Empirische Analyse von RAG Evaluation Tools für Betriebliche Abläufe

Bachelorarbeit zur Erlangung des akademischen Grades
Bachelor of Arts/Engineering/Science
im Studiengang <Name des Studiengangs>
an der Fakultät für Informatik und Ingenieurwissenschaften
der Technischen Hochschule Köln

vorgelegt von:	Leon Alexander Bartz
Matrikel-Nr.:	1114236017
Adresse:	Richard-Wagner-Straße 47 50679 Köln leon_alexander.bartz@smail.th-koeln.de
eingereicht bei:	Prof. Dr. Boris Naujoks
Zweitgutachter*in:	Prof. Dr. Dietlind Zühlke

Ort, TT.MM.JJJJ

Kurzfassung/*Abstract*

Eine Kurzfassung (wenn verlangt) in Deutsch und/oder in Englisch (*Abstract*) umfasst auf etwa 1/2 bis 1 Seite die Darstellung der Problemstellung, der angewandten Methode(n) und des wichtigsten Ergebnisses.

Wie man ein gelungenes Abstract verfasst, erfahren Sie in den Seminaren oder der Beratung des Schreibzentrums der Kompetenzwerkstatt[1].

Schlagwörter/Schlüsselwörter: evtl. Angabe von 3 bis 10 Schlagwörtern.

[1] https://www.th-koeln.de/schreibzentrum

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

1 Einleitung

1.1 Was ist ein RAG

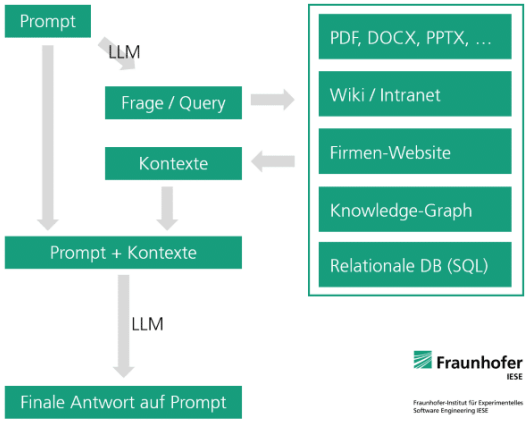

Abbildung 1.1: Stuktur eines RAGS, Quelle: [1]

> Bei Retrieval Augmented Generation (RAG) erweitert man den Prompt für das Large Language Model um Suchergebnisse aus einer Dokumentensammlung, einer Datenbank, einem Wissensgraph (Knowledge Graph) oder einer anderen Suche (z.B. Internetsuche). Das Wissen für die Antwort kommt also aus angebundenen Quellen und nicht aus dem LLM.

[1]

Die nutzung eines RAGs ist eine der beiden Möglichkeiten, um ein LLM zu verbessern. Die andere Möglichkeit ist das Fine-Tuning des LLMs, es gibt jedoch wichtige Unterscheide zwischen diesen beiden Methoden.

Es gibt einige Faktoren welche die Entscheidung beeinflussen können, ob ein RAG oder ein Fine-Tuning besser für den betrieblichen Ablauf geeignet ist.

1.1.1 Kompetenz

Beim Fine-Tuning ist ein gewisses technisches Wissen notwendig, um die Themen Natural Language Processing (NLP), Deep Learning, Modellkonfiguration, Datenaufbereitung und Evaluierung an zu wenden. Der gesammte Prozess des Fine-Tunings ist sehr komplex und erfordert viel Zeit und Ressourcen.

1.1.2 Art der Daten

Sollten die Daten Dynamisch sein, ist das RAG die bessere Lösung, da es die Daten schneller und kontinuierlich aktualisieren kann. Der Prozess des Fine-Tunings erstellt immer eine Snapshot der ein erneutes Training erfordert. Beim Fine-tuning ist es möglich, dass das Modell Muster erkennt und Firmen eigene Begriffe verstehen kann.

1.1.3 Budget

Das Fine-Tuning benötigt tuere Rechenzeit auf hochleistungs GPUs, das macht das training eines Modells sehr teuer. Das RAG hat dagegen zusätliche Kosten die durch das Speichern der Daten in einer Vektordatenbank entstehen.

