende der Arbeit setzt sich aus der Entwicklung eines praxistauglichen Frameworks zur Bewertung von KI-Potenzialen in Plattforummgebung sowie mit einer prototypischen Implementierung zusammen. Damit leistet die Arbeit sowohl einen wissenschaftlichen als auch einen praktischen Beitrag zur Weiterentwicklung moderner Plattform-Engineering Praktiken.

1.3. Aufbau der Arbeit

Dieser Abschnitt gibt einen Überblick über den strukturellen Aufbau der Arbeit.

2

Grundlagen und verwandte Arbeiten

In diesem Kapitel werden die theoretischen Grundlagen dargestellt, die für das Verständnis der Arbeit notwendig sind, sowie verwandte Arbeiten eingeordnet.

2.1. Grundlagen

2.1.1. Cloud Native Technologien und Platform Engineering

Ein Überblick über Cloud-Native-Prinzipien (Containment, Orchestrierung, Microservices) und die Rolle von Platform Engineering.

2.1.2. DevOps, CI/CD und Plattformbetrieb

DevOps-Praktiken, CI/CD-Pipelines und Betriebsaspekte für zuverlässige Softwarebereitstellung.

2.1.3. Künstliche Intelligenz im Software-Engineering

Anwendungsfelder und Nutzen von KI innerhalb des Software-Engineering-Lebenszyklus.

2.1.4. AIOps und verwandte Konzepte

Definition und Abgrenzung von AIOps, Monitoring, Observability und autonomen Betriebsansätzen.

2.2. Verwandte Arbeiten

Relevante wissenschaftliche Arbeiten, Industriereports und State-of-the-Art-Übersichten werden beschrieben und kritisch eingeordnet.

3

Methodisches Vorgehen

Dieses Kapitel beschreibt das methodische Vorgehen dieser Arbeit. Es erläutert die zugrunde liegende Forschungslogik, die Auswahl des methodischen Ansatzes sowie die Verfahren zur Datenerhebung und -auswertung. Ziel ist es, ein wissenschaftlich fundiertes und zugleich praxisorientiertes Vorgehen aufzuzeigen, das eine systematische Untersuchung der Forschungsfragen ermöglicht. In Abschnitt 3.1 wird eine systematische Mapping Study (SMS) nach Petersen et al [2] angewendet. Abschnitt 3.2 definiert die spezifischen Ziele und konkreten Forschungsfragen dieser Arbeit, welche als Basis für die anschließende Analyse dienen. Darauf aufbauend beschreibt Abschnitt 3.3 den Prozess der Literaturanalyse. Dieser umfasst die Entwicklung einer Suchstrategie, die Definition von Ein- und Ausschlusskriterien, die Schneeballmethode sowie die Schritte der Datenextraktion und -synthese.

3.1. Vorgehen der Mapping Study

Für diese Arbeit wird eine Systematic Mapping Study (SMS) nach den Richtlinien von Petersen, Vakkalanka und Kuzniarz [2] durchgeführt. Diese Methodik dient dazu, den aktuellen Forschungsstand zu einem Themengebiet systematisch zu erfassen, zu kategorisieren und bestehende Forschungslücken zu identifizieren.

Das Vorgehen umfasst die Phasen Planung, Durchführung und Auswertung. In der Planungsphase werden die Forschungsfragen definiert und die Suchstrategie entwickelt,

einschließlich der Auswahl relevanter wissenschaftlicher Datenbanken. Dabei wird gezielt nach bestimmten Keywords gesucht, um Publikationen zu finden, die KI-Anwendungen im Kontext von Platform Engineering adressieren.

Um eine fundierte Datenerhebung und Auswertung sicherzustellen, werden einzelne Prinzipien einer Systematic Literature Review (SLR) nach Kitchenham und Charters [3] berücksichtigt. In der Durchführungsphase werden identifizierte Studien anhand festgelegter Ein- und Ausschlusskriterien geprüft. Zusätzlich wird das Schneeballverfahren nach Wohlin [4] eingesetzt, um die Literatursammlung zu erweitern. Alle Schritte werden dokumentiert, um die Nachvollziehbarkeit und Reproduzierbarkeit sicherzustellen. Die Auswertung erfolgt durch eine systematische Kategorisierung der Studien entlang zentraler Themenfelder des Platform Engineerings. Darauf aufbauend werden Muster, Trends und Forschungslücken identifiziert.

