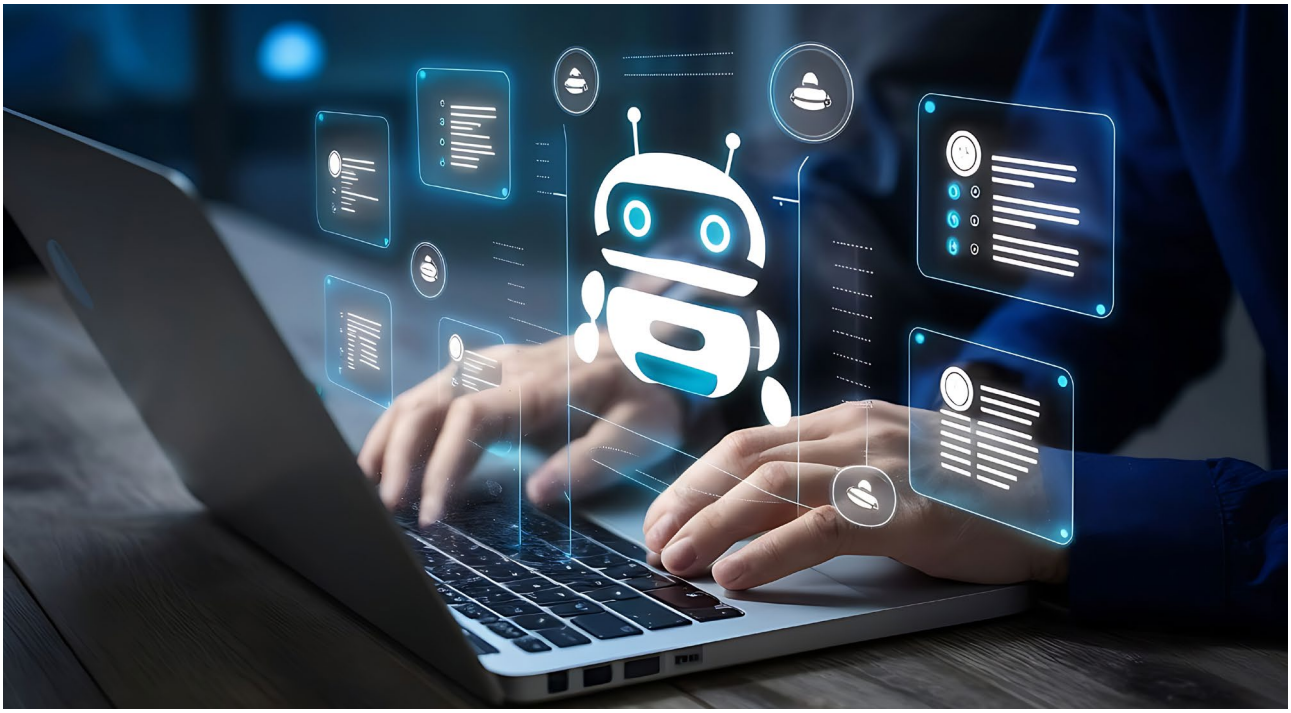
Mittelstand-Digital
Zentrum
Hannover

# Leitfaden

# Implementierung lokaler Chatbots mit RAG-Technologie



Quelle: Naiem/stock.adobe.com, generiert mit KI

Mittelstand-Digital

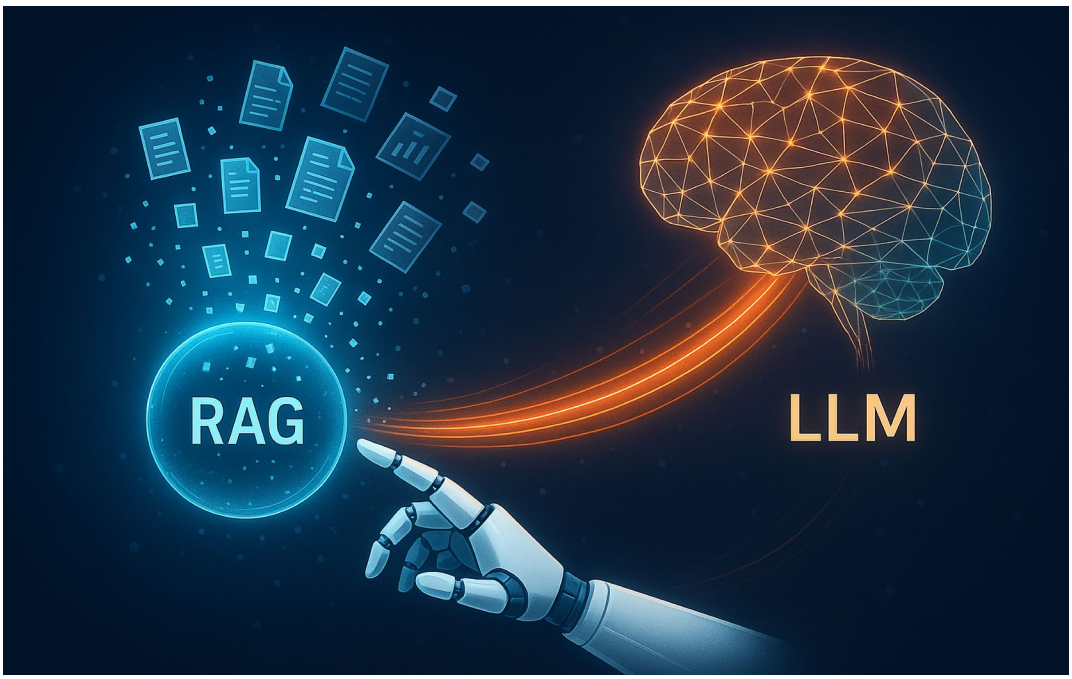
Gefördert durch:

Bundesministerium für Wirtschaft und Energie

aufgrund eines Beschlusses des Deutschen Bundestages

# Einleitung

Dieser Leitfaden ist im Rahmen eines Digitalisierungsprojekts des Mittelstand-Digital Zentrums Hannover entstanden, das gemeinsam mit der ES Edgar Schröder Unternehmensberatungsgesellschaft für Zeitarbeit mbH durchgeführt wurde. Die dabei gewonnenen Erkenntnisse wurden verallgemeinert und in die folgenden Empfehlungen überführt. Der Leitfaden bietet einen umfassenden Überblick über die Implementierung lokal betriebener Chatbots mit Retrieval-Augmented Generation (RAG) in Unternehmen und berücksichtigt aktuelle technologische Entwicklungen, Hardware-Empfehlungen und Best Practices (Stand: Juli 2025).




Quelle: Muhammad/stock.adobe.com, generiert mit KI

## Was ist RAG und warum ist es für Unternehmen relevant?

Retrieval Augmented Generation (RAG) ist eine Methode, die einfache Nutzeranfragen an ein Sprachmodell durch zusätzliche Ressourcen erweitert. Das Sprachmodell leitet anschließend seine Antworten aus diesen Ressourcen ab. Dadurch können beispielsweise Fragen zu spezifischen Unternehmensinhalten mithilfe des Sprachmodells in Verbindung mit zugehörigen Unternehmensdokumenten beantwortet werden (Dethier & Hennekeuser, 2024).

