DHBW
Duale Hochschule
Baden-Württemberg

# Evaluierung von LLM-basiertem QA

Studienarbeit

Studiengang Informatik

Duale Hochschule Baden-Württemberg Stuttgart

Anton Seitz

**Eingereicht am:** 12.06.2025
**Matrikelnummer, Kurs:** 3626401, INF22B
**Betreuer an der DHBW:** Dr. Armin Roth

# Zusammenfassung

Führende Large Language Models (LLMs) werden im Zuge des aktuellen Hypes häufig als Alleskönner dargestellt. Aus vorangegangenen Studien ist jedoch ersichtlich, dass diese Modelle oftmals ein mangelhaftes Faktenwissen aufweisen und selbst bei bekannten Informationen natürlichsprachliche Fragen fehlerhaft beantworten. Sogar das beste getestete Modell, GPT-4, erreichte hier nur 40,3% korrekte Antworten. Diese Diskrepanz zwischen den Erwartungen und der tatsächlichen Leistungsfähigkeit ist sehr problematisch, da sich Millionen von Menschen täglich auf LLMs wie ChatGPT verlassen. Ergebnisse werden selten hinterfragt, was zu Desinformation und Verwirrung führt. Das Ziel ist daher, die Grenzen der Leistungsfähigkeit von LLM systematisch zu erforschen. Insgesamt lassen sich die Ergebnisse der vorangegangenen Studien bestätigen. Wir fanden hier im Schnitt eine Performance von immerhin etwa 69%. Basierend auf diesen Erkenntnissen müsste man daher in etwa einem Drittel der Fälle mit einer falschen Antwort rechnen.

# Inhalt

1

# Kurzbeschreibung der Arbeit

In dieser Studienarbeit wird die Leistungsfähigkeit moderner Large Language Models (LLMs) im Bereich des Question Answering (QA) systematisch untersucht. Dabei betrachten wir ausschließlich Faktenwissen, da LLMs hier notorisch schlecht abschneiden. Ausgangspunkt hierfür ist die Erkenntnis, dass selbst hochentwickelte Modelle wie GPT-4 nur rund 40 % der Fragen korrekt beantworten, obwohl ihnen häufig universelle Problemlöserfähigkeiten zugeschrieben werden (Sun et al., 2023). Zunächst wird ein thematisch geeigneter Textkorpus ausgewählt und für die spätere Evaluierung aufbereitet. Darauf aufbauend werden Testfragen formuliert und Referenzantworten erstellt, um eine belastbare Vergleichsbasis zu schaffen.

Anschließend wird ein ausgewähltes LLM in einer speziell eingerichteten Testumgebung eingesetzt. Hierbei werden sowohl quantitative Metriken wie Genauigkeit und Vollständigkeit als auch qualitative Kriterien zur Bewertung herangezogen. Die Experimentierphase umfasst Tests unter variierenden Modellparametern und Anpassungen der Pipeline, um deren Einfluss auf die Antwortqualität zu erfassen.

Im letzten Schritt erfolgt die systematische Auswertung der gewonnenen Daten. Dabei werden Limitationen der Modelle aufgezeigt und mögliche Optimierungsansätze diskutiert. Die Dokumentation fasst sämtliche Ergebnisse zusammen und liefert Handlungsempfehlungen für den praktischen Einsatz von LLM-basierten QA-Systemen, insbesondere in ressourcenbeschränkten Umgebungen.

2

# Einleitung

## 2.1 Motivation

Heutige Large Language Models (LLMs) wie GPT-4 erreichen teils überraschend niedrige Genauigkeit im Fakten-QA (Sun et al., 2023). Diese Diskrepanz zwischen Erwartung und Realität motiviert die vorliegende Arbeit. Die Zuverlässigkeit und Limitationen solcher Systeme soll hier Anhand eines Test-Environment systematisch untersucht werden.

## 2.2 Zielsetzungen

- Aufbau eines wiederholbaren QA-Test-Environments
- Evaluierung mit vollständigem vs. reduziertem Kontext
- LoRA-basiertes Fine-Tuning auf domänenspezifischen Text
- Systematischer Vergleich der Performance
- Ableitung von Empfehlungen für Praxis-Deployments

Die Arbeit gliedert sich in drei Phasen: eine Vorbereitungsphase mit Literaturrecherche, Korpuserstellung und Methodendefinition, eine Experimentierphase mit Implementierung und Testdurchführung sowie eine abschließende Auswertungs- und Dokumentationsphase. Auf diese Weise sollen fundierte Erkenntnisse über die tatsächliche Leistungsfähigkeit von LLMs im Question Answering gewonnen werden.

3

# Grundlagen und Definitionen

Zunächst werden die nötigen Grundlagen für das Verständnis der Arbeit geschaffen.

## 3.1 Question-Answering- Systeme

Question-Answering-Systeme (QA-Systeme) sind Anwendungen, die automatisch auf natürlichsprachliche Fragen Textantworten liefern. Sie kombinieren Information Retrieval (z. B. Dokumentensuche) und Natural Language Processing (z. B. Named Entity Recognition, Parsing) und KI, um in einem Korpus oder internem Modellwissen die richtige Antwort zu finden (*Question Answering with BERT*, 2023).

### 3.1.1 Arten von Wissen

Knowledge lässt sich in verschiedene Kategorien unterteilen, die für QA-Systeme relevant sind. Basierend auf **types and qualities of knowledge** (De Jong & Ferguson-Hessler, 1996) lassen sich folgende Typen unterscheiden:

- **Factual Knowledge** (auch **Conceptual knowledge**): Dieses Wissen umfasst statische Fakten und Konzepte, z. B. „Berlin ist die Hauptstadt Deutschlands“. QA-Systeme greifen hier häufig auf explizite Datenbanken oder Textpassagen zurück (De Jong & Ferguson-Hessler, 1996).

- **Procedural Knowledge**: Beschreibt Abläufe und Handlungsanweisungen, z. B. Kochrezepte oder Montageanleitungen. QA im prozeduralen Bereich muss oft Schritt-für-Schritt antworten.

- **Metacognitive Knowledge**: Umfasst Wissen über die eigenen Wissensgrenzen und -prozesse, etwa „Ich weiß, dass ich etwas nicht weiß". Für QA weniger direkt relevant, kann aber bei Unsicherheitserkennung helfen.

- **Semantic Knowledge**: Erklärt Bedeutungen und Zusammenhänge zwischen Konzepten, z. B. Taxonomien in Ontologien. Semantisch angereicherte QA-Systeme nutzen dieses Wissen, um Antworten präziser zu formulieren.

- **Contextual Knowledge**: Form von Wissen, das an einen bestimmten Kontext gebunden ist (z. B. aktuelle Nachrichten, persönliche Vorlieben). Open-Domain-QA-Systeme müssen dynamisch darauf zugreifen.

Wir konzentrieren uns in dieser Arbeit auf **Factual Knowledge** („Conceptual knowledge"), da aktuelle LLMs hier erhebliche Defizite zeigen. Studien belegen, dass selbst GPT-4 im Fakten-QA nur ca. 40,3 % korrekte Antworten liefert, obwohl diese Informationen während des Pre-Training oft mehrfach auftauchen (Sun et al., 2023).

### 3.1.2 Typen von QA-Systemen

Im Folgenden werden die üblichen Typen des QA beschrieben und erläutert, welcher davon sich am besten für den bestehenden Anwendungsfall eignet.

- **Extractive QA**:

  Bei dieser Methode erhält das Modell eine Frage und einen zusammenhängenden Textabschnitt (Kontext). Es identifiziert dann genau den oder die Wortgruppen (Spans), die die beste Antwort enthalten. Zum Beispiel sucht ein System in einem Wikipedia-Artikel nach der Textstelle, die erklärt, wofür Einstein den Nobelpreis erhielt (Rajpurkar et al., 2016). Extractive QA ist besonders zuverlässig, da die Antwort wortwörtlich aus dem vorgegebenen Text stammt und so keine inhaltliche Erfindung (Halluzination) erfolgt.
  - **Arbeitsweise:** Das Modell nutzt einen Token-basierten Klassifikator, um Start- und End-Position der Antwort im Kontext vorherzusagen.
  - **Vorteile:** Hohe Präzision und Nachvollziehbarkeit; geringe Gefahr von Halluzinationen.
  - **Nachteile:** Antworten müssen wortwörtlich im Kontext stehen; keine freie Formulierung.

(*Question Answering with BERT*, 2023)

- **Generative QA** Hier erzeugt das Modell die Antwort eigenständig aus Frage und Kontext, statt sie wortwörtlich zu übernehmen. Moderne LLMs wie GPT-Modelle erstellen frei formulierte Fließtext-Antworten (Wolf et al., 2020).
- **Closed-Book QA** Das Modell nutzt nur im Pre-Training erworbenes Wissen, ohne zusätzliche Kontext-Eingabe. Typisches Beispiel sind GPT-basierten Chatbots, die über intern gelernten Wissensspeicher verfügen (Wolf et al., 2020).
- **Open-Domain QA** Systeme greifen auf ein großes Wissensreservoir (z.B. Wikipedia) zu. Ein Retriever identifiziert relevante Dokumente, die ein Reader oder Generator anschließend für die Antwort nutzt (Retrieval-Augmented Generation) (Lewis et al., 2020).
- **Closed-Domain QA** Beschränkt auf ein Fachgebiet (z.B. Medizin). Hier kann das System auf Domänen-Ontologien oder spezialisierte Korpora zugreifen, um präzisere Antworten zu liefern (Kwiatkowski et al., 2020).
- **Cross-Lingual QA** Frage und/oder Kontext können in unterschiedlichen Sprachen sein. Benchmarks wie TyDiQA oder MLQA prüfen die Fähigkeit, in mehreren Sprachen zu antworten (Clark & Dalan, 2019).
- **Semantically Constrained QA** Nutzt zusätzliche semantische Regeln oder Ontologien, um nur Antworten eines bestimmten Typs zuzulassen. Diese Form steigert die Präzision in spezialisierten Anwendungen (Reimers & Gurevych, 2019).

Für unseren Anwendungsfall haben wir uns für Extractive QA entschieden, da hier die Antworten direkt als Textspans aus einem vorgegebenen Dokument extrahiert werden und somit hohe Präzision und Nachvollziehbarkeit gewährleisten. Anders als bei generativen Modellen, die freie Fließtext-Antworten erzeugen und dabei zu Halluzinationen neigen können (Wolf et al., 2020), sucht das Extractive-System gezielt nach der Start- und Endposition der korrekten Antwort im Kontexttext, wie es beispielsweise im SQuAD-Datensatz üblich ist (Rajpurkar et al., 2016). So lassen sich falsche Vorhersagen einfach analysieren und korrigieren, weil der Modell-Output immer klar auf eine Textstelle zurückzuführen ist. Zudem bedarf es kaum Prompt-Engineering, sondern lediglich einer geeigneten Hugging-Face-Pipeline, die in Jupyter-Notebooks effizient auf verschiedene Dokumente skaliert. Diese Kombination aus Verlässlichkeit, schneller Integrationsfähigkeit und geringem Anpassungsaufwand macht Extractive QA für unsere Evaluierung ideal.