Die Ergebnisse der Mapping Study bilden die Grundlage für die anschließende Analyse der Bosch Digital Manufacturing Platform sowie für die Entwicklung eines Frameworks zur Bewertung von KI-Potenzialen in Cloud-Native Plattformumgebungen.

3.2. Forschungsfragen

Die Forschungsfragen werden durch ein strukturiertes methodisches Vorgehen beantwortet. Jede Forschungsphase ist darauf ausgelegt, das Verständnis über den Einsatz von Künstlicher Intelligenz im Platform Engineering schrittweise zu vertiefen. Die Mapping Study dient der Beantwortung der ersten beiden Forschungsfragen, indem sie eine systematische Übersicht über bestehende KI-Ansätze und deren Anwendungsfelder liefert. Auf Grundlage der Ergebnisse der Mapping Study zielt die dritte Forschungsfrage darauf ab, ein praxisorientiertes Framework zur Beantwortung und Übertragbarkeit von KI-Lösungen zu entwickeln. Die Arbeit ist entlang der folgenden Forschungsfragen strukturiert:

- RQ1:** Welche typische Anwendungsfelder und Herausforderungen bestehen im Platform Engineering, in denen KI-Technologien potenziell Mehrwert bieten können?
Ziel: Systematische Erfassung und Kategorisierung relevanter Use Cases.
- RQ2:** Welche KI-Technologien (einschließlich Frameworks und Tools) finden derzeit im Platform Engineering Anwendung, und welche Formen des maschinellen Lernens und Algorithmen kommen dabei zum Einsatz?
Ziel: Erstellung einer Übersicht über vorhandene KI-Ansätze und deren typi-

sche Einsatzkontexte.

RQ3: Wie lassen sich die identifizierten KI-Lösungen auf typische Anwendungsfälle in Cloud-Native-Platform-Umgebungen übertragen und hinsichtlich ihres Mehrwertes und ihrer Umsetzbarkeit bewerten?

Ziel: Entwicklung eines Bewertungsschemas, das die Passung zwischen KI-Lösungen und spezifischen Platform-Use-Cases beschreibt und die praktische Umsetzbarkeit aufzeigt.

3.3. Literatur Analyse Prozess

Der folgende Abschnitt beschreibt den Ablauf der Literaturanalyse im Rahmen der durchgeführten Mapping Study. Ziel ist es, den methodischen Prozess transparent darzustellen. Dazu werden zunächst die Suchstrategie und die Auswahlkriterien erläutert, gefolgt von der Anwendung der Schneeballmethode. Danach wird die Datenextraktion sowie die Datensynthese beschrieben.

3.3.1. Suchstrategie

Zur Durchführung der Mapping Study wurde eine systematische Suchstrategie angewendet, um relevante wissenschaftliche Publikationen zu identifizieren. Die Literaturrecherche erfolgte in den Datenbanken Google Scholar, SpringerLink, ScienceDirect und ResearchGate, da diese eine breite Abdeckung im Bereich Software Engineering, Cloud Native Technologien und KI bieten. Ziel der Suche war, eine möglichst vollständige Übersicht aktueller Forschungsarbeiten zu KI-Anwendungen im Platform Engineering zu erhalten. Dafür wurden gezielt Suchbegriffe und Kombinationen von Suchstrings verwendet, die zentrale Themen der Arbeit abbilden. Die Suchbegriffe waren hierbei: „Platform Engineering“, „Cloud-Native“, „AIOps“, „Artificial Intelligence“, „Machine Learning“, „MLOps“, „DevOps“ und „Kubernetes Cluster“.

Zur Transparenz und Vollständigkeit sind die exakten Suchstrings sowie ihre logischen Verknüpfungen im Anhang dokumentiert.