1.2 Objektive Beurteilung von RAGs

Desto mehr Daten einem RAG zur Verfügung stehen, desto aufwendiger ist es die Qualität des RAGs zu beurteilen. Eine beurteilung durch menschen müsste bei Anpassungen am RAG oder Änderungen an den Daten immer wieder neu durchgeführt werden. Menschliche Beurteilungen sind teurer, daher ist es ergibt es Sinn, mithilfe von LLMs die Beurteilung zu automatisieren. Es gibt bereits tools wie RAGAs die versuchen diesen Prozess unter anderem mithilfe von LLMs zu automatisieren. Diese Tools generieren aus den ihnen gegebenen Daten Fragebögen, die auf eine Frage eine beispielhafte Antwort und die genutzten Stellen aus den vorher gegebenen Dokumenten beinhalten. Sollten nach diesem autmatisierten Test die gewünschten Ergebnisse nicht erreicht werden können zum Beispiel Veröffentlichung blockiert werden.

Sowohl menschlich Bewertungen als auch die reine subjektive Bewertung durch LLMs sind nicht objektiv. Mithilfe von mehreren Techniken kann versucht werden die Bewertung mithilfe von LLMs objektiv zu machen.

1.3 Darstellung des Themas und der Forschungsfragen

In dieser Bachelorarbeit wird untersucht, wie gut diese Tools sowohl subjektive als auch objektive Bewertungen durchführen können. Im Mitelpunkt werden die beiden Tools RAGAS und Gistkard stehen, welche die Bewertung durchführen.

1.4 Praxistauglichkeit und Herausforderungen

Es stellen sich mehrere Herausforderungen für die Bewertung von RAGs durch diese Tools. Der erste ist die die Kosten die bei der Bewertung entstehen, für die Bewertung muss das neue System welches getestet werden aufgesetzt werden. Dies beinhaltet eine eventuelle doppelte Speicherung der Daten und die für das Testen benötigten Aufrufe des LLMs. Neben den Kosten ist auch die Zeit welche es dauert die Bewertung durch zu führen relevant, da die Bewertung schneller durchgeführt werden kann, wenn mehr Ressourcen zur Verfügung stehen. Das System muss auch auf dem neusten Stand gehalten werden, da sich diese noch realtiv junge Thema schnell entwickelt.

Content filtering

1.5 Softwaretechnische Fragestellungen

Fehler beim generieren https://pixion.co/blog/rag-in-practice-test-set-generation

LLM positional bias https://arxiv.org/pdf/2305.17926

Pitfalls in LLM Assisted Evaluation https://medium.aiplanet.com/evaluate-rag-pipeline-using-ragas-fbdd8dd466c1

Rate Limits

Sind RAGS bald tod? https://x.com/agishaun/status/1758561862764122191 https://x.com/ptsi/status/1758511315646320920

2 Metrics

In diesem Kapitel geht es um die verschiedenen Metriken, die für die Bewertung von RAG Evaluations Tools verwendet werden können. Metriken sind das Herzstück der Bewertung von RAGs, da sie die Qualität des RAGs bewerten und somit die Entwicklung und den Fortschritt des RAGs messen.

Diese Metriken basieren auf Faktenextraktion, mithilfe welcher sich dann Scores berechnen lassen. Für die Extraktion der Fakten wird häufig ein LLM verwendet, welcher als Richter fungiert.

2.1 Retrieval Augmented Generation

Diese Metriken basieren auf Faktenextraktion mithilfe welcher sich dann Scores berechnen lassen. Für die extraktion der Fakten wird häufig ein LLM verwendet welcher als Richter fungiert.

2.1.1 Context Precision

> Die Kontextpräzision ist eine Metrik, die den Anteil relevanter Textabschnitte in den abgerufenen Kontexten misst. Sie wird als Mittelwert der Präzision@k für jeden Textabschnitt im Kontext berechnet. Die Präzision@k ist das Verhältnis der Anzahl relevanter Textabschnitte auf Rang k zur Gesamtanzahl der Textabschnitte auf Rang k. (eigene Übersetzung nach [3])

Diese Metrik ist für uns als Qualitätskontrolle wichtig, da sie uns sagt, ob es Probleme beim Testen mit dem Vectorstore gibt.