## 3.2 Künstliche Intelligenz

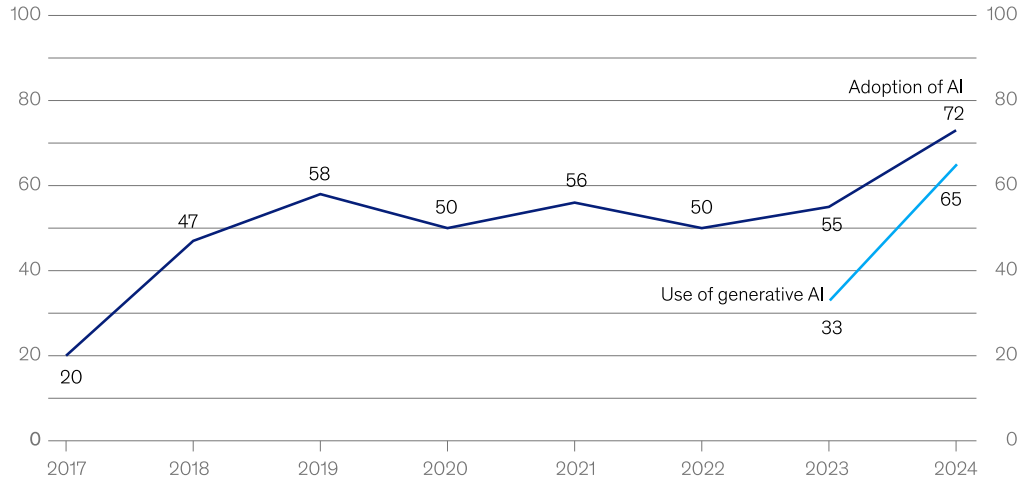
Künstliche Intelligenz (KI) ist der Oberbegriff für Technologien, die Computern ermöglichen, menschliche Denkprozesse wie Lernen, Schlussfolgern und Entscheidungsfindung zu simulieren (IBM, 2024). Moderne KI setzt vor allem auf Machine Learning:

- Computer erhalten eine große Menge von Beispieldaten (z. B. frühere Käufe),
- sie erkennen darin Muster und Zusammenhänge (z. B. welche Produkte häufig gemeinsam gekauft werden) und
- passen ihre internen Parameter so an, dass sie Vorhersagen für neue Daten treffen können (MIT Sloan Management Review, 2019).

Dieses „Musterlernen“ erlaubt es, Konsumenten individuelle Produktempfehlungen auszugeben oder Preise dynamisch anzupassen, was nachweislich die Conversion-Rate erhöht und das Kundenerlebnis verbessert (Harvard Business Review, 2023).

**AI adoption worldwide has increased dramatically in the past year, after years of little meaningful change.**

**Organizations that have adopted AI in at least 1 business function,**[1] % of respondents




[1]In 2017, the definition for AI adoption was using AI in a core part of the organization's business or at scale. In 2018 and 2019, the definition was embedding at least 1 AI capability in business processes or products. Since 2020, the definition has been that the organization has adopted AI in at least 1 function.
Source: McKinsey Global Survey on AI, 1,363 participants at all levels of the organization, Feb 22–Mar 5, 2024

McKinsey & Company

Abbildung 1 — (McKinsey & Company, 2024a) Immer mehr Unternehmen benutzen KI um einen oder mehrere Geschäftsprozesse zu automatisieren. Seit der einfachen Verfügbarkeit von generativer KI, wurde diese auch rapide adaptiert. Im Abschnitt 3.3 wird diese Technologie noch genauer beleuchtet

## 3.3 Generative AI

Generative AI bezeichnet KI-Ansätze, die neue Inhalte wie Texte, Bilder oder Videos erzeugen können (McKinsey & Company, 2024b). Ausschlaggebend war das Transformer-Modell von Vaswani et al. (2017). Dieses Forscherteam bei Google Brain legte mit dem Paper *Attention Is All You Need* den Grundstein für die heute gängigen Sprachmodelle wie ChatGPT. Sie setzten vollständig auf Self-Attention – ein Verfahren, bei dem jedes Element (z. B. ein Wort) alle anderen im Satz „gewichtet", um die für den Kontext wichtigsten Informationen herauszufiltern und zu kombinieren (Vaswani et al., 2017).

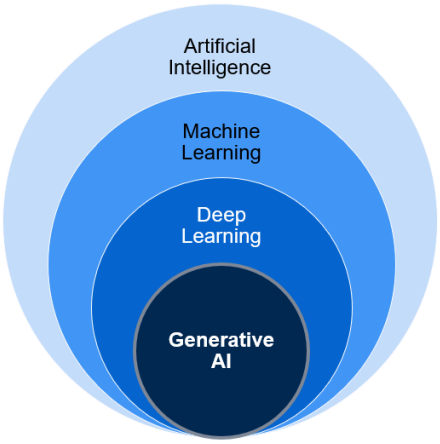



Abbildung 2 — Einordnung von GenAI (SAS Communities Library, 2024)

Unternehmen nutzen Generative AI z.B. um in Echtzeit Produktbilder oder Werbeclips zu erzeugen, die exakt zu Nutzerpräferenzen passen. So kann z. B. eine Online-Modeplattform automatisch Outfits in verschiedenen Stilen generieren (McKinsey & Company, 2024a).

### 3.3.1 Trainingsverfahren

LLMs durchlaufen zwei Phasen:

- **Pre-Training** • Masked Language Modeling (BERT) (Devlin et al., 2019) • Autoregressive Next-Token-Prediction (GPT) (Wolf et al., 2020)
- **Fine-Tuning** Spezialisierung auf Aufgaben oder Domänen. Moderne Systeme wie GPT-4 nutzen zusätzlich **Reinforcement Learning from Human Feedback** (RLHF) (Hu et al., 2021).

## 3.4 Fine-Tuning

Fine-Tuning bezeichnet das Anpassen eines vortrainierten LLM an eine konkrete Aufgabe durch weiteres Training mit gelabelten Beispielen. Das heißt, dass den Trainingsbeispielen jeweils eine korrekte Antwort zugewiesen wurde. Dabei wird die Performance des Modells gezielt auf domänenspezifische Eingaben optimiert.

### 3.4.1 Full-Parameter-Fine-Tuning

Beim klassischen Fine-Tuning werden alle Modellgewichte

$$\boldsymbol{W} \in \mathbb{R}^{d \times k} \tag{1}$$

aktualisiert. Die Gewichte werden dabei durch Minimierung einer passenden Verlustfunktion wie z.B. der Kreuzentropie angepasst:

$$\min_{\boldsymbol{W}} \sum_{i=1}^{N} \mathcal{L}(f(\boldsymbol{x}_i; \boldsymbol{W}), y_i) \tag{2}$$

- Vorteile:
  - Hohe Ausdrucksstärke durch vollständige Anpassung aller Schichten.
- Nachteile:
  - Hoher Speicherverbrauch (alle Parameter müssen im Training gehalten werden)
  - Geringe Wiederverwendbarkeit des Modells (Task-spezifisch)
  - Lange Trainingsdauer und hoher Rechenbedarf
  - Da alle Gewichte geändert werden ist es schwerer, den Einfluss einzelner Parameter auf Vorhersagen nachzuvollziehen.

### 3.4.2 LoRA-Fine-Tuning

**Low-Rank Adaptation (LoRA)** ist eine Methode aus dem Bereich **Parameter Efficient Fine-Tuning** PEFT, bei der nur wenige zusätzliche Gewichte trainiert werden.

Anstatt $\boldsymbol{W}$ direkt zu aktualisieren, wird eine Veränderung

$$\Delta \boldsymbol{W} \tag{3}$$

als Produkt zweier kleiner Matrizen eingeführt:

$$\boldsymbol{W} = \boldsymbol{W}_0 + \Delta \boldsymbol{W} = \boldsymbol{W}_0 + \boldsymbol{A} \cdot \boldsymbol{B} \tag{4}$$

Dabei sind:
- $\boldsymbol{A} \in \mathbb{R}^{d \times r}$
- $\boldsymbol{B} \in \mathbb{R}^{r \times k}$
- $r \ll \min(d, k)$

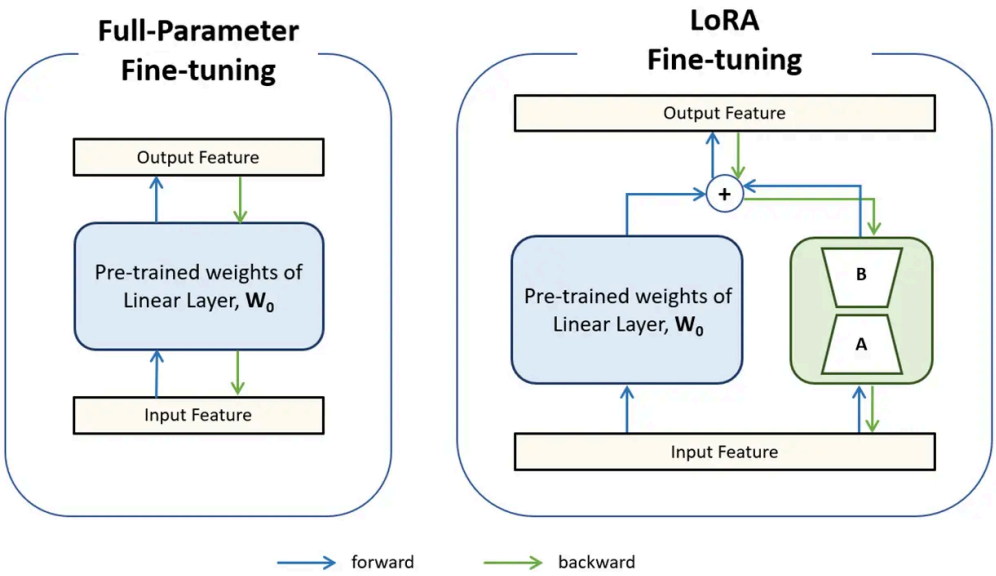



Abbildung 3 — Full-Parameter-Fine-Tuning vs Low-Rank Adaption (LoRA) (Intel Corporation, 2024)

Eine von Intel durchgeführte Studie zeigt, dass beim Full-Parameter-Tuning alle Gewichte (inklusive Bias) eines vortrainierten Layers direkt angepasst werden, während LoRA die ursprünglichen Parameter einfriert und ausschließlich zwei low-rank, bzw. Matrizen A und B trainiert, deren skaliertes Produkt als Residual zum ursprünglichen Layer-Output addiert wird (Intel Corporation, 2024). Dadurch reduziert LoRA den Speicher- und Rechenaufwand beim Fine-Tuning erheblich, da nur ein Bruchteil der Parameter trainiert werden. Das bedeutet, anstelle von $d \cdot k$ Parametern werden nur $(d + k) \cdot r$ Parameter trainiert.

Beispiel: Für $d = k = 768$, $r = 8$ ergibt sich eine Reduktion auf nur ca. 2% der ursprünglichen Parameteranzahl.

- Vorteile:
  - Geringer Speicherbedarf
  - Task-spezifische Adapter lassen sich effizient laden
  - Vortrainiertes Modell bleibt unangetastet
- Nachteile:
  - Potenziell geringere Performanz bei zu kleinem $r$
  - Mehr Aufwand beim Deployment verschiedener Adapter

#### 3.4.3 Mathematischer Vergleich

| Methode | Trainierbare Parameter | Speicherbedarf |
|---|---|---|
| Full-Tuning | $d \cdot k$ | $O(d \cdot k)$ |
| LoRA (rank $r$) | $(d + k) \cdot r$ | $O((d + k) \cdot r)$ |

Da $r$ typischerweise deutlich kleiner ist als $d$ und $k$, fällt der Parameter- und Speicheraufwand bei LoRA im Vergleich zum Full-Tuning erheblich geringer aus. Dadurch eignet sich LoRA besonders für ressourcenoptimiere Umgebungen oder große Modelle.

## 3.5 SQuAD

Ein beliebter Datensatz für QA-Systeme ist SQuAD. Dort wurden in einem strukturierten Format über 100000 Fragen zu Wikipedia-Artikeln aufbereitet (Rajpurkar et al., 2016). SQuAD 2.0 ergänzt unbeantwortbare Fragen (Rajpurkar et al., 2018).