3.3.2. Auswahlkriterien

Um relevante Studien und Publikationen zu identifizieren, wurde ein systematischer Auswahlprozess durchgeführt, der auf klar definierten Ein- und Ausschlusskriterien basiert. Eine Studie wurde in die Analyse aufgenommen, wenn sie alle Einschlusskriterien erfüllt und zugleich keinem der Ausschlusskriterien unterlag. Die vollständige Übersicht der Kriterien ist in Tabelle 3.1 dargestellt.

Tabelle 3.1: Auswahlkriterien

Kriterium	Beschreibung
EK1	Die Publikation befasst sich mit dem Einsatz von KI oder Machine Learning im Kontext von Platform Engineering, Cloud-Native-Technologien, DevOps oder AIOps.
EK2	Die Studie beschreibt konkrete KI-Methoden, Anwendungen, Architekturen oder Use Cases, die sich auf Plattformumgebungen beziehen.
EK3	Nur peer-reviewte Veröffentlichungen werden berücksichtigt.
AK1	Arbeiten, die keinen klaren Bezug zu KI im Platform Engineering oder verwandten Domänen aufweisen.
AK2	Studien, die keine empirische Analyse, Evaluation oder methodische Beschreibung ihrer Ansätze enthalten, werden ausgeschlossen.
AK3	Bücher, kommerzielle Reports, technische Dokumentationen und graue Quellen werden ausgeschlossen.
AK4	Studien, die vor 2020 veröffentlicht wurden.

3.3.3. Schneeballmethode

Zur Ergänzung der systematischen Suche wurde eine Schneeballmethode nach den Leitlinien von Wohlin [wohlinGuidelinesConductingSystematic2014] angewendet. Dabei erfolgte sowohl eine Rückwärtssuche als auch eine Vorwärtssuche. Als Ausgangspunkt dienten vier relevante Paper, auf deren Basis zwei Iterationen der Vorwärts- und Rückwärtssuche durchgeführt wurden. Die neu gefundenen Publikationen wurden nach denselben Ein- und Ausschlusskriterien geprüft. Der Prozess wurde beendet, sobald keine weiteren relevanten Studien identifiziert werden konnten. Der gesamte Ablauf der Schneeballmethode ist in Abbildung 1 schematisch dargestellt.

3.3.4. Datenextraktion

Zur Sicherstellung von Konsistenzen und Nachvollziehbarkeit wurde ein strukturierter Prozess zur Datenextraktion umgesetzt. Hierzu wurde eine eigene Extraktionsvorlage entwickelt, die die wesentlichen Merkmale der identifizierten Studien erfasst. Diese Merkmale wurden anschließend entlang von fünf zentralen Dimensionen kategorisiert,

wie in Tabelle 3.2 dargestellt.

Tabelle 3.2: Datenextraktion

Dimension	Beschreibung
Forschungskontext	Beschreibt Ziel, Umfang und Art der Studie.
KI-Ansatz und Methode	Erfasst die verwendeten KI- oder ML-Verfahren.
Plattform-Domänen	Ordnet den Beitrag einem Bereich des Platform Engineerings zu.
Ergebnisse und Use-Cases	Fasst die zentralen Erkenntnisse, Anwendungsfälle oder Evaluationsergebnisse zusammen.
Forschungslücke	Dokumentiert identifizierte Limitationen und Ansätze für zukünftige Arbeiten.

3.3.5. Datensynthesis

Nach der Datenextraktion wurden die Ergebnisse zusammengeführt und ausgewertet, um zentrale Themen, Muster und Forschungslücken zu erkennen. Die ausgewählten Studien wurden nach ihren Inhalten und Schwerpunkten strukturiert und den Forschungsfragen zugeordnet. Zur besseren Übersicht erfolgte die Kategorisierung der Arbeit entlang wichtiger Bereiche des Platform Engineerings. Innerhalb dieser Kategorien wurden die identifizierten KI-Ansätze, Anwendungsfälle und Herausforderungen miteinander verglichen, um wiederkehrende Trends sichtbar zu machen. Die Ergebnisse der Synthese werden anschließend in Form einer Mapping Study dargestellt. Diese Übersicht zeigt, in welchen Themenfelder bereits Forschungsschwerpunkte existieren und wo noch Forschungslücken bestehen.