**RAG kombiniert zwei zentrale Technologien:**

- **Information Retrieval**: Das System sucht relevante Informationen aus einer internen oder externen Wissensdatenbank.
- **Generative KI**: Die KI nutzt diese Informationen, um verständliche und kontextbezogene Antworten zu generieren (Bozkurt, 2025).

Die Hauptvorteile liegen auf der Hand: Ohne RAG sind KI-Modelle oft auf statisches Wissen beschränkt, das nur zum Zeitpunkt ihres Trainings aktuell war. In Unternehmen gibt es jedoch ständig neue Informationen über Produkte, Richtlinien oder Prozesse. Ein RAG-gestützter Chatbot kann auf die neuesten internen Dokumente zugreifen. Das verbessert den Kundensupport und stellt die Einhaltung von Compliance-Anforderungen sicher (Bozkurt, 2025)

## Hauptanwendungen für Unternehmen

Basierend auf Interviews und Nutzerstudien lassen sich verschiedene zentrale Anwendungsbereiche identifizieren. Ein wesentlicher Anwendungsfall ist die automatisierte **Informationsgewinnung**: Dabei werden relevante Inhalte aus einer Vielzahl von Dokumenten extrahiert und in kompakter, verständlicher Form aufbereitet. Dies ermöglicht eine schnelle und zielgerichtete Informationsbeschaffung, ohne dass Mitarbeitende umfangreiche Unterlagen selbst durchsuchen müssen.