## 3.6 Weitere Benchmarks

- Natural Questions dokumentiert reale Suchanfragen und ist offen für Closed-Book QA (Kwiatkowski et al., 2020)
- HotpotQA erfordert Multi-Hop-Reasoning
- TyDiQA, XQuAD und MLQA testen multilinguale Fähigkeiten (Clark & Dalan, 2019)

## 3.7 Metriken zur QA-Bewertung

In diesem Kapitel werden die zentralen Kennzahlen erläutert, mit denen wir die Qualität von Question-Answering-Systemen messen. Jede Metrik beleuchtet einen spezifischen Aspekt: von der reinen Worttreue bis zur semantischen Tiefe der Antwort. Für unseren Use Case sind besonders robuste Metriken wie F1-Score und Semantic Answer Similarity (SAS) entscheidend, da sie auch bei variierenden Formulierungen zuverlässige Bewertungen ermöglichen (Powers, 2020), (Risch et al., 2021). Im Folgenden sind die wichtigsten Begriffe zum Verständnis der Arbeit kurz definiert.

**Definition 3.7.1** (Accuracy): Misst den Anteil aller Antworten, die als korrekt klassifiziert werden können – unabhängig von der Antwortlänge oder Position.

$$\text{Accuracy} = \frac{\text{Anzahl korrekter Antworten}}{\text{Gesamtanzahl Fragen}} \tag{5}$$

Definition 1 — Accuracy

In unserem Fall bedeutet Accuracy: Die generierte Antwort stimmt entweder exakt mit der Referenzantwort überein oder übersteigt eine definierte Schwelle semantischer Ähnlichkeit. Da jede Frage genau eine Antwort generiert, wird Accuracy hier als binäre Bewertungsmetrik verwendet. Der Einfachheit halber wird auf *halb* korrekte Antworten verzichtet.

**Definition 3.7.2** (F1-Score): Das harmonische Mittel aus Precision und Recall. F1 vereint beide Perspektiven und ist besonders dann sinnvoll, wenn ein ausgewogenes Verhältnis von Genauigkeit und Vollständigkeit gefordert ist.

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

Definition 2 — F1-Score

Der F1-Score ist wird oft bei QA vewendet, wo man sowohl richtige als auch vollständige Antworten benötigt.

**Definition 3.7.3** (Exact Match (EM)): Misst den Anteil der Antworten, die exakt mit den Referenzantworten übereinstimmen. EM ist besonders streng, da nur ganz genaue Textübereinstimmungen als korrekt gewertet werden.

$$\text{EM} = \frac{\text{Anzahl exakter Antworten}}{\text{Gesamtanzahl Fragen}} \tag{7}$$

Definition 3 — Exact Match (EM)

Für QA-Systeme, die exakte Textspans ausgeben, bildet EM den härtesten Qualitätsmaßstab. Für die meisten Fragen ist ein Exact Match nicht nötig um eine Antwort objektiv für korrekt zu befinden.

**Definition 3.7.4** (Semantic Answer Similarity (SAS)): Ein lernbarer semantischer Metrik-Score im Bereich $[0, 1]$. Semantic Answer Similarity (SAS) bewertet, wie inhaltlich ähnlich eine generierte Antwort zur Gold-Antwort ist, selbst wenn sie anders formuliert ist.

$$\text{SAS liegt im Intervall } [0, 1] \tag{8}$$

Hier verwenden wir Cosine-Similariy zur Berechnung der SAS. Die Formel hierfür lautet:

$$\text{cosine similarity} = \frac{\sum a_i b_i}{(\sum a_i^2)^{\frac{1}{2}} (\sum b_i^2)^{\frac{1}{2}}} \tag{9}$$

Definition 4 — Semantic Answer Similarity (SAS)

Diese Metrik ergänzt string-basierte Maße und ist in unserem Use Case wichtig, weil sie semantisch korrekte Paraphrasen erkennt (Evidently AI, 2023).

Diese Metriken kombiniert erlauben eine umfassende Beurteilung:

- **Accuracy** bietet eine einfache Erfolgsquote in binärer Form.
- **F1** bewertet Token- und Span-Ebene direkt.
- **EM** prüft wortwörtliche Korrektheit.
- **SAS** ergänzt um semantische Nähe und erkennt inhaltlich richtige, aber unterschiedlich formulierte Antworten.

Für unseren Use Case ist SAS zentral, da sie sowohl Teil- als auch semantische Übereinstimmung messen und somit robust gegen kleine Formulierungsunterschiede sind.

4

# Realisierung

In diesem Kapitel wird die praktische Umsetzung des Projekts beschrieben. Es wird erklärt, wie der Textkorpus erstellt und aufbereitet wurde, welche Fragesätze daraus abgeleitet wurden und wie darauf basierend Prototypen und Experimente zur Kontextreduktion und Modellanpassung umgesetzt wurden.

## 4.1 Textkorpus

Zunächst wurde ein umfangreicher Textkorpus zusammengestellt. Als Thema wurde **Judo** gewählt, da sich der Autor damit auskennt und Judo sich für Faktenwissen-Tests eignet. Es gibt zahlreiche Details – von Technikklassifizierungen über historische Daten bis hin zu Wettkampfergebnissen, die sich sinnvoll abfragen lassen. Dabei wurden für den Textkorpus folgende Quellen gewählt:

| Quelle | Begründung |
|---|---|
| https://en.wikipedia.org/wiki/Judo | Wikipedia liefert eine umfassende Übersicht über Geschichte, Regeln und Begriffe. |
| https://en.wikipedia.org/wiki/List_of_judo_techniques | Detaillierte Auflistung aller Techniken, ideal für technische Beispiele. |
| https://en.wikipedia.org/wiki/List_of_judoka | Informationen zu bedeutenden Judoka, nützlich für biografische Fragen. |
| https://martialarts.fandom.com/wiki/Judo | Populärkulturelle Perspektive und weiterführende Details. |

| | |
|---|---|
| https://chas-ma.com/JudoManual/Chapter_2%28HistoryofJudo%29.pdf | Fachlicher PDF-Quelltext zur historischen Entwicklung von Judo. |
| https://www.ijf.org/history | Offizielle Historie der International Judo Federation (IJF). |
| https://blackbelttrek.com/judo-vs-jiu-jitsu-the-ultimate-comparison/ | Vergleich mit Jiu-Jitsu, um Abgrenzungen und historische Zusammenhänge zu verdeutlichen. |

## 4.2 Fragesätze

Die Fragesätze liegen in einem flachen JSON-Format vor, bei dem jedes Objekt genau die Felder `question` und `answer` enthält. Im Laufe der Arbeit wurden drei Prototypen entwickelt und evaluiert:

**Prototyp 1 – verbos und unstrukturiert** Im ersten Ansatz enthielten die Antworten oft vollständige Sätze oder sogar ganze Absätze. Diese verbosen Rückgaben führten zu schlechter Performance, da das QA-Modell auf kurze, prägnante Antworten optimiert ist. Zudem waren manche Fragen nicht rein auf Faktenwissen ausgelegt, sondern erforderten etwas längere Erklärungen, was die Auswertung zusätzlich erschwerte.

**Prototyp 2 – atomare Antworten** Um die Performance deutlich zu steigern, wurden die Fragen so angepasst, dass jede Antwort **atomar** ist – also nur das absolut Notwendige enthält, die sogenannte *ground truth*. Beispiel: Anstelle „Judo bedeutet ‚der sanfte Weg' und wurde 1882 von Kanō Jigorō gegründet" steht nur noch „der sanfte Weg". Durch diese Reduktion auf einfache Stichwortantworten verringerte sich der Fehleranteil spürbar.

**Prototyp 3 – Erweiterung und Strukturierung** Im dritten Schritt wurde nicht nur die Atomizität beibehalten, sondern auch das Volumen der Fragen erhöht und eine zusätzliche Kategorisierung eingeführt. Alle Fragesätze wurden anhand definierter Heuristiken in die drei Schwierigkeitsstufen **Easy**, **Medium** und **Hard** sortiert. Die Kriterien hierfür – wie Termfrequenz, Antwortlänge oder semantische Komplexität – werden im Abschnitt „Klassifikation nach Schwierigkeit" ausführlich erläutert. Diese strukturierte Vorgehensweise erlaubt eine gezieltere Analyse des Modellverhaltens je nach Fragenprofil.

## 4.3 Prototypen und Experimente

Dieser Abschnitt dokumentiert die iterative Entwicklung und Evaluierung verschiedener Ansätze zur Beantwortung von Fragen im Kontext eines umfangreichen Judo-Korpus. Ziel war es, die Antwortgenauigkeit zu maximieren und gleichzeitig die Effizienz des Systems zu verbessern.

### 4.3.1 Baseline: Vollständiger Korpus

**Vorgehen**: Für jede Frage wurde der gesamte Textkorpus (bestehend aus mehreren Quellen) als Kontext an das Frage-Antwort-Modell übergeben. Dieser Kontext wurde in einer Text-Datei abgelegt und beinhaltet die o.g. Webinhalte, die einfach aneinander konkateniert wurden. Dabei erreicht er eine Länge von ca. 140 000 Zeichen.

**Beobachtungen**:
- **Laufzeit**: Sehr hohe Antwortzeiten aufgrund des umfangreichen Kontextes: Jede Frage benötigt etwa 2 Minuten Rechenzeit auf dem Laptop des Entwicklers. Später wurde das Jupyter-Notebook auf Google Colab ausgeführt, was eine deutlich schnellere Laufzeit ermöglichte (ca. 10-20x schneller).
- **Genauigkeit**: Besser als in früheren Studien, aber nicht optimal.
- **Token-Limit**: Gefahr des Überschreitens des maximalen Token-Limits des Modells, was zu abgeschnittenen Kontexten führen kann (deepset-roberta-squad2).

## 4.4 Kontextreduktion mittels semantischem Chunking

In diesem Abschnitt wird ausführlich beschrieben, wie der Umfang des Textkorpus systematisch reduziert wurde, um sowohl die Effizienz des QA-Systems zu verbessern als auch die Genauigkeit weitestgehend zu erhalten. Wir erläutern die Motivation, das methodische Vorgehen, die experimentelle Konfiguration, stellen die Ergebnisse grafisch dar und diskutieren zentrale Beobachtungen.

### 4.4.1 Motivation

Der ursprüngliche Textkorpus setzt sich aus verschiedenen Quellen zusammen (Wikipedia-Artikel, PDF-Kapitel, Fandom- und IJF-Webseiten) und umfasst mehr als 140 000 Zeichen. Bei Abfragen unter Verwendung des gesamten Kontexts

zeigte sich, dass die Verarbeitung deutlich länger dauerte und das Modell aufgrund großer Mengen an irrelevanten Informationen weniger präzise antwortete. Ziel der Kontextreduktion ist es daher, den Korpus so stark wie möglich zu verkleinern, ohne wichtige Antworten zu verlieren. Dafür setzen wir semantisches Chunking ein, um nur inhaltlich relevante Abschnitte auszuwählen.

### 4.4.2 Methodik

- Segmentierung in Chunks

  Der vollständige Korpus wurde durch Methoden einer Bibliothek *sentence-transformers* in Absätze bzw. thematische Einheiten unterteilt und ergab zunächst 378 Chunks.
- Filterung nach Mindestlänge

  Nur Chunks mit mindestens 20 Wörtern blieben erhalten, um triviale oder zu kurze Abschnitte auszuschließen. Nach diesem Filter verblieben 215 Chunks, die als **potenziell relevant** galten.