4

Ergebnisse

Dieses Kapitel präsentiert die Ergebnisse der Arbeit.

4.1. Quantitativ Analyse

Der folgende Abschnitt stellt die Ergebnisse der quantitativen Analyse dar.

4.2. Mapping Study

Darstellung der Ergebnisse der Mapping Study (Kategorien, Häufigkeiten, Trends).

4.3. Matching/ Framework

Beschreibung des Matching-/Framework-Ansatzes und seiner Ergebnisse.

4.3.1. Überblick und Kategorisierung der KI-Anwendungen

Zusammenfassung der identifizierten Kategorien von KI-Anwendungen und deren Einordnung.

5

Theoretisches Konzept

Darstellung des theoretischen Konzepts, seiner Bestandteile und Begründung.

6

Analyse der Bosch Digital Manufacturing Platform

Analyseziele, Untersuchungsgegenstand und Ergebnisse bezogen auf die Bosch Digital Manufacturing Platform.

7

Diskussion

Die Ergebnisse werden eingeordnet, Limitationen diskutiert und Implikationen abgeleitet.

7.1. Beantwortung der Forschungsfragen

Zusammenführung der Ergebnisse zur direkten Beantwortung der Forschungsfragen.

7.2. Praxisrückschluss

Übertragbarkeit und Nutzen der Ergebnisse für die Praxis.

7.3. Limitationen

Grenzen der Studie und Ansatzpunkte für Verbesserungen.

8

Handlungsempfehlung

Konkrete Empfehlungen, Leitlinien oder Maßnahmen, die sich aus den Ergebnissen ableiten lassen.

9

Zusammenfassung und Ausblick

Abschließende Zusammenfassung der Arbeit sowie ein Ausblick auf zukünftige Forschung.

9.1. Zusammenfassung

Die wichtigsten Erkenntnisse und Ergebnisse der Arbeit werden hier zusammengefasst.

9.2. Ausblick

Mögliche zukünftige Forschungsrichtungen und offene Fragen werden hier diskutiert.

Tabellenverzeichnis

3.1 Auswahlkriterien	8
3.2 Datenextraktion	9

Literaturverzeichnis

- [1] Rahul Amte. "Cloud-Native AI: Challenges and Innovations in Deploying Large-Scale Machine Learning Models". In: *ISCSITR - INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING (ISCSITR-IJSRAIML) ISSN (Online): 3067-753X* 6.2 (März 2025), S. 9–18. (Besucht am 05. 11. 2025).
- [2] Kai Petersen, Sairam Vakkalanka und Ludwik Kuzniarz. "Guidelines for Conducting Systematic Mapping Studies in Software Engineering: An Update". In: *Information and Software Technology* 64 (Aug. 2015), S. 1–18. ISSN: 0950-5849. DOI: [10.1016/j.infsof.2015.03.007](https://doi.org/10.1016/j.infsof.2015.03.007). (Besucht am 06. 11. 2025).
- [3] Barbara Kitchenham und Stuart M. Charters. (PDF) *Guidelines for Performing Systematic Literature Reviews in Software Engineering*. <https://www.researchgate.net/publication/3067753X> (Besucht am 06. 11. 2025).
- [4] Claes Wohlin. "Guidelines for Snowballing in Systematic Literature Studies and a Replication in Software Engineering". In: *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*. EASE '14. New York, NY, USA: Association for Computing Machinery, Mai 2014, S. 1–10. ISBN: 978-1-4503-2476-2. DOI: [10.1145/2601248.2601268](https://doi.org/10.1145/2601248.2601268). (Besucht am 06. 11. 2025).
- [5] Muhammad Talha Tahir Bajwa u. a. "CLOUD-NATIVE ARCHITECTURES FOR LARGE-SCALE AI-BASED PREDICTIVE MODELING". In: *Journal of Emerging Technology and Digital Transformation* 4.2 (Aug. 2025), S. 207–221. ISSN: 3006-9726. (Besucht am 25. 10. 2025).
- [6] V. Hemanth Das Chowdary u. a. "DevOps 2.0: Embracing AI/ML, Cloud-Native Development, and a Culture of Continuous Transformation". In: *2024 4th International Conference on Pervasive Computing and Social Networking (ICPCSN)*. Mai 2024, S. 673–679. DOI: [10.1109/ICPCSN62568.2024.00112](https://doi.org/10.1109/ICPCSN62568.2024.00112). (Besucht am 04. 11. 2025).
- [7] Osinaka Chukwu Desmond. "The Convergence of AI and DevOps: Exploring Adaptive Automation and Proactive System Reliability". In: 12.9 (2024).