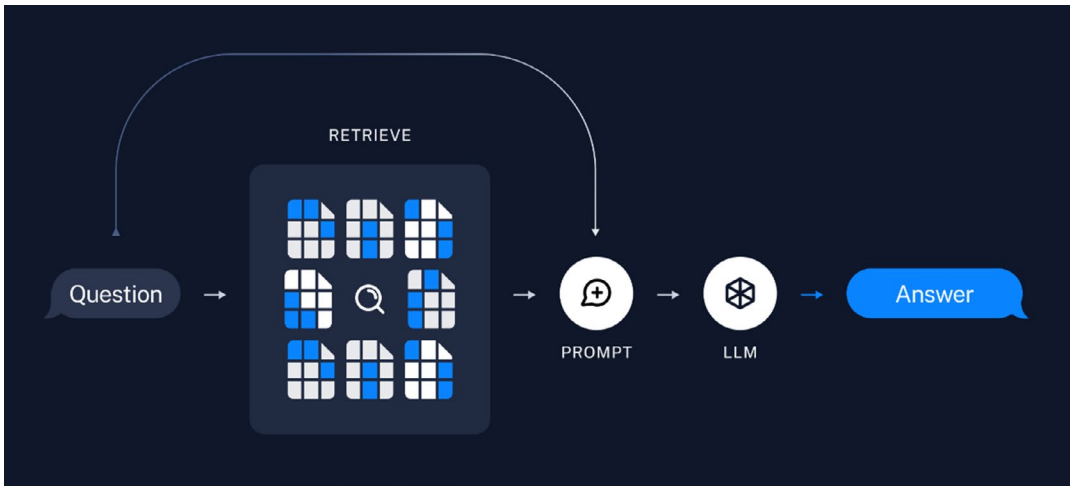
Ein weiteres bedeutendes Einsatzfeld ist der **Kundensupport**. Chatbots mit RAG-Technologie liefern personalisierte und kontextbezogene Antworten und gewährleisten den Zugriff auf stets aktuelle Inhalte wie FAQs, Handbücher und interne Richtlinien. So lassen sich Kundenanfragen effizienter bearbeiten, was zu einer höheren Kundenzufriedenheit und -bindung führt.

Innerhalb des Unternehmens eignet sich RAG-Technologie hervorragend zur **Unterstützung der Wissensvermittlung**. Die Systeme beschleunigen die Einarbeitung neuer Mitarbeitenden, indem sie unkomplizierten Zugang zu relevanten Informationen bieten und gezielt in Mitarbeiterschulungen und Onboarding-Prozesse eingebunden werden. Gleichzeitig entlasten sie zentrale Fachabteilungen wie Personalwesen, IT oder Qualitätsmanagement, indem Routineanfragen automatisiert bearbeitet werden. Dadurch können sich Mitarbeitende stärker auf strategische Aufgaben konzentrieren, was zur Effizienzsteigerung und zur Mitarbeitermotivation beiträgt, da monotone, wiederkehrende Aufgaben reduziert werden.

Ein zusätzlicher praktischer Nutzen liegt in der **Sicherstellung der Compliance**. RAG-Chatbots können bei spezifischen Anfragen gezielt auf einschlägige Vorschriften und unternehmensspezifische Richtlinien verweisen und so die Einhaltung gesetzlicher wie interner Vorgaben unterstützen (Bozkurt, 2025) (Dethier & Hennekeuser, 2024) (Vaziri, 2025).

## Technische Grundlagen von lokalen RAG-Systemen

Ein **RAG-System (Retrieval-Augmented Generation)** kombiniert klassische Information-Retrieval-Technologien mit generativer KI, um kontextbezogene und fundierte Antworten auf komplexe Fragestellungen zu liefern. Besonders in lokalen Anwendungen – etwa in Unternehmen oder auf Forschungsservern – ist es wichtig, diese Systeme datensicher, effizient und nachvollziehbar zu gestalten. Das beigefügte Schaubild veranschaulicht die Funktionsweise eines solchen Systems:




Quelle: LangChain, 2025

### 1. Frageeingabe (Question)

Der Prozess beginnt mit der Formulierung einer **Frage** durch die Nutzenden. Diese Frage bildet den Ausgangspunkt für das gesamte System. Da große Sprachmodelle (LLMs) ohne externen Kontext oft auf ihr internes, statisches Wissen beschränkt sind, ist eine zusätzliche Kontextanreicherung notwendig, um akkurate und aktuelle Antworten zu ermöglichen.