### 4.4.3 Semantische Ähnlichkeitsberechnung

- Embedding-Repräsentation

  Für jeden der 215 Chunks erzeugen wir ein **SBERT-Embedding** (Modell: `all-MiniLM-L6-v2`). Ebenso wird jede Frage in ein SBERT-Embedding überführt.
- Cosine-Similarity

  Für jede Frage berechnen wir die Cosine Similarity zwischen dem Frage-Embedding und jedem Chunk-Embedding. Anschließend sortieren wir die Chunks nach ihrem Similarity-Score ( höherer Score $\longrightarrow$ größere semantische Relevanz).
- Auswahl der Top-K Chunks

  Für verschiedene Reduktionsstufen wählen wir jeweils die **Top K** Chunks aus, um den reduzierten Kontext zu bilden. K entspricht einem Prozentsatz der 215 relevanten Chunks (z. B. 75 % von 215 $\approx$ 161).

### 4.4.4 Experimenteller Versuchsaufbau

Für vier Kontextgrößen (Vollkontext und drei Reduktionsstufen) wurden folgende Abläufe realisiert:

- **Full Context** Der komplette Korpus (alle 378 Chunks) dient als Kontext. – Ergebnis: **Accuracy** = 54/78 = 69.2 %.
- **75 % der relevanten Chunks** K = 0.75 × 215 = 161 Chunks. – Cosine-Scores der ausgewählten Chunks liegen zwischen 0.7320 (höchste Relevanz) und 0.4622 (niedrigste in den Top 161). – Reduzierter Kontext: 108 619 Zeichen (Reduktion um 23.32 %). – **Accuracy**: 49/78 = 62.8 %.
- **50 % der relevanten Chunks** K = 0.50 × 215 = 108 Chunks. – Cosine-Scores liegen zwischen 0.7320 und 0.5442 für die Top 108. – Reduzierter Kontext: 76 622 Zeichen (Reduktion um 45.91 %). – **Accuracy**: 39/78 = 50.0 %.
- **25 % der relevanten Chunks** K = 0.25 × 215 = 54 Chunks. – Cosine-Scores liegen zwischen 0.7320 und 0.6133 für die Top 54. – Reduzierter Kontext: 45 325 Zeichen (Reduktion um 68.00 %). – **Accuracy**: 38/78 = 48.7 %.

**Hinweis:** Je kleiner der Kontext, desto schneller laufen die QA-Anfragen, da das Modell weniger Text verarbeiten muss.

## 4.5 Ergebnisse und Visualisierung

Die folgenden Abbildungen veranschaulichen
1. die **Accuracy** in % für jede Reduktionsstufe und
2. den Zusammenhang zwischen prozentualer Kontextreduktion und Accuracy.

Die Farben basieren auf einer Rot–Gelb–Grün-Skala, wobei niedrige Accuracy-Werte rot und hohe Accuracy-Werte grün eingefärbt werden.

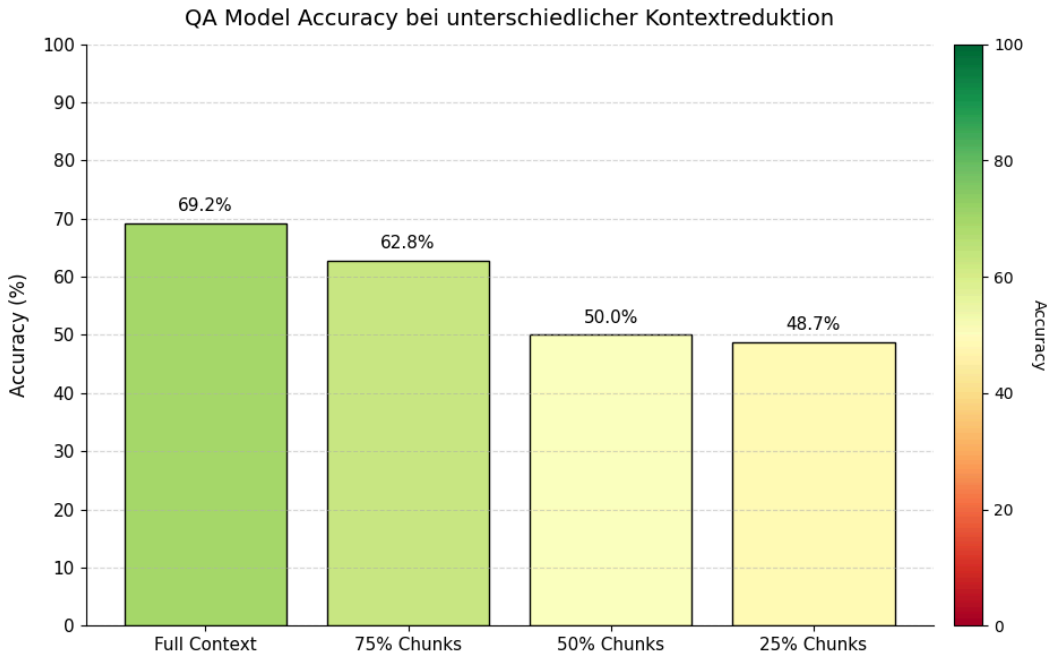



Abbildung 4 — Accuracy bei unterschiedlicher Kontextreduktion (Full, 75 %, 50 %, 25 %) – farbcodiert von Rot (niedrig) bis Grün (hoch).

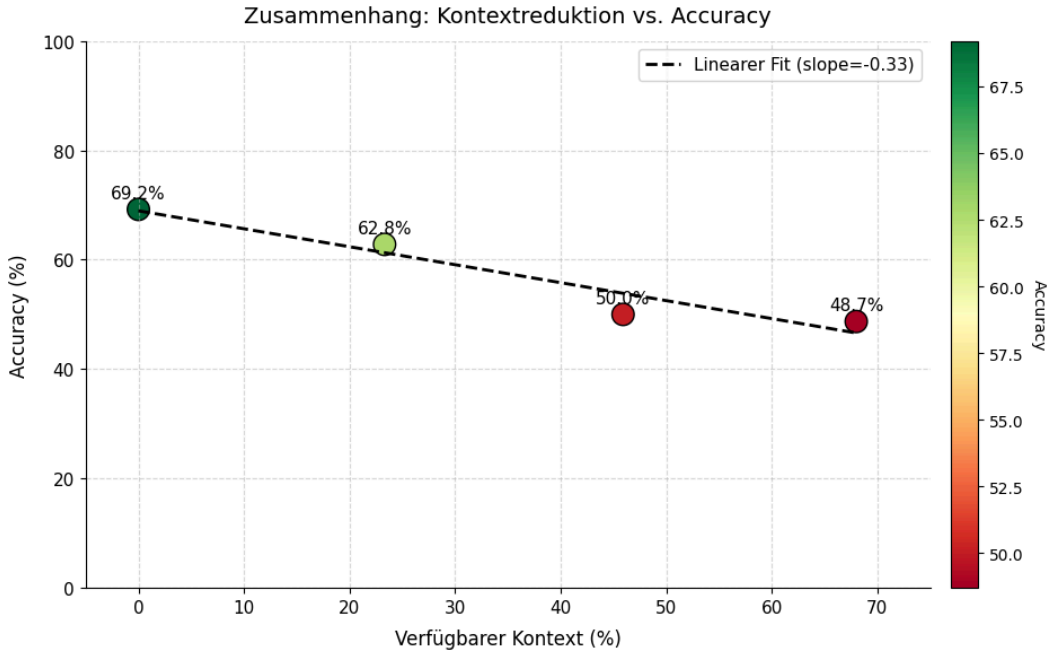



Abbildung 5 — Zusammenhang: Prozentuale Kontextreduktion vs. Accuracy mit linearer Trendlinie.

**Interpretation der Resultate**

- **Full Context** (0 % Reduktion): Höchste Accuracy (69.2 %).
- **75 % Chunks** (23.32 % Zeichenreduktion): Accuracy sinkt moderat auf 62.8 %.

- **50 % Chunks** (45.91 % Zeichenreduktion): Accuracy fällt auf 50.0 %.
- **25 % Chunks** (68.00 % Zeichenreduktion): Accuracy liegt bei 48.7 %, der Informationsverlust zeigt deutliche Auswirkungen.

## 4.6 Beobachtungen

- **Starker Genauigkeitsverlust unter 50 % Reduktion**

  Ab 50 % Reduktion sinkt die Accuracy auf 50.0 % oder darunter, was für faktische QA-Anwendungen zu ungenau ist. Wenn der für eine bestimmte Frage relevante Teil des Textkorpus nicht mehr im reduzierten Kontext enthalten ist, entstehen Nonsense-Antworten. Dies konnte hier allerdings nichtmehr weiter behandelt werden, da man manuell wenig Einfluss auf das interne Chunking-Vefahren. Diese wurde hier besonders bei der 50% Schwelle bemerkbar, wo teilweise auch gar keine Antwort geliefert wurde.

- **Semantische Qualität der Chunks**

  Top-Chunks (Score > 0.7) enthalten häufig Definitionen oder Listen mit QA-relevanten Fakten (z. B. Judo-Grundbegriffe). Chunks mit Scores < 0.5 liefern eher allgemeine oder philosophische Inhalte und sind weniger hilfreich.

- **Kompromiss zwischen Vollständigkeit und Präzision**

  Eine adaptive Auswahlstrategie könnte sinnvoll sein, indem man beispielsweise alle Chunks mit einem Score ≥ 0.6 einbezieht, statt fixe Prozentsätze zu verwenden.

- **Optimierungsmöglichkeiten**
  - Adaptive K-Wahl: Anstatt fixer Prozentsätze (75 %, 50 %, 25 %) könnte die Chunk-Anzahl dynamisch anhand der Score-Verteilung gewählt werden.
  - Strukturiertes Context-Building: Chunks, die Überschriften oder Definitionen enthalten, priorisieren, um schnelle Treffer bei einfacheren Fragen zu erzielen.

Die semantische Chunk-Reduktion erweist sich als effektive Methode, um QA-Performance und Effizienz zu steigern, solange man nicht unter eine kritische Chunk-Schwelle (≈ 50 %) fällt. Bei 75 % Reduktion (– 23.32 %) erreicht man mit 62.8 % Accuracy einen guten Kompromiss. Weitere Optimierungen sind über adaptive Auswahlkriterien realisierbar, um sowohl Accuracy als auch Effizienz zu maximieren.

### 4.6.1 Fine-Tuning mit Low-Rank Adaption (LoRA)

**Vorgehen**:
- Das vortrainierte Modell wurde mittels LoRA auf den spezifischen Judo-Korpus feinjustiert.
- LoRA ermöglichte effizientes Fine-Tuning durch Anpassung einer kleinen Anzahl von Parametern, wodurch der Speicherbedarf reduziert wurde.