- [8] Aliyu Enemosah. “Enhancing DevOps Efficiency through AI-Driven Predictive Models for Continuous Integration and Deployment Pipelines”. In: *International Journal of Research Publication and Reviews* Vol 6 (Jan. 2025), S. 871–887. DOI: [10.55248/gengpi.6.0125.0229](https://doi.org/10.55248/gengpi.6.0125.0229).
- [9] Vijay Govindarajan. *Machine Learning Based Approach for Handling Imbalanced Data for Intrusion Detection in the Cloud Environment*. März 2025, S. 815. DOI: [10.1109/ICDT63985.2025.10986614](https://doi.org/10.1109/ICDT63985.2025.10986614).
- [10] Abhishek Gupta und Yashovardhan Chaturvedi. “Cloud-Native ML: Architecting AI Solutions for Cloud-First Infrastructures”. In: *Nanotechnology Perceptions* 20 (Dez. 2024), S. 930–939. DOI: [10.62441/nano-ntp.v20i7.4004](https://doi.org/10.62441/nano-ntp.v20i7.4004).
- [11] Satyadhar Joshi. “Review of Data Pipelines and Streaming for Generative AI Integration: Challenges, Solutions, and Future Directions”. In: *International Journal of Research Publication and Reviews* 6 (Feb. 2025), S. 2348–2357. DOI: [10.55248/gengpi.6.0225.0919](https://doi.org/10.55248/gengpi.6.0225.0919).
- [12] Giridhar Kankanala und Sudheer Amgothu. “AI/ML – DevOps Automation”. In: 13 (Okt. 2024), S. 111–117.
- [13] Karthik Puthraya, Rachit Gupta und Beverly DSouza. “The Role of Cloud-Native Architectures in Accelerating Machine Learning Workflows through Data Engineering Innovations”. In: (Feb. 2025). ISSN: 2945-3437. DOI: [10.5281/ZENODO.15106432](https://doi.org/10.5281/ZENODO.15106432). (Besucht am 05. 11. 2025).
- [14] Gopinath Kathiresan. “Cybersecurity Risk Modeling in CI/CD Pipelines Using Reinforcement Learning for Test Optimization”. In: *International Journal of Innovative Science and Research Technology* (Mai 2025), S. 15–25. DOI: [10.38124/ijisrt/25may339](https://doi.org/10.38124/ijisrt/25may339). (Besucht am 03. 11. 2025).
- [15] Ehsan Azizi Khadem und Ali Movaghar. “From Challenges to Metrics: An LLM-driven DevOps Recommendation System Grounded in Evidence-Based Mappings”. In: *Array* 28 (Dez. 2025), S. 100547. ISSN: 25900056. DOI: [10.1016/j.array.2025.100547](https://doi.org/10.1016/j.array.2025.100547). (Besucht am 03. 11. 2025).
- [16] Dominik Kreuzberger, Niklas Kühl und Sebastian Hirschl. “Machine Learning Operations (MLOps): Overview, Definition, and Architecture”. In: *IEEE Access* 11 (2023), S. 31866–31879. ISSN: 2169-3536. DOI: [10.1109/ACCESS.2023.3262138](https://doi.org/10.1109/ACCESS.2023.3262138). (Besucht am 03. 11. 2025).
- [17] *MLOps Approach in the Cloud-Native Data Pipeline Design — Acta Technica Jaurinensis*. <https://acta.sze.hu/index.php/acta/article/view/581>. (Besucht am 04. 11. 2025).