### 2. Informationsbeschaffung (RETRIEVE)

Im nächsten Schritt wird die Frage an ein **Retrieval-Modul** übergeben. Dieses durchsucht eine lokale Wissensdatenbank – häufig bestehend aus Vektorindizes von Dokumenten – und identifiziert die relevantesten Inhalte. Dabei kommen semantische Suchverfahren wie Vektorähnlichkeitsvergleiche (z. B. auf Basis von Sentence Embeddings) zum Einsatz. Ziel ist es, nur die relevantesten Textausschnitte („Chunks") aus den Dokumenten zu extrahieren, die zur Beantwortung der durch den User gestellten Frage nützlich sind. In Tabelle 1 sind wichtige Komponenten des Retrievals und Praxistipps dargestellt:

| Komponente | Funktion | Praxistipp |
|---|---|---|
| **Optical Character Recognition (OCR)** | Extraktion von Text aus Bildern und gescannten Dokumenten | **Docling** von IBM ermöglicht Texterkennung und Berücksichtigung der Dokumentstruktur; ergänzend können Tesseract oder Amazon Textract verwendet werden. |
| **Dokument-Chunking** | Aufteilung von Dokumenten in kleinere, verarbeitbare Einheiten | Chunks von **~400 Tokens** bieten eine gute Balance zwischen Kontexttiefe und Performanz. |
| **Embedding-Modelle** | Umwandlung von Text in Vektoren für die Ähnlichkeitssuche | Deutschsprachige Embeddings wie **jina-embeddings-v2-base-de** zeigen bessere Ergebnisse für deutsche Inhalte. |
| **Vektordatenbank** | Effiziente Speicherung und Suche nach Vektoren | Für lokale Anwendungen eignet sich z.B. **ChromaDB**. |

### 3. Anreicherung des Prompts (PROMPT)

Die extrahierten Textpassagen werden anschließend mit der ursprünglichen Frage zu einem **angereicherten Prompt** kombiniert. Dieses sogenannte Prompt Engineering stellt sicher, dass das Sprachmodell nicht „halluziniert", sondern auf faktisch korrekte und aktuelle Informationen zugreift. Das Prompt enthält also die Benutzerfrage, den zugehörigen kontextuellen Hintergrund aus der Dokumentensammlung und eine Anweisung, wie die Frage beantwortet werden soll. Je nach Anweisung fällt die Qualität der Antwort unterschiedlich aus. Im Folgenden ein paar Faustregeln:

- Weisen Sie dem LLM eine klare Rolle zu.
- Geben Sie Instruktionen zum Umgang mit dem bereitgestellten Kontext, zum Beispiel: „Nutze nur den gegebenen Kontext, ohne eigenes Wissen einzubeziehen."
- Geben Sie Anweisungen zum Umgang mit Unsicherheiten, zum Beispiel: „Kommuniziere bei Unwissenheit transparent."
- Definieren Sie Formatvorgaben für die Antwort, zum Beispiel hinsichtlich der Antwortlänge.

Weitere Prompting-Instruktionen sind unter (Bsharat, Myrzakhan, & Shen, 2024) zu finden.

**4. Antwortgenerierung (LLM)**

Das angereicherte Prompt wird an das **Large Language Model (LLM)** übergeben, das daraus eine kohärente, verständliche und faktenbasierte Antwort generiert. In einem lokalen RAG-System läuft dieses Modell entweder direkt auf eigenen Servern oder in einer isolierten Infrastruktur, um Datenschutz und Reproduzierbarkeit zu gewährleisten. LLMs sind in der Lage, natürlichsprachliche Antworten zu generieren. Kleinere Modelle (3B-7B Parameter) bieten oft das beste Verhältnis von Antwortqualität zu Ressourcenbedarf, was jedoch stark vom jeweiligen Anwendungsfall abhängt. Ein kleiner Überblick aktueller Modelle findet sich weiter unten.