Für diesen konkreten Usecase hat das Finetuning allerdings keinen Genauigkeitszuwachs im Vergleich zum Basis-Modell bewirkt. Es wurden hier erneut

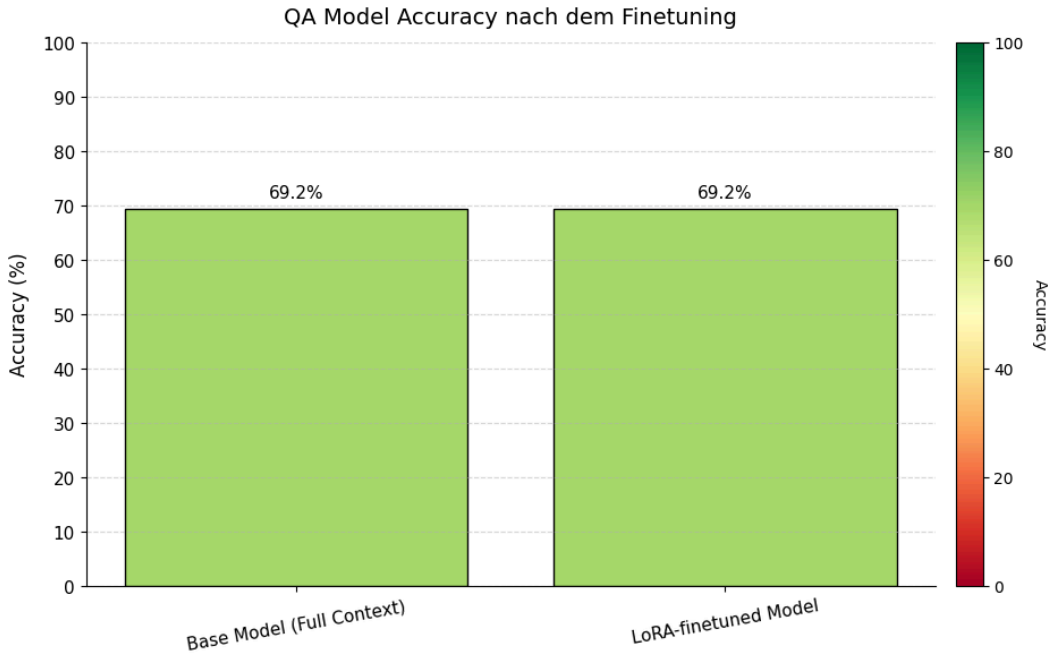



Abbildung 6 — Fine-Tuning liefert in diesem Fall zwar teils andere und subjektiv bessere Antworten, erhöht aber nicht die Accuracy

### 4.6.2 Evaluation der Modelle

In der ersten Evaluierungsphase kam eine **rein stringbasierte** Methodik zum Einsatz, bei der Antworten als korrekt galten, wenn sie exakt mit den Musterantworten übereinstimmten oder eine hohe Zeichenübereinstimmung (≥ 80 %) aufwiesen. Dieses Verfahren zeigte allerdings deutliche Schwächen:

- **Synonyme und Namensvarianten** werden nicht erkann: „Jigoro Kano“ vs. „Kanō Jigorō“ oder „International Judo Federation“ vs. „IJF“.

- **Unterschiedliche Formulierungen und Satzstellungen** gelten als falsch, z. B. „sanfter Weg“ vs. „der sanfte Weg“ oder „1882 gründete Kanō Jigorō den Kōdōkan“ vs. „Der Kōdōkan wurde 1882 von Kanō Jigorō gegründet“.
- **Mehrdeutigkeit bei offenen Fragen**, etwa „Nenne einen Hüftwurf“, erlaubt mehrere gültige Antworten, die stringbasiert schwer zu erfassen sind.

Aus diesen Gründen wurde die Evaluierung auf eine **semantische** Methodik umgestellt. Anstelle des Fuzzy Matching wird die **Cosine Similarity** zwischen der Einbettung der Modellantwort und der Einbettung der Referenzantwort herangezogen. So können inhaltlich identische, aber unterschiedlich formulierte Antworten zuverlässig als korrekt bewertet werden.

Zur ganzheitlichen Beurteilung der Prototypen wurden folgende Metriken definiert:

- **Antwortgenauigkeit**: Semantische Ähnlichkeit zwischen erwarteter und generierter Antwort, gemessen via Cosine Similarity (Schwellenwert z. B. 0.70).
- **Laufzeit**: Durchschnittliche Zeit, die das Modell zur Beantwortung einer einzelnen Frage benötigt.
- **Ressourcennutzung**: Speicher- und Rechenzeitaufwand während der Inferenz, um Effizienz und Skalierbarkeit abzuschätzen.

## 4.7 Klassifikation nach Schwierigkeit

In Anlehnung an das *Head-to-Tail*-Paper wurde ein mehrstufiges Schema entwickelt, um die Fragen systematisch in **Easy**, **Medium** und **Hard** zu unterteilen (Sun et al., 2023). Ziel war es, eine nachvollziehbare Balance zwischen **häufig vorkommendem Basiswissen** und **tiefgehenden Spezialfragen** herzustellen. Die Einteilung erfolgte in einem iterativen Prozess, bei dem quantitative Heuristiken mit qualitativen Einschätzungen kombiniert wurden.

Die Klassifikation basiert auf vier zentralen Heuristiken:

- **Frequenz und Prominenz**

Zunächst wurde die Verteilung von Schlüsselbegriffen im Korpus analysiert. Häufig zitierte Begriffe wie **judo**, **Kanō Jigorō** oder **Kōdōkan** markieren grundlegende Konzepte und bilden damit das Fundament für **Easy**-Fragen. Selten auftretende oder nur in Fachabschnitten erwähnte Terme weisen dagegen auf eine höhere Schwierigkeit hin.

- **Informationsdichte und Antwortkomplexität**

Der Umfang und die Struktur der erwarteten Antworten wurden berücksichtigt: Sehr kurze, prägnante Antworten (ein oder zwei Wörter) kennzeichnen Fragen der Stufe **Easy**. Im Gegensatz dazu erfordern mittellange Antworten in zusammengesetzten Fachbegriffen (**Medium**), während lange oder mehrteilige Antworten —etwa diejenigen, die Kombinationen von Datum, Ort und Person enthalten—typischerweise als **Hard** eingestuft wurden. In der Praxis zeigte sich, dass übermäßig komplexe Frageformate die QA-Performance deutlich verschlechtern und daher eher vermieden wurden.

- **Kognitive Anforderungen und Kontextverknüpfung**

Nicht nur die Länge, sondern auch der Grad der gedanklichen Verknüpfung spielt eine Rolle:

**Easy**-Fragen fordern reines und gängiges Faktenwissen.

**Medium**-Fragen setzen eine Einordnung ins historische oder terminologische Umfeld voraus (z.B. *In welchem Jahr wurde der Kōdōkan gegründet?*).

**Hard**-Fragen verlangen die Verknüpfung mehrerer Aspekte, etwa wenn es gilt, eine Person direkt mit einem historischen Ereignis zu verbinden.

- **Semantische Ambiguität**

Schließlich wurde geprüft, wie eindeutig eine Antwort im Text lokalisiert ist. Antworten, die mehrfach in identischer Form auftauchen, neigen zu moderater Schwierigkeit (**Medium**), da die korrekte Stelle nicht immer sofort ersichtlich ist. Einzigartige oder sehr verstreut gelagerte Antwortpassagen erhöhen die Schwierigkeit auf **Hard**, weil das Modell den relevanten Span präzise identifizieren muss.

Die Fragen wurden manuell nach definierten Heuristiken in Schwierigkeitsgrade unterteilt. Mit 36 % Easy, 24 % Medium und 40 % Hard ergibt sich eine ausgewogene Verteilung, die den Anforderungen des Use Cases genügt. Das entspricht je 28, 19 und 31 Fragen, also in Summe 78. Zuvor wurden einige Fragen entfernt, die z.B. nicht objektiv beantwortbar waren oder andere Mängel aufwiesen.

## 4.8 Beispiele der Einordnung

Um das Schema anschaulich zu machen, werden hier exemplarische Fragestellungen je Kategorie aufgelistet:

**Easy**

Fragen aus dem Bereich der grundlegenden Terminologie und Farben, die in Einsteigerliteratur und Zusammenfassungen häufig erwähnt werden: – *What does judo mean?* – *What color belt do novices wear?*

**Medium**

Fragen, die den historischen oder organisatorischen Kontext erfordern und moderat komplexe Antworten liefern: – *From which martial art did judo originate?* – *What influenced European and Russian judoka?*

**Hard**

Tiefgehende Detailfragen zu speziellen Techniken, historischen Figuren oder seltenen Regelaspekten, die nur in Fachtexten oder speziellen Quellen zu finden sind: – *Name a forbidden sacrifice throw in competition.* – *Who succeeded Aldo Torti as IJF president?*

Aufgrund der überschaubaren Fragenanzahl war die Klassifikation hier manuell möglich. In zukünftigen Tests von QA-Systemen wäre es sinnvoll diese Einordnung durch ein LLM durchzuführen. Dies wurde hier ebenfalls probiert, allerdings hatte das dabei verwendete LLM Schwierigkeiten die Fragen konsistent nach den definierten Heuristiken zu klassifizieren.

5

# Evaluierung

In diesem Abschnitt begründen wir die Wahl der verwendeten Metrik für das Question-Answering-System. Wir fokussieren uns ausschließlich auf die SAS, gemessen als Cosine Similarity zwischen Antwort-Embeddings, mit einem Schwellenwert von 0.7.

## 5.1 Warum nur SAS?

Für QA-Systeme, die in einem homogenen Korpus kurze, atomare Fakten abfragen, sind stringbasierte Metriken wie **Exact Match** (EM) oder der tokenbasierte **F1-Score** grundsätzlich einfach zu implementieren. Allerdings zeigen sich folgende Nachteile:

- **String-Variationen**: Kleinste Unterschiede in Groß-/Kleinschreibung oder Präpositionen („sanfter Weg„ vs. „der sanfte Weg“) führen bei EM oft zu *falsch*.
- **Paraphrasen**: In vielen Fällen ist eine inhaltlich korrekte Paraphrase (*maximum efficiency, minimum effort* statt *maximum efficient use of energy*) möglich, wird aber von reinen String-Vergleichen nicht erkannt.
- **Fehleinschätzung von Teilantworten**: Der F1-Score auf Token-Ebene kann zwar Teilkorrektheit bewerten, nimmt aber an, dass beide Antworttexte dieselben Token-Vokabulare verwenden (Stoppwörter, Zeichensetzung etc.) Abschnitt 4.6.2.

Durch den Einsatz von SAS und Cosine Similarity in Kombination mit der Bibliothek sentence-transformers werden genau diese Einschränkungen umgangen:

1. Robustheit gegen Paraphrasen

   SAS vergleicht semantische Embeddings. Zwei unterschiedlich formulierte, aber inhaltlich identische Antworten erzielen eine hohe Cosine-Similarity ($\geq 0.7$).

2. Toleranz gegenüber kleinen Abweichungen

   Selbst wenn Wörter weggelassen oder ergänzt werden („Judo-Anzug„ vs. „Judo-Uniform“), bleiben semantisch nahe Embeddings eng beieinander. Stringmetriken würden hier häufig scheitern.

3. Einfachheit der Umsetzung

   Mit einem vortrainierten SBERT-Modell (z. B. `all-MiniLM-L6-v2`) ist es in wenigen Zeilen möglich, jede Modellantwort und die Referenzantwort in Vektoren zu überführen und die Cosine Similarity zu berechnen.

Aus diesen Gründen haben wir uns entschieden, ausschließlich SAS mit einem festen Schwellenwert von 0.7 als alleiniges Bewertungsverfahren einzusetzen.

## 5.2 Implementierung von SAS

Für jede Frage gehen wir wie folgt vor:

1. Wir erstellen die Embeddings für die Referenzantwort $A_{\text{gold}}$ und für die Modellantwort $A_{\text{pred}}$ mittels eines SBERT-Modells:

$$\boldsymbol{e}_{\text{gold}} = \text{SBERT}\big(A_{\text{gold}}\big), \quad \boldsymbol{e}_{\text{pred}} = \text{SBERT}\big(A_{\text{pred}}\big) \tag{10}$$

2. Die Cosine Similarity zwischen den beiden Vektoren wird berechnet als:

$$\text{sim}\big(\boldsymbol{e}_{\text{gold}}, \boldsymbol{e}_{\text{pred}}\big) = \frac{\boldsymbol{e}_{\text{gold}} \cdot \boldsymbol{e}_{\text{pred}}}{\big\|\boldsymbol{e}_{\text{gold}}\big\| \big\|\boldsymbol{e}_{\text{pred}}\big\|} \tag{11}$$

   wobei das Skalarprodukt im Zähler und das Produkt der Normen im Nenner steht.