- [18] Pranav Murthy und Sundeep Bobba. "AI-Powered Predictive Scaling in Cloud Computing: Enhancing Efficiency through Real-Time Workload Forecasting". In: 5.4 (2021).
- [19] Oyekunle Claudius Oyeniran u. a. "AI-driven Devops: Leveraging Machine Learning for Automated Software Deployment and Maintenance". In: *Engineering Science & Technology Journal* 4.6 (Dez. 2023), S. 728–740. ISSN: 2708-8952, 2708-8944. DOI: [10.51594/estj.v4i6.1552](https://doi.org/10.51594/estj.v4i6.1552). (Besucht am 04. 11. 2025).
- [20] Sudip Poudel u. a. "AI-Driven Intelligent Auto-Scaling for Cloud Resource Optimization 1". In: *Journal of Advanced College of Engineering and Management* Vol. 11 (Okt. 2025), S. 27–36. DOI: [10.3126/jacem.v11i1.84521](https://doi.org/10.3126/jacem.v11i1.84521).
- [21] Satheesh Reddy Gopireddy. "Integrating AI into DevOps: Leveraging Machine Learning for Intelligent Automation in Azure". In: *International Journal of Science and Research (IJSR)* 11.6 (Juni 2022), S. 2035–2039. ISSN: 23197064. DOI: [10.21275/SR22619111757](https://doi.org/10.21275/SR22619111757). (Besucht am 03. 11. 2025).
- [22] Satyadhar Joshi. "Introduction to Generative AI and DevOps: Synergies, Challenges and Applications". In: *International Journal of Advanced Research in Science, Communication and Technology* (März 2025), S. 205–225. ISSN: 2581-9429. DOI: [10.48175/IJARSCT-23634](https://doi.org/10.48175/IJARSCT-23634). (Besucht am 03. 11. 2025).
- [23] Vijay Kartik Sikha. "Cloud-Native Application Development for AI- Conducive Architectures." In: *International Journal on Recent and Innovation Trends in Computing and Communication* 11.11 (2023).
- [24] Suprit Pattanayak, Pranav Murthy und Aditya Mehra. "Integrating AI into DevOps Pipelines: Continuous Integration, Continuous Delivery, and Automation in Infrastructural Management: Projections for Future". In: *International Journal of Science and Research Archive* 13.1 (Okt. 2024), S. 2244–2256. ISSN: 25828185. DOI: [10.30574/ijjsra.2024.13.1.1838](https://doi.org/10.30574/ijjsra.2024.13.1.1838). (Besucht am 03. 11. 2025).
- [25] Georgios Symeonidis u. a. "MLOps - Definitions, Tools and Challenges". In: *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*. Jan. 2022, S. 0453–0460. DOI: [10.1109/CCWC54503.2022.9720902](https://doi.org/10.1109/CCWC54503.2022.9720902). (Besucht am 05. 11. 2025).
- [26] Venkata Mohit Tamanampudi. "AI-Enhanced Continuous Integration and Continuous Deployment Pipelines: Leveraging Machine Learning Models for Predictive Failure Detection, Automated Rollbacks, and Adaptive Deployment Strategies in Agile Software Development". In: 10 ().

- [27] Varun Tamminedi. "Automating Kubernetes Operations with AI and Machine Learning". In: *IJFMR - International Journal For Multidisciplinary Research* 6.6 (Dez. 2024). ISSN: 2582-2160. DOI: [10.36948/ijfmr.2024.v06i06.33430](https://doi.org/10.36948/ijfmr.2024.v06i06.33430). (Besucht am 03. 11. 2025).
- [28] Jeanette Uddoh u. a. "AI-Based Threat Detection Systems for Cloud Infrastructure: Architecture, Challenges, and Opportunities". In: *Journal of Frontiers in Multidisciplinary Research* 2.2 (2021), S. 61–67. ISSN: 30509718, 30509726. DOI: [10.54660/.IJFMR.2021.2.2.61-67](https://doi.org/10.54660/.IJFMR.2021.2.2.61-67). (Besucht am 04. 11. 2025).