**5. Ausgabe der Antwort (Answer)**

Die generierte Antwort wird abschließend an den Nutzenden zurückgegeben. Durch die Kombination von dokumentbasierter Informationsbeschaffung und generativer Verarbeitung ist die Antwort nicht nur sprachlich überzeugend, sondern auch inhaltlich nachvollziehbar und überprüfbar (LangChain, 2025) (Florian, 2024). Als grafische Benutzeroberfläche kann zum Beispiel das Open-Source-Framework **Streamlit** eingesetzt werden.

# Hardware-Anforderungen für lokale Chatbots mit RAG

Die Anforderungen an die Hardware sind ein entscheidender Faktor für die erfolgreiche Implementierung eines lokalen Chatbots mit RAG-Technologie. Je nach Modellgröße und gewünschter Performance variieren die Anforderungen erheblich.

## GPU- und vRAM-Anforderungen

Für den lokalen Betrieb von LLMs wird eine dedizierte GPU mit ausreichend VRAM empfohlen. VRAM steht für Video Random Access Memory und ist ein spezieller Arbeitsspeicher auf Grafikkarten. Nachfolgend sind drei Leistungsklassen von Hardware exemplarisch dargestellt:

- **High-End:**
  NVIDIA RTX 4090 (24 GB VRAM) oder A100 (80 GB VRAM) - ideal für Profis, die Top-Tier KI-Performance mit ausreichend VRAM benötigen. Erstere ist für Modelle mit der Größe 13B-30B geeignet, wobei die GPU A100 multi-GPU Setups und LLMs mit der Größe 65B+ unterstützt.
- **Gute Balance:**
  NVIDIA RTX 3090 Ti (12 GB VRAM) oder 4080 Ti (16 GB VRAM) - geeignet für lokale LLM-Inferenz und Training und mittelgroße Modelle (7B-13B).
- **Einstieg:**
  NVIDIA RTX 3080 (12 GB VRAM) - gute Einstiegsoption für kleinere Modelle (3B-7B) (Thukral, 2025) (Murallie, 2025).

Für eine optimale Erfahrung wird Hardware ab dem Jahr 2021 mit einer dedizierten GPU mit mehr als 6-7GB GPU-RAM empfohlen (Team, 2024).

## CPU, RAM und Speicher

Auch wenn GPUs den Großteil der Modellberechnungen übernehmen, ist ein leistungsstarker CPU unerlässlich:

- Beispielhafte CPUs: AMD Ryzen 9 7950X, Intel Core i9-13900K
- RAM: Für optimale Leistung werden mindestens 64 GB RAM empfohlen
- Speicher: Schneller SSD-Speicher ist essenziell. Für einfache Sprachmodelle können mehrere TB erforderlich sein (Thukral, 2025) (Dauser & Utomo, 2025).

## Auswahl geeigneter LLMs

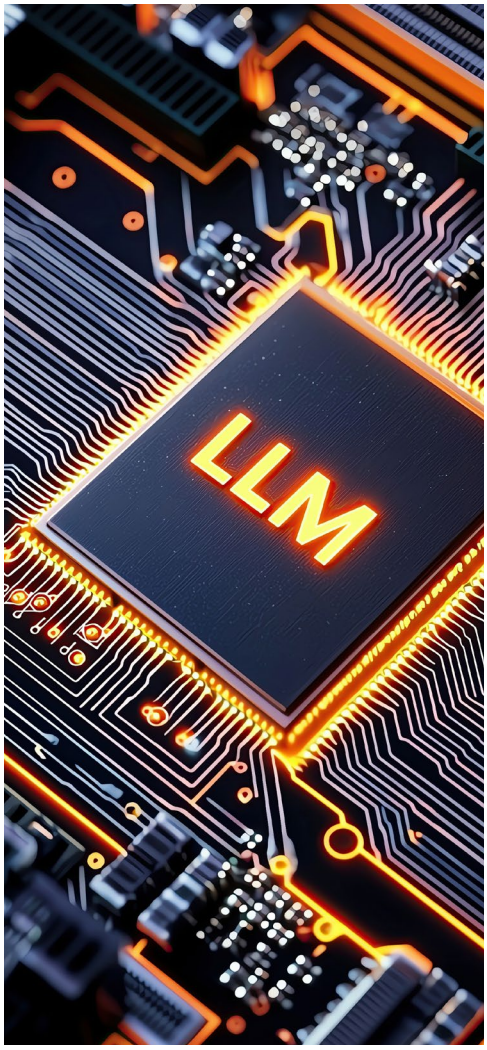
Die Wahl eines geeigneten LLM hängt von verschiedenen Faktoren wie Hardwareverfügbarkeit, Sprachanforderungen und Anwendungsfall ab. Für den lokalen Einsatz eignen sich besonders **Open-Source-Modelle**. Je nach Anwendung und Systemvoraussetzung kommen unterschiedliche Modelle in Frage. Nachfolgend eine kleine Auswahl aktueller Modelle verschiedener Größe:

**1. Llama 3.2:**
Verfügbar in verschiedenen Größen (1B und 3B), gute Balance zwischen Leistung und Ressourcenbedarf (wurde im Rahmen dieses Projektes verwendet).

**2. Mixtral (12B):**
Besonders geeignet für Konversationen in deutscher Sprache.

**3. Llama 3.1 70 B:**
Größeres Modell, welches auch für komplexere Aufgaben verwendet werden kann. Hierzu werden ca. 70 vRAM benötigt.

**4. DeepSeek-r1:**
Großes Modell, das eine ähnliche Qualität wie die OpenAI-Modelle vorweist. Hier sollten mehrere GPUs eingesetzt werden (Murallie, 2025).




Quelle: SoGood/stock.adobe.com, generiert mit KI

## Herausforderungen und Risiken

Trotz der zahlreichen Vorteile, die ein lokal implementierter RAG-Chatbot mit sich bringt (s.o.), stehen Unternehmen und Organisationen bei der Einführung vor mehreren Herausforderungen. Eine zentrale Problematik ist die **Qualität** der zugrundeliegenden **Daten**. Veraltete, unvollständige oder fehlerhafte Informationen können dazu führen, dass der Chatbot ungenaue oder irreführende Antworten liefert. Die Sicherstellung aktueller und korrekter Daten ist daher essenziell für die Verlässlichkeit des Systems. Ein weiteres Hindernis stellt die Integration in bestehende IT-Systeme dar. Die RAG-Modelle müssen so angepasst werden, dass sie mit der vorhandenen **Infrastruktur kompatibel** sind. Dies kann umfangreiche Schnittstellenanpassungen und die Entwicklung maßgeschneiderter Lösungen erfordern. Hinzu kommt der hohe Bedarf an Rechenressourcen. Die Verarbeitung großer Dokumentenmengen sowie der Betrieb anspruchsvoller KI-Modelle beanspruchen **erhebliche Rechenkapazitäten**. Dies kann insbesondere bei der lokalen Ausführung, ohne Auslagerung in die Cloud, zu einem kostspieligen und technisch aufwendigen Unterfangen werden. Insgesamt erfordert die Implementierung eines lokalen RAG-Chatbots daher nicht nur technisches Know-how, sondern auch eine sorgfältige Planung hinsichtlich Datenqualität, Systemintegration und Hardwareanforderungen (Vaziri, 2025) (Waggert, 2024).

Quelle: Johnnii/stock.adobe.com, generiert mit KI

## Fazit und Ausblick

Lokale Chatbots auf Basis von RAG-Technologie eröffnen Unternehmen neue Wege, ihr internes Wissen effizient zu erschließen und gleichzeitig den Kundenservice auf ein neues Niveau zu heben. Durch die Kombination gezielter Informationssuche und generativer Sprachverarbeitung können unternehmensspezifische Inhalte direkt genutzt und in prägnante, kontextbezogene Antworten überführt werden – ohne auf externe Cloud-Dienste angewiesen zu sein. Dies bringt nicht nur Vorteile im Hinblick auf Datenschutz und Datensouveränität, sondern auch bei der Anpassungsfähigkeit an individuelle Prozesse und Anforderungen.

Die erfolgreiche Implementierung eines solchen Systems erfordert jedoch eine sorgfältige Planung: Die Wahl geeigneter Hardware, die Auswahl passender Sprach- und Embedding-Modelle sowie eine durchdachte Datenstruktur spielen eine zentrale Rolle. Gleichzeitig ist die Qualität der genutzten Dokumente entscheidend für die Ergebnisgüte.