3. Die Antwort gilt als korrekt, wenn

$$\text{sim}\big(\boldsymbol{e}_{\text{gold}}, \boldsymbol{e}_{\text{pred}}\big) \geq 0.7 \tag{12}$$

4. Andernfalls wird sie als falsch klassifiziert.

## 5.3 Begründung des Schwellenwerts 0.7

Der Schwellenwert von 0.7 wurde folgendermaßen bestimmt:

- Einschluss semantischer Äquivalenz: In Testreihen zeigte sich, dass Paraphrasen und Synonyme meist eine Cosine Similarity $\geq 0.7$ erreichen.
- Ausschluss zufälliger Koinzidenzen: Werte deutlich unter 0.7 (z. B. 0.5–0.6) traten bei thematisch verwandten, aber inhaltlich unterschiedlichen Phrasen auf (z. B. „throwing„ vs. „grappling“).
- Abwägung Präzision vs. Recall: Ein höherer Schwellenwert (z. B. 0.9) hätte zu streng agiert und korrekte, aber leicht variierte Formulierungen als falsch gewertet. Ein niedrigerer Schwellenwert (z. B. 0.6) hätte zu viele semantisch entfernte Phrasen als korrekt akzeptiert. Die Wahl von 0.7 balanciert beide Effekte aus und liefert in unseren Validierungssets das beste F1-Ergebnis.

Nichtsdestotrotz führt diese semantische Evaluierungsmethodik in ca. 10–20% der Fälle immer noch zu Fehleinschätzungen, die manuell korrigiert werden müssen.

## 5.4 Hinweise

- Vorverarbeitung: Vor der Embedding-Berechnung wurden die Texte größtenteils normalisiert (Trimmen, Entfernen unnötiger Leerzeichen), um inkonsistente Tokenisierung zu reduzieren.
- Auswertung: Beim Reporting der Ergebnisse wird die Accuracy (Anteil richtig klassifizierter Fragen) berechnet als

$$\text{Accuracy} = \frac{\text{Anzahl der Fragen mit sim} \geq 0.7}{\text{Gesamtanzahl Fragen}} \times 100\% \tag{13}$$

## 5.5 Zusammenfassung

Durch die ausschließliche Verwendung von SAS (Cosine Similarity $\geq 0.7$) erreichen wir:

- Hohe Semantische Robustheit: Erlaubt vielfältige, aber inhaltlich korrekte Antwortvariationen.
- Einfache Implementierung: Nur wenige Zeilen Code und eine einzige externe Abhängigkeit (`sentence-transformers`).

- Stabile Evaluation ohne das Rauschen, das stringbasierte Metriken bei kleinen Änderungen verursachen.
- Klare Entscheidungsbasis durch einen festen Schwellenwert, der in Validierungs-Experimenten empirisch gerechtfertigt wurde.

Dadurch wird gewährleistet, dass unsere Evaluation semantisch besonders robust ist und vielfältige, inhaltlich korrekte Antwortvariationen zulässt, dabei mit minimalem Implementierungsaufwand auskommt, stabil gegenüber kleinen Textänderungen bleibt und dank eines fest definierten, empirisch begründeten Schwellenwerts eine klare und nachvollziehbare Entscheidungsgrundlage bietet.

6

# Analyse falsch beantworteter Fragen

In diesem Kapitel werden systematisch Fragen analysiert, die das QA-System nicht korrekt beantwortet hat. Ziel ist es, zu prüfen, ob die erhaltenen Antworten tatsächlich falsch sind, an welcher Stelle im Kontext das Modell sie gefunden hat und welche Ursachen dafür verantwortlich sein könnten. Anschließend werden mögliche Verbesserungen diskutiert. Dabei liegt der Fokus auf Antworten die zwar inkorrekt, aber plausibel sind. Das hilft dabei die *Gedanken* und Muster zu verstehen nach denen das Modell agiert, bzw. wo es Schwierigkeiten hat.

Dabei orientieren wir uns an der zuvor vorgenommenen Einteilung in easy, medium und hard fragen, inspiriert von (Sun et al., 2023).

## 6.1 Easy-Fragen

Die folgenden Easy-Fragen wurden vom Modell fehlerhaft oder ungenau beantwortet. Da Easy-Fragen grundlegendes Faktenwissen abfragen, bzw. oft mehrmals im Textkorpus vorkommen, ist hier das Erwartungsniveau hoch.

#### *What is the objective of judo?*

- Expected: throw, pin, or submit opponent
- Span: *free practice* (Start: 17829, End: 17842)
- Similarity Score: 24.82

Prüfung der Antwort:

*Free practice* (jp.: randori) ist **nicht** das Ziel eines Kampfes, sondern eher das Ziel einer Trainingseinheit bzw. deren Hauptfokus. Die Frage richtet sich jedoch auf das Ziel eines Wettkampfes. Die Antwort wurde aus der Passage *Kano's emphasis on randori (free practice) in Judo* extrahiert.

Mögliche Ursachen:

Verwechslungsgefahr ähnlicher Phrasen: In der Nähe der Definition des Wettkampf-Ziels steht die Erwähnung des Fokus einer Trainingseinheit.

Verbesserungsmöglichkeit: Präzisierung durch zusätzliche Schlagworte: Frage eventuell als *What is the objective in a judo competition?* oder *How to win a judo match?* formulieren, um klar auf Wettkampfaspekte hinzuweisen.

#### *Who is the person performing the throw?*

- Expected: tori
- Span: *judoka* (Start: 4912, End: 4918)
- Similarity Score: 28.81

Prüfung der Antwort:

*Judoka* ist ein allgemeiner Begriff für Personen, die Judo machen und funktioniert als Oberbegriff. Die exakte Bezeichnung, die in der Frage gewünscht ist, lautet *tori*. Die Antwort ist daher zwar prinzipiell korrekt, aber nicht präzise.

Mögliche Ursachen:

Generalisierung durch das Modell: Häufig spricht man von *Judoka* und seltener von dem speziellieren Begriff *tori*, also de Judoka der die Technik ausführt.

Verbesserungsvorschläge:

- Einführung eines Fachbegriffs-Lexikons: Eine Nachschlage-Liste bereitstellen, die das Modell bei Antworten zwingt, zwischen generischen und spezifischen Termini zu unterscheiden (z. B. *tori* vs. *judoka*).
- Frage umformulieren: Mit *What is the specific Japanese term for the person performing the throw?* das Modell noch stärker auf Fachbegriffe lenken.

Verbesserungsvorschläge:

- Kontextgewinnung verfeinern: Eine semantische Nachbearbeitung einführen, die prüft, ob der gefundene Span überhaupt eine Person bezeichnet. Wörter wie *philosophy* können so automatisch ausgeschlossen werden.
- Regex-Pattern für Personennamen: Antworten, die keine Personennamen oder spezifische Fachbegriffe (hier *uke*) darstellen, sollten verworfen und nach einer neuen Top-Span-Auswahl gesucht werden.

#### *Name a shime-waza technique.* / *Name a kansetsu-waza technique.* / *Name an osaekomi-waza technique.*

Bei diesen drei Fragen kam es zu einem ähnlichen Fehler:
- Korrekte Antworten wären z.B. Juji-jime, Ude-garami, Kesa-gatame
- Erhalten wurden die Antworten *throwing* bzw. *throwing techniques*

Prüfung der Antwort:

Alle drei Fragen verlangen spezifische Techniken aus unterschiedlichen Kategorien: Würgegriffe (shime-waza), Hebelgriffe (kansetsu-waza) und Haltegriffe (osaekomi-waza). Die erhaltene Antwort *throwing* (bzw. *throwing techniques*) bezieht sich jedoch auf *nage-waza* (Wurftechniken) und ist damit falsch.

Mögliche Ursachen:

1. Falsche Kategoriereferenz: Im Korpus gibt es Überschneidungen bei den Domain-Begriffen (*throwing techniques*, *grappling techniques*, *waza*), und das Modell scheint kurzfristig die nächstbeste Technik-Kategorie (*throwing*) ausgewählt zu haben, anstatt die korrekte Unterkategorie abzufragen.
2. Nicht spezifizierte Frageformulierung: Weil die Frage nur *Name a shime-waza technique* lautet, besteht keine implizite Beschränkung auf eine konkrete Liste, und das Modell weicht auf die nächstliegende Kategorie aus, die im Kontext häufiger vorkommt.

Verbesserungsvorschläge:

- Das Hauptproblem bei der Auswertung, ist dass es viele mögliche korrekte Antworten gibt, und selbst eine semantische Evaluierungsmethode wie Cosine Similarity (Abschnitt 5) wahrscheinlich falsch evaluiert. Es wäre daher sinnvoll für zukünftige Iterationen solche Fragen entweder völlig wegzulassen oder eine komplette Liste der möglichen Antworten in dem *answer*-Feld der JSON-Datei abzulegen.

#### *Is judo mixed-sex?*

- Expected: no
- Span: *Mixed-sex* (Start: 59806, End: 59815)
- Similarity Score: 18.51

Prüfung der Antwort:

Die Frage verlangt eine Ja-/Nein-Antwort: Im modernen Wettkampf ist Judo getrennt nach Geschlechtern (Männer- und Frauenwettbewerbe), also *no*. Die Antwort *Mixed-sex* deutet darauf hin, dass das Modell eine generische Aussage über gemischte Trainingsgruppen zurückgegeben hat, aber nicht erkannte dass es sich um eine Ja-/Nein-Antwort handelt.

Mögliche Ursachen:

Question-Answering-Modelle wie Roberta-Bert sind auf Extractive-QA optimiert. Eine Ja-/Nein-Antwort ist daher oft nicht direkt aus dem Textkorpus extrahierbar.

Verbesserungsvorschläge: : Frage offen umformulieren, bzw. geschlossene Fragen weglassen/vermeiden.

#### *What does judogi translate to?*

- Expected: judo attire
- Span: *uniform* (Start: 58252, End: 58259)
- Similarity Score: 45.15

Prüfung der Antwort:

*Uniform* ist im weitesten Sinne korrekt, aber nicht exakt: *judogi* bezeichnet wörtlich *Judo-Bekleidung* bzw. *Judo-Anzug*. Die Antwort *uniform* ist also nicht genau genug, wenn die Begriffsspezifikation gefordert ist.

Mögliche Ursachen:

1. Generalisierung durch das Modell: Bei Übersetzungen wählt das Modell häufig einen allgemeineren Begriff, ähnlich wie bei der Unterscheidung in Frage
2. Kontextdominanz synonym verwendeter Wörter: *Judo uniform* wird oft synonym eingesetzt, sodass das Modell *uniform* extrahiert und *judo* weglässt.

Verbesserung: Ergänzte Frage: *What is the literal translation of 'judogi'?* zielt auf eine wortgetreue Übersetzung ab.

#### *What is the traditional judo attire made of?*

- Expected: strong white cloth
- Span: *kimono* (Start: 100679, End: 100685)
- Similarity Score: 20.56

Prüfung der Antwort:

Ein *kimono* ist ein traditionelles japanisches Gewand, wird aber auch für Judoanzüge verwendet. Die Frage bezieht sich auf das Material, nicht auf ein Synonym oder den Oberbegriff. Die Antwort *kimono* ist naheliegend aber unpräzise, bzw. leicht fehlgeleitet.

