Sichere Integration von LLMs via WebAPIs

Hochschule für Wirtschaft und Recht Berlin
Berlin School of Economics and Law

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

Sandro Hartenstein, M.Sc.
sandro.hartenstein@hwr-berlin.de
www.hwr-berlin.de 25.6.2025

Motivation

Die Integration von LLMs über WebAPIs schafft neue Sicherheitsrisiken im Software Engineering, die über traditionelle API-Sicherheit [1] hinausgehen. Aufgrund des nicht-deterministischen LLM-Verhaltens reichen bestehende Methoden nicht aus [2]. Ziel ist ein systematischer, risikobasierter Ansatz zur Unterstützung von Entwicklern bei der sicheren Integration von LLMs über APIs.

Frage: Wie adressieren Anbieter die Sicherheit ihrer KI-Modelle über APIs?

Hypothese: Die Robustheit und Fairness von KI-Modellen, die über APIs zugänglich sind, variiert signifikant zwischen Anbietern.

Ziel: Entwicklung eines praxisorientierten Rahmenwerks zur sicheren Integration von LLMs über WebAPIs

Methode

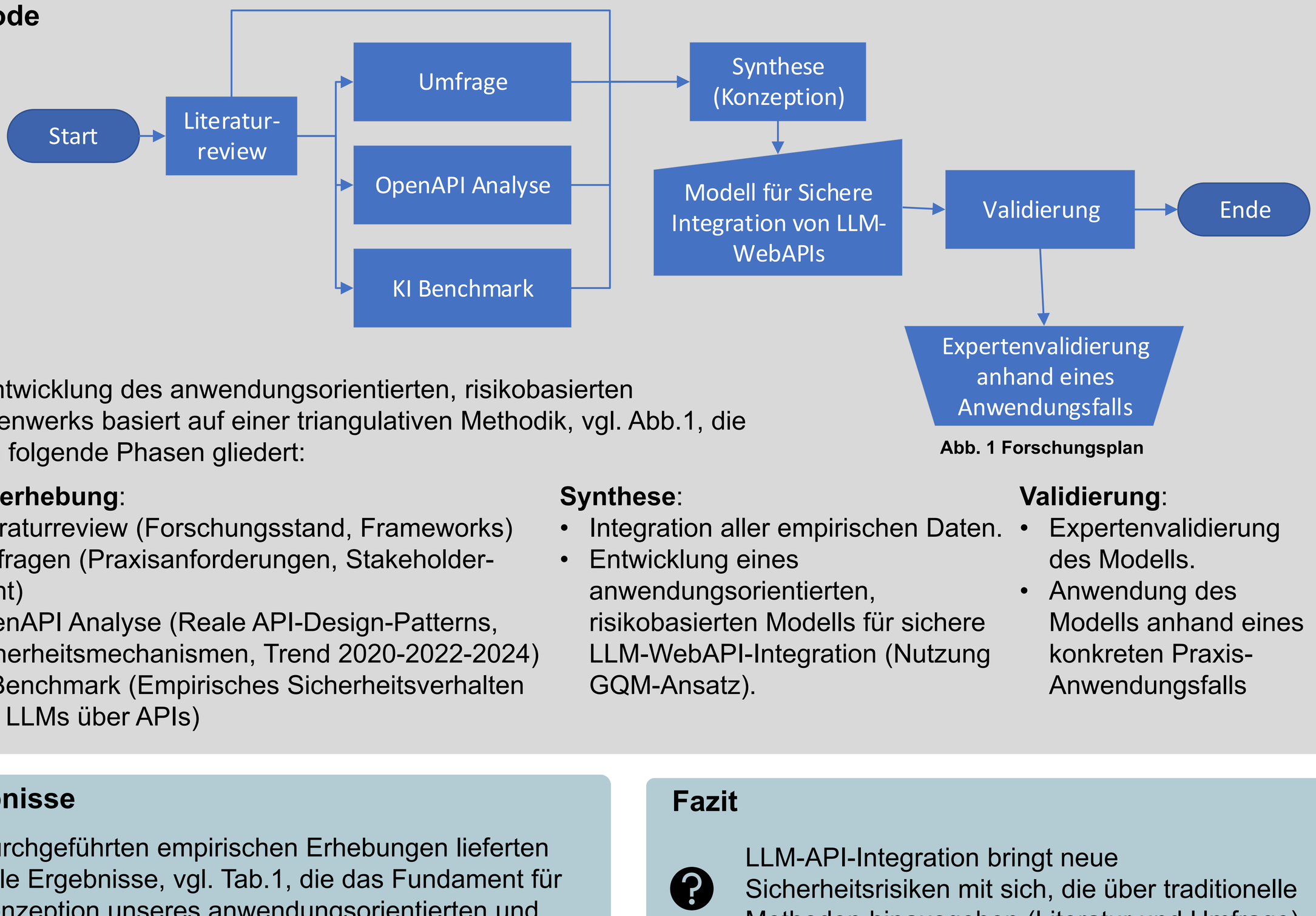

Abb. 1 Forschungsplan

Die Entwicklung des anwendungsorientierten, risikobasierten Rahmenwerks basiert auf einer triangulativen Methodik, vgl. Abb.1, die sich in folgende Phasen gliedert:

Datenerhebung:
- Literaturreview (Forschungsstand, Frameworks)
- Umfragen (Praxisanforderungen, Stakeholder-Sicht)
- OpenAPI Analyse (Reale API-Design-Patterns, Sicherheitsmechanismen, Trend 2020-2022-2024)
- KI Benchmark (Empirisches Sicherheitsverhalten von LLMs über APIs)

Synthese:
- Integration aller empirischen Daten.
- Entwicklung eines anwendungsorientierten, risikobasierten Modells für sichere LLM-WebAPI-Integration (Nutzung GQM-Ansatz).

Validierung:
- Expertenvalidierung des Modells.
- Anwendung des Modells anhand eines konkreten Praxis-Anwendungsfalls

Ergebnisse

Die durchgeführten empirischen Erhebungen lieferten zentrale Ergebnisse, vgl. Tab.1, die das Fundament für die Konzeption unseres anwendungsorientierten und risikobasierten Modells bilden:

Tabelle 1 Ausgewählte Ergebnisse der Datenerhebungsphase

Datenerhebungsphase	Ausgewählte Ergebnisse
Literaturreview	Identifikation von LLM-spezifischen Bedrohungen[3] wie **Prompt Injection**, **Data Poisoning**, **Excessive Agency**. Lücke bei **integrationsspezifischen** Security Frameworks[4] wurde aufgezeigt.
Umfragen in Fokusgruppen [5,6]	Top-Prioritäten in KI-Sicherheit: **Robustheit** (#1), **Zuverlässigkeit**, **Datenschutz**. Bevorzugte Lösungswege: **Verbindliche Standards**, **Bessere Entwickler-Ausbildung**.
OpenAPI Analyse [6] (parsen von 1000 Spezifikationen)	Trend zu **HTTPS (96.2% in 2024)** und modernen Auth-Methoden (**OAuth 2.0 ~43%, API Key ~45%**). **Schwierigkeit**, KI-Services anhand der Spezifikation zu identifizieren.
KI Benchmark via Prompting 2024 [6] (Microsoft,Oracle, IBM, Google)	**Hohe Privacy-Scores (>89%)**, aber **starke Variation bei Fairness (40-73%)** und **Robustheit (66-87%)** zwischen Anbietern. **Jailbreak-Handling** unterscheidet sich.

Fazit

LLM-API-Integration bringt neue Sicherheitsrisiken mit sich, die über traditionelle Methoden hinausgehen (Literatur und Umfrage).

Empirische Analysen und KI-Benchmarks zeigten, dass das Sicherheitsverhalten (Robustheit, Fairness) von LLMs über APIs signifikant zwischen den Anbietern variiert.

Zur Adressierung wird ein anwendungsorientiertes, risikobasiertes Modell entwickelt. Dieses leitet systematisch bei der Identifizierung, Bewertung und Gestaltung von Integrationsrisiken an.

Quellen

[1] Hartenstein, Sandro; Nadobny, Konrad; Schmidt, Steven; Schmietendorf, Andreas (2020): Sicherheits- und Compliance-Management im Lebenszyklus von Web-APIs. Ergebnisse eines Forschungsprojektes an der HWR Berlin / Otto-von-Guericke-Universität Magdeburg. Unter Mitarbeit von Konrad Nadobny, Steven Schmidt und Andreas Schmietendorf. Berlin: Logos Verlag Berlin.
[2] Yao, Yifan; Duan, Jinhao; Xu, Kaidi; Cai, Yuanfang; Sun, Zhibo; Zhang, Yue (2024): A survey on large language model (LLM) security and privacy: The Good, The Bad, and The Ugly. In: High-Confidence Computing 4 (2), S. 100211. DOI: 10.1016/j.hcc.2024.100211.
[3] Vassilev, Apostol; Oprea, Alina; Fordyce, Alie; Anderson, Hyrum (2024): Adversarial machine learning. Adversarial Machine Learning A Taxonomy and Terminology of Attacks and Mitigations (NIST AI 100-2e2023). DOI: 10.6028/NIST.AI.100-2e2023.
[4] OWASP: LLM Security Verification Standard. Online verfügbar unter https://owasp.org/www-project-llm-verification-standard/.
[5] Hartenstein, S.; Schmietendorf, A. (2025): Software Engineering prototypischer KI-Implementierungen im Zusammenhang mit domänenspezifischen Problemen
[6] Akzeptierter Artikel für ICSCA 2025 (https://www.icsca.org) : Hartenstein,Sandro: Bridging the Security Gap: An Empirical Analysis of LLM-API Integration Vulnerabilities and Mitigation Strategies.