Die Praxis zeigt, dass ein lokal betriebener RAG-Chatbot bei kluger Modellwahl und technischer Umsetzung sehr präzise Antworten liefern kann und damit sowohl intern als auch im Kundenkontakt echten Mehrwert schafft. Ein Engpass bleibt allerdings bestehen: Die Hardware – insbesondere ausreichend GPU-Kapazitäten – stellt weiterhin eine zentrale Herausforderung dar, wenn LLMs lokal performant betrieben werden sollen.

Mit Blick auf die Zukunft lässt sich bereits heute eine klare Entwicklungsrichtung erkennen. Zukünftige RAG-Systeme werden voraussichtlich von mehreren technologischen Fortschritten profitieren. So ist mit der Einführung automatisierter Tools zur Datenqualitätsanalyse zu rechnen, die eine objektive Bewertung und gezielte Verbesserung von Trainings- und Referenzdaten ermöglichen. Auch die Einbindung erweiterter Kontextinformationen – etwa durch Wissensgraphen und Metadaten – wird die Antwortgenauigkeit weiter steigern. Darüber hinaus gewinnen Methoden zur Hyperparameter-Optimierung an Bedeutung, die eine systematische Feinjustierung von RAG-Systemen durch parallele, kombinatorische Tests erlauben (Waggert, 2024).

Für Unternehmen, die den Einstieg in diese Technologie planen, empfiehlt sich ein schrittweises Vorgehen. Ausgangspunkt sollte ein klar definierter Anwendungsfall sein – beispielsweise im Bereich des Onboardings oder Kundenservices. Aufbauend darauf sollte in eine geeignete Hardware-Infrastruktur investiert und besonderes Augenmerk auf die Qualität, Struktur und Konsistenz der bereitgestellten Dokumente gelegt werden. Mit diesem Fundament lassen sich die Potenziale der RAG-Technologie voll ausschöpfen und Unternehmen zukunftssicher in Richtung digitaler Wissensnutzung weiterentwickeln.

## Literaturverzeichnis

Bozkurt, H. (21. 01 2025). *VIER*. Von KI-Tipp des Monats: KI & RAG für echt smarte Bots: https://www.vier.ai/unternehmen/blog/ki-und-rag/ abgerufen

Bsharat, S. M., Myrzakhan, A., & Shen, Z. (18. 01 2024). *Principled Instructions Are All You Need for Questioning LLaMA-1/2, GPT-3.5/4*. arXiv.

Dauser, D., & Utomo, m. (2025). *KI-Chatbots Marke Eigenbau?!* f-bb-online.

Dethier, E., & Hennekeuser, D. (21. 02 2024). *Was wünschen sich Nutzer:innen im Mittelstand von RAG-basierten Chatbots?* Von https://www.digitalzentrum-fokus-mensch.de/kos/WNetz?art=News.show&id=2279 abgerufen

Florian. (02. 08 2024). *AI Exploration Journey*. Von The Best Practices of RAG: https://aiexpjourney.substack.com/p/the-best-practices-of-rag abgerufen

LangChain. (21. 05 2025). *LangChain*. Von https://python.langchain.com/docs/tutorials/rag/ abgerufen

Murallie, T. (2025). *Exactly How Much vRAM (and Which GPU) Can Serve Your LLM?*

Team, T. P. (18. 03 2024). *P/Blog*. Von How to Run an LLM Locally with Pieces: https://pieces.app/blog/how-to-run-an-llm-locally-with-pieces abgerufen

Thukral, S. (29. 01 2025). *Rational Systems*. Von Best Hardware for Running Large Language Models (LLMs) Locally: https://rational.co.in/best-hardware-for-running-large-language-models-llm/ abgerufen

Vaziri, D. D. (21. 03 2025). *Mittelstand-Digital Zentrum Fokus Mensch*. Von Retrieval-Augmented Generation: Potenziale, Herausforderungen und Praxisbeispiele: https://www.digitalzentrum-fokus-mensch.de/kos/WNetz?art=News.show&id=2530 abgerufen

Waggert, H. (31. 05 2024). *LinkedIn*. Von KI und Unternehmensdaten. Best Practices in RAG: https://www.linkedin.com/pulse/ki-und-unternehmensdaten-best-practices-rag-herbert-wagger-i0qef/ abgerufen