Mögliche Ursachen:

Frage nicht ausreichend präzise formuliert. *Stattdessen wäre z.B. What type of fabric is judo attire made of?* Da *traditional* oft mit *kimono* in Verbindung gebracht wird würde es Sinn ergeben dies nicht extra zu erwähnen um das Modell nicht fehlzuleiten.

## 6.2 Medium-Fragen

Die Medium-Fragen stellen ein moderates Anspruchsniveau dar und verlangen oft zusätzliche Einordnung. Nachfolgend die falsch beantworteten Beispiele und ihre Analyse.

#### *What is the category for sacrifice throws?*

- Expected: sutemi-waza
- Span: *nage waza* (Start: 9353, End: 9362)
- Similarity Score: 57.74

Prüfung der Antwort:

*nage waza* (Wurftechniken im Allgemeinen) ist eine Oberkategorie, die *sutemi-waza* (Würfe bei denen man auch selbst fällt) unter sich fasst, aber nicht identisch damit ist. Die Antwort ist deswegen unpräzise.

Mögliche Ursachen:

1. Hierarchie-Verwechslung/ Generalisierung: Das Modell erkennt *waza* im Kontext, wählt jedoch die bekanntere Oberkategorie *nage waza*.

2. Verschiedene Häufigkeit im Text: Im Korpus taucht *sutemi waza* 9 mal auf, *nage waza* hingegen 22 mal, wodurch *nage waza* als statistisch relevanter gilt.

Verbesserungsvorschläge:

- Gezielte Fine-Tuning-Beispiele: QA-Paare, in denen zweimal hintereinander Unterkategorien abgefragt werden, damit das Modell den Unterschied lernt.
- Semantische Constraints: Regeln implementieren, die verhindern, dass eine Oberkategorie akzeptiert wird, wenn eine spezifischere Unterkategorie gesucht ist.

#### *What influenced European and Russian judoka?*

- Expected: their strong wrestling traditions
- Span: *traditional forms of combat* (Start: 7039, End: 7066)
- Similarity Score: 28.51

Prüfung der Antwort:

*Traditional forms of combat* eine etwas weniger präzise, aber durchaus plausible Antwort. Hier zeigt sich demnach nicht die Schwäche des QA-Modells sondern die der Evaluierungsmethodik mit Cosine-Similarity.

#### *Which American judoka is also an MMA fighter?*

- Expected: Ronda Rousey
- Span: *Hidehiko Yoshida* (Start: 133357, End: 133373)
- Similarity Score: 29.70

Prüfung der Antwort:

Hidehiko Yoshida ist ein japanischer Judoka, der auch MMA-Kämpfe bestritt, aber die Frage verlangt explizit nach einem US-Judoka. Ronda Rousey ist korrekt und kommt in Textkorpus 7 mal vor, Hidehiko Yoshida nur 2 mal. Daher ist die falsche Antwort wohl der nicht-deterministischen Natur von LLMs geschuldet.

#### *Name a forbidden sacrifice throw in competition.*

- Expected: Kani basami
- Span: *Finger, toe and ankle locks* (Start: 77790, End: 77817)
- Similarity Score: 5.55

Prüfung der Antwort:

*Finger, toe and ankle locks* sind verboten im Judo, stimmen also thematisch, aber die Frage verlangt einen verbotenen **sacrifice throw**. Die Antwortmethode ist deswegen nicht vollständig falsch, aber inkonsequent zur Kategorie.

Mögliche Ursachen:

- Kategorienverschachtelung: Das Modell hat erkannt, dass *locks* verboten sind, aber nicht unterschieden, ob es sich um Hebel-, Würge- oder Wurftechniken handelt. Bei dieser Frage wird ähnlich wie bei der vorherigen eine Einschränkung ignoriert (z.B. dass es sich hier um sacrifice throws handeln soll).

Verbesserungsvorschläge:

- Spezifische Schlüsselwörter: Frage um *sacrifice throw (sutemi waza)* erweitern, damit das Modell sich auf Wurftechniken fokussiert.

Verbesserungsvorschläge:

- Konsistente Quellenaufbereitung: Vor dem Training oder der Chunk-Selektion sicherstellen, dass jede Technik klar ihrer richtigen Unterkategorie zugeordnet ist.

#### *Which Olympic Games marked judo's competitive transformation?*

- Expected: 1964 Tokyo Olympics
- Span: *Summer Olympic Games* (Start: 230, End: 250)
- Similarity Score: 50.80

Prüfung der Antwort:

*Summer Olympic Games* ist zu allgemein – Judo wurde erstmals 1964 in Tokio zum Medaillenwettbewerb. Die korrekte Antwort muss die spezielle Ausgabe *1964 Tokyo Olympics* nennen.

Mögliche Ursachen:

1. Unklare Abgrenzung der verschiednen Olympischen Spiele: Das Modell hat zwar den Olympischen Kontext erfasst, aber nicht die genaue Jahreszahl.
2. Generalisierung: Bei Fragen nach *which Olympics* tendiert das Modell dazu, auf den Oberbegriff *Summer Olympic Games* zurückzugreifen, anstatt die Jahreszahl/Austragungsort auszuwählen.

Verbesserungsvorschläge:

- Konkretere Frage: *At which Olympic Games did judo become an official medal sport?*

## 6.3 Hard-Fragen

Hard-Fragen erfordern oft sehr spezifisches Fachwissen oder historische Details:

#### *What are the two guiding principles of judo?*

- Expected: Seiryoku-Zen'yō and Jita-Kyōei
- Span: *life, art and science* (Start: 79065, End: 79086)
- Similarity Score: 15.01

Prüfung der Antwort:

*Life, art and science* beschreibt die Philosophie von Judo, aber nicht die beiden Kodokan-Leitsätze. Die Antwort ist daher nicht präzise.

Mögliche Ursachen:

1. Konflikt philosophischer Passagen: Das Modell extrahiert allgemeine Philosophiebeschreibungen, wenn nach Prinzipien gefragt wird.
2. Ungenaue Formulierung der Frage: *Guiding principles* kann auch breit interpretiert werden, aber hier sind spezifische japanische Leitsätze gefordert.

Verbesserungsvorschläge:

- Explizite Begriffsvorgabe: Frage als *What are the two Japanese guiding principles of the Kodokan?* stellen.

## 6.4 Zusammenfassung der Verbesserungsansätze

1. Präzisere Frage-Formulierungen
   - Zusätzliche Schlüsselwörter (z. B. *literal translation*, *in competition*, *in Europe*) helfen dem Modell, die Antwortspan-Auswahl zu fokussieren.

2. Semantische und regelbasierte Nachbearbeitung
   - Filtersysteme für technische Begriffe, Personen-NER und numerische Werte.

- Post-Processing: Auswertung des Antwortspans, um Vollständigkeit und Kategoriezugehörigkeit zu prüfen.

3. Data Augmentation und Fine-Tuning
   - Hinzufügen von QA-Beispielen, die häufige Fehlerfälle adressieren.
   - Nutzung von kontrastiven Beispielen: Positiv- und Negativ-Beispiele einbinden (Few-Shot Learning).

4. Glossarerweiterung
   - Aufbau eines Judo-Wörterbuchs mit Verweisen auf offizielle Begriffe (Techniken, Prinzipien, historische Daten). Das wäre ein strukturierterer Ansatz als der jetzige, wo unterschiedliche Texte einfach konkateniert werden.
   - Nutzung eines externen Knowledge Graphs, um die semantische Validität der extrahierten Antworten zu prüfen. So könnten auch klare Hierarchien zwischen Techniken definiert werden.

Aus den erwähnten Punkten folgt, dass Textkorpus, Frageformulierung und Evaluierungsmethodik alle noch Optimierungspotentiale haben, die das Test-Environment robuster machen können.

7

# Anwendungsfälle von QA-Systemen in der Praxis

In dieser Arbeit wurde der Fokus auf Extractive-Question-Answering-Systeme (QA) gelegt, die sowohl Faktenwissen als auch konzeptuelle Zusammenhänge extrahieren. Im Folgenden werden zentrale Einsatzfelder kurz skizziert.

QA-Systeme mit Fokus auf Fakten- und Konzeptwissen bieten in zahlreichen Domänen hohen Mehrwert. Sie beschleunigen Recherche, verbessern Informationszugang und unterstützen komplexe Entscheidungsprozesse. Zukünftige Fortschritte in semantischen Repräsentationen und multimodaler Integration werden die Einsatzmöglichkeiten weiter ausdehnen.

| Bildung & E-Learning |
| --- |
| Unterstützt interaktives Lernen durch präzise Fakten und kontextuelle Erklärungen zu Lehrtexten und entlastet Lehrkräfte, indem es automatisch Quizfragen generiert und komplexe Inhalte erläutert. |
| Customer Support & Helpdesk |
| Automatisiert die Bearbeitung von FAQs durch Extraktion von Produktinformationen und Prozessschritten aus Dokumentationen und verbessert die Fehlerdiagnose, indem es Fehlermeldungen erkennt und zugrunde liegende Konzepte erklärt. |

| Enterprise Knowledge Management |
| --- |
| Beschleunigt die Dokumentenrecherche durch schnelle Faktenfindung (z. B. Ansprechpartner, Fristen) und semantische Extraktion von Prozessabläufen; liefert im Compliance- und Audit-Kontext zitierfähige Textstellen und erklärt Risiken bei Nichteinhaltung. |
| Medizinische Informationssysteme |
| Beantwortet patientenbezogene Fragen zu Dosierungen und erklärt Krankheitszusammenhänge; unterstützt die Forschung durch Extraktion von Studiendaten und Identifizierung konzeptioneller Hypothesen aus Fachliteratur. |
| Recht & Compliance |
| Ermöglicht juristische Recherchen durch Zitieren relevanter Gesetzesartikel und Erklären rechtlicher Konzepte; vereinfacht die Vertragsanalyse, indem es zentrale Klauseln erkennt und deren juristische Tragweite bewertet. |
| Wissenschaftliche Forschung & Literaturübersicht |
| Extrahiert aus Fachartikeln Stichprobengrößen und konzeptionelle Limitationen; erleichtert interdisziplinäre Recherchen durch Erklärung von Methodenübertragungen und Vergleich zentraler Fachbegriffe. |
| Öffentliche Dienste & Behörden |
| Beantwortet Fragen zu benötigten Unterlagen in Verwaltungsportalen und erläutert rechtliche Grundlagen; liefert in der Krisenkommunikation Notfallinformationen und konzeptionelle Handlungsempfehlungen aus Leitfäden. |
| Digitale Bibliotheken & Archive |
| Unterstützt historische Recherchen durch Auffinden von Datumsangaben und kontextuelle Einordnung von Ereignissen; vereinfacht die multilingualen Dokumentenerschließung durch Übersetzung von Fragestellungen und Extraktion von Fakten sowie Konzepten aus fremdsprachigen Quellen. |

8

# Fazit

Die Studienarbeit untersuchte systematisch die Leistungsfähigkeit von LLM-basierten Question-Answering-Systemen im Bereich faktischen Wissens. Basierend auf einem domänenspezifischen Korpus (Judo) und einem methodisch fundierten Test-Environment wurden folgende Kernaussagen validiert:

- **Begrenzte Faktenzuverlässigkeit**: Selbst moderne LLMs zeigen signifikante Limitationen bei der Beantwortung faktischer Fragen. Die maximale Accuracy von 69,2 % (Voller Kontext) bestätigt frühere Studien wie (Sun et al., 2023) – LLMs sind keine universellen Wissensspeicher.

- **Kontextreduktion als Double-Edged Sword**: Semantisches Chunking steigert die Effizienz (bis zu 68 % kürzere Kontexte), führt aber ab >50 % Reduktion zu drastischem Accuracy-Einbruch (≤50 %). Der Sweet Spot liegt bei 75 % relevanter Chunks (23 % Zeichenreduktion, 62,8 % Accuracy).

- **Evaluierungsmetrik entscheidend**: Herkömmliche String-Metriken (EM, F1) scheitern an semantischen Nuancen. SAS (Cosine Similarity ≥0.7) erwies sich als robuste Alternative zur Bewertung inhaltlicher Korrektheit.

- **Fehlerprofile**: Um eine genauere Evaluation zu ermöglichen wurden Fehlerprofile in Anlehnung definiert (Sun et al., 2023). Dabei kommen in den Kategorien folgende Fehler besonders vor:
  - **Easy-Fragen**: Fehler durch Generalisierung
  - **Medium/Hard-Fragen**: Kategorienverwechslungen und Kontextinkonsistenzen.
  - **Geschlossene Fragen**: (Ja/Nein) bereiten LLMs besondere Schwierigkeiten.

- **LoRA-Fine-Tuning ohne Breakthrough**: In diesem Use Case brachte parameter-effizientes Fine-Tuning keinen Accuracy-Zuwachs – ein Hinweis auf inhärente Wissenslücken, nicht nur Domänenadaptionsbedarf.

9

# Ausblick

Basierend auf den Erkenntnissen ergeben sich folgende Forschungs- und Optimierungsperspektiven:

## 9.1 Methodische Weiterentwicklungen

- **Adaptives Chunking**: Dynamische Chunk-Auswahl basierend auf semantischer Score-Verteilung (z. B. Threshold ≥0.6) statt fixer Reduktionsraten.
- **Hybride Evaluierung**: Kombination von SAS mit regelbasierten Filtern (z. B. Fachbegriffslexika) zur Reduktion von Fehlklassifikationen bei Synonymen. Dies ist auch durch die Angabe von mehreren Musterantworten pro Frage realisierbar.
- **Strukturierte Kontextanreicherung**: Integration von Knowledge Graphs zur expliziten Modellierung von Begriffs-Hierarchien (z. B. sutemi-waza ⊂ nage-waza).

## 9.2 Architekturinnovationen

- **Few-Shot Prompt Engineering**: Kontrastive Beispiele (positiv und negativ) in Prompts, um das Modell auf präzise Begriffe oder die erwartete Antwort-Art zu konditionieren.
- **Multimodale Erweiterung**: Kombination von Text-QA mit visuellen Technik-Diagrammen würde weitere spannende Usecases ermöglichen.

Die Arbeit unterstreicht: LLM-basiertes QA ist ein leistungsfähiges, aber begrenztes Werkzeug. Seine Zuverlässigkeit hängt maßgeblich von präziser Fragenformulierung und einem passend zugeschnittenen Kontext ab – besonders bei faktisch kritischen Anwendungen.

10

# Bibliographie

Clark, C., & Dalan, e. a. (2019). TyDi QA: A Typologically Diverse Question Answering Dataset. *Transactions of the Association for Computational Linguistics*, *7*, 454–470.

De Jong, T., & Ferguson-Hessler, M. G. (1996). Types and qualities of knowledge. *Educational psychologist*, *31*(2), 105–113.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL-HLT 2019*, 4171–4186.

Evidently AI. (2023, ). *Mean Reciprocal Rank (MRR) explained.*

Harvard Business Review. (2023). *How Machine Learning Can Improve the Customer Experience.*

Hu, E., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Chen, L., Liu, W., & Wang, Z. (2021). LoRA: Low-Rank Adaptation of Large Language Models. *arXiv preprint arXiv:2106.09685*.

IBM. (2024, ). *What Is Artificial Intelligence (AI)?*.

Intel Corporation. (2024, ). *Fine-Tune Llama 2 70B on Intel® Gaudi® 2 AI Accelerators*. https://www.intel.com/content/www/us/en/developer/articles/llm/fine-tuning-llama2-70b-and-lora-on-gaudi2.html

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Platt, A., Epstein, M., & Polosukhin, I. (2020). Natural Questions: A Benchmark

for Question Answering in the Real World. *Transactions of the Association for Computational Linguistics*, *8*, 450–466.

Lewis, P., Oguz, B., Rinott, R., Riedel, S., Stoyanov, V., & Zettlemoyer, L. (2020, ). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Proceedings of NeurIPS 2020*.

McKinsey & Company. (2024a). *The State of AI in 2024*.

McKinsey & Company. (2024b). *What Is Generative AI?*.

MIT Sloan Management Review. (2019, ). *Machine learning, explained*.

Powers, D. M. W. (2020, ). *Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation*. https://arxiv.org/abs/2010.16061

*Question Answering with BERT*. (2023, ).

Rajpurkar, P., Jia, R., & Liang, P. (2018). SQuAD 2.0: \textit{The} 2.0 Leading Challenge of Unanswerable Questions. *arXiv preprint arXiv:1806.03822*.

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv preprint arXiv:1606.05250*.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of EMNLP-IJCNLP 2019*, 3982–3992.

Risch, J., Möller, T., Gutsch, J., & Pietsch, M. (2021, ). *Semantic Answer Similarity for Evaluating Question Answering Models*. https://arxiv.org/abs/2108.06130

SAS Communities Library. (2024, ). *Where does GenAI fit within the AI landscape*.

Sun, K., Xu, Y. E., Zha, H., Liu, Y., & Dong, X. L. (2023). Head-to-tail: how knowledgeable are large language models (LLMs)? AKA will LLMs replace knowledge graphs?. *arXiv preprint arXiv:2308.10168*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. *Advances in Neural Information Processing Systems (NeurIPS) 30*, 5998–6008.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., & Brew, J. (2020). Transformers: State-of-the-Art Natural Language Processing. *arXiv preprint arXiv:1910.03771*.

11

# Anhang

## 11.1 Easy-Fragen

| Frage | Antwort |
|---|---|
| What does judo mean? | gentle way |
| Who founded judo? | Kanō Jigorō |
| What is the name of the school Kanō Jigorō established? | Kōdōkan |
| What is the judo uniform called? | jūdōgi |
| What color belt do novices wear? | white |
| What color belt do masters wear? | black |
| What is the term for judo students? | jūdōka |
| What is the term for free practice in judo? | randori |
| What is the objective of judo? | throw, pin, or submit opponent |
| What does ippon mean? | one point |
| What does waza-ari mean? | half point |
| What is a minor point called? | yuko |
| What is the category for standing techniques? | tachi-waza |
| What is the category for ground techniques? | ne-waza |

| Frage | Antwort |
|---|---|
| Who is the person performing the throw? | tori |
| Who is the person receiving the throw? | uke |
| Name a shime-waza technique. | Nami-juji-jime |
| Name a kansetsu-waza technique. | Ude-garami |
| Name an osaekomi-waza technique. | Kesa-gatame |
| What type of contact is judo? | full contact |
| Is judo mixed-sex? | no |
| What is the focus of judo? | throwing |
| What does judogi translate to? | judo attire |
| What does judoka mean? | judo performer |
| What does nage-waza include? | throwing techniques |
| What is the governing body of international judo? | International Judo Federation |
| What do judoka wear on their feet during practice? | bare feet |
| What is the traditional judo attire made of? | strong white cloth |

## 11.2 Medium-Fragen

| Frage | Antwort |
|---|---|
| What is the term for pre-arranged forms in judo? | kata |
| From which martial art did judo originate? | jujitsu |
| What is the Japanese term for throwing techniques? | nage-waza |
| What is the category for sacrifice throws? | sutemi-waza |
| What is the category for hip throwing techniques? | koshi-waza |
| What is the category for foot throwing techniques? | ashi-waza |
| What is the maximum dan rank in judo? | 10th dan |
| What is the symbol at the center of the Kodokan emblem? | red circle |

| Frage | Antwort |
|---|---|
| What black belt rank is shodan? | first rank |
| Name a Kodokan kata. | Ju-no-kata |
| What does katame-waza include? | grappling techniques |
| What did Kano eliminate from his art? | the most dangerous techniques |
| What did Kano preserve in kata? | classical techniques of jujitsu |
| What influenced European and Russian judoka? | their strong wrestling traditions |
| What Russian art was based on judo? | sambo |
| Which American judoka is also an MMA fighter? | Ronda Rousey |
| How many national federations does the IJF bring together? | more than 200 |
| How many continental unions does the IJF have? | 5 |
| Where is the International Judo Federation headquartered? | Budapest, Hungary |

## 11.3 Hard-Fragen

| Frage | Antwort |
|---|---|
| In what year was judo founded? | 1882 |
| What is the Japanese term for grappling techniques? | katame-waza |
| What is the Japanese term for body-striking techniques? | atemi-waza |
| What is the Japanese term for blocks and parries? | uke-waza |
| What is the Japanese term for resuscitation techniques? | kappo |
| How many throws are in the Kodokan Gokyo-no-waza? | 67 |
| When did men's judo first appear at the Olympics? | 1964 |

| Frage | Antwort |
|---|---|
| When did women's judo first appear at the Olympics? | 1992 |
| In what year did the first women's World Judo Championships take place? | 1980 |
| In what year did women's judo debut as a demonstration sport at the Olympics? | 1988 |
| What are the two guiding principles of judo? | Seiryoku-Zen'yō and Jita-Kyōei |
| What does Seiryoku-Zen'yō mean? | maximum efficient use of energy |
| What does Jita-Kyōei mean? | mutual welfare and benefit |
| In what year was the International Judo Federation founded? | 1951 |
| Who was the first president of the International Judo Federation? | Aldo Torti |
| Who succeeded Aldo Torti as IJF president? | Risei Kano |
| What shape is the Kodokan emblem? | octagonal mirror |
| Which two students received the first ever shodan ranks? | Tsunejirō Tomita and Shiro Saigo |
| Name a forbidden sacrifice throw in competition. | Kani basami |
| Name a prohibited katame-waza technique. | Do-jime |
| Name a non-Kodokan Japanese kata. | Go-no-sen-no-kata |
| In what year did Kano open a women's section at the Kodokan? | 1923 |
| Who dedicated her life to spreading women's judo? | Keiko Fukuda |
| In what year was the European Judo Union first created? | 1932 |
| Where were the first European Judo Championships held? | Dresden |
| What does uke-waza include? | blocks and parries |

| Frage | Antwort |
|---|---|
| What does atemi-waza include? | body-striking techniques |
| What did Kano stress in practice? | randori |
| What English phrase describes ju yoku go o seisu? | softness controls hardness |
| What was judo's inclusion status for the 1940 Tokyo Olympics? | demonstration sport |
| Which Olympic Games marked judo's competitive transformation? | 1964 Tokyo Olympics |

# A Abkürzungen

| | |
|---|---|
| **API** | Application Programming Interface |
| **EM** | Exact Match |
| **GPU** | Graphics Processing Unit |
| **HTTP** | Hypertext Transfer Protocol |
| **KI** | Künstliche Intelligenz |
| **LLM** | Large Language Model |
| **LoRA** | Low-Rank Adaption |
| **NLP** | Natural Language Processing |
| **QA** | Question Answering |
| **REST** | Representational State Transfer |
| **SAS** | Semantic Answer Similarity |
| **SQuAD** | Stanford Question Answering Dataset |

# Selbstständigkeitserklärung

Gemäß Ziffer 1.1.13 der Anlage 1 zu §§ 3, 4 und 5 der Studien- und Prüfungsordnung für die Bachelorstudiengänge im Studienbereich Technik der Dualen Hochschule Baden- Württemberg vom 29.09.2017. Ich versichere hiermit, dass ich meine Arbeit mit dem Thema:

**Evaluierung von LLM-basiertem QA**

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass alle eingereichten Fassungen übereinstimmen.

Stuttgart, 12.06.2025

Anton Seitz