Wenn es einen guten Context Precision Score gibt, dann lässt sich hier gut bewerten, ob das LLM in der Lage ist, die relevanten Informationen in dem Kontext zu finden. Da dies ein wichtiger Aspekt eines guten RAGs ist, wird diese Metrik im Rahmen dieser Arbeit betrachtet.

2.1.2 Context Recall

> Context Recall misst, wie viele der relevanten Dokumente (oder Informationsstücke) erfolgreich abgerufen wurden. Es konzentriert sich darauf, keine wichtigen Ergebnisse zu verpassen. Ein höherer Recall bedeutet, dass weniger relevante Dokumente ausgelassen wurden. Kurz gesagt geht es beim Recall darum, nichts Wichtiges zu übersehen. (eigene Übersetzung nach [4])

Wenn es eine gute Context Precision Score gibt dann lässt sich hier gut bewerten ob das LLM in der Lage ist die relevanten Informationen in dem kontext zu finden. Da dies ein wichtiger Aspekt eines guten RAGs ist wird diese Metrik im Rahmen dieser Arbeit betrachtet.

2.1.3 Context Entities Recall

In diesem Kontext ist eine Entity eine Informationseinheit, die im Kontext vorkommt. Dies könnte z.B. ein Name, ein Ort, ein Datum oder eine andere Informationseinheit sein.

> Die ContextEntityRecall-Metrik misst den Recall des abgerufenen Kontexts, basierend auf der Anzahl der Entitäten, die sowohl in der Referenz als auch im abgerufenen Kontext vorkommen, relativ zur Gesamtanzahl der Entitäten in der Referenz.
> Einfach ausgedrückt misst sie, welcher Anteil der Entitäten aus der Referenz im abgerufenen Kontext wiedergefunden wird.
> (eigene Übersetzung nach [2])

Diese Metrik ist für uns als Qualitätskontrolle wichtig da sie uns sagt, ob es Probleme beim Testen mit dem Vectorstore gibt.

2.1.4 Noise Sensitivity

> NoiseSensitivity misst, wie häufig ein System Fehler macht, indem es falsche Antworten gibt, wenn entweder relevante oder irrelevante abgerufene Dokumente verwendet werden.
> Um die Noise Sensitivity zu bestimmen, wird jede Aussage in der generierten Antwort daraufhin überprüft, ob sie auf der Grundlage der Referenz korrekt ist und ob sie dem relevanten (oder irrelevanten) abgerufenen

> Kontext zugeordnet werden kann.
> (eigene Übersetzung nach [6])

Diese Metrik ist eine der wichtigsten Metriken in dieser Arbeit da sie die Richtigkeit der Antworten und damit die Qualität des RAGs bewertet.

2.1.5 Response Relevancy

> Die ResponseRelevancy-Metrik misst, wie relevant eine Antwort im Bezug auf die Nutzereingabe ist. Höhere Werte zeigen eine bessere Übereinstimmung mit der Nutzereingabe an, während niedrigere Werte vergeben werden, wenn die Antwort unvollständig ist oder redundante Informationen enthält.
> (eigene Übersetzung nach [8])

Diese Metrik bildet mit der Noise Sensitivity eine wichtige Grundlage für die Bewertung des RAGs. Denn selbst wenn die Antworten richtig sind, ist die Bewertung des RAGs nicht gut, wenn die Antworten nicht relevant zu der Frage sind.

2.1.6 Faithfulness

> Die Faithfulness-Metrik misst, wie faktentreu eine Antwort im Vergleich zum abgerufenen Kontext ist.
>
> Eine Antwort gilt als faktentreu, wenn alle ihre Aussagen durch den abgerufenen Kontext gestützt werden können.
>
> Die Berechnung erfolgt nach folgender Formel:

$$\text{Faithfulness Score} = \frac{\begin{array}{c}\text{Anzahl der durch den Kontext} \\ \text{gestützten Aussagen in der Antwort}\end{array}}{\begin{array}{c}\text{Gesamtanzahl der Aussagen} \\ \text{in der Antwort}\end{array}} \tag{2.1}$$

> (eigene Übersetzung nach [5])

2.1.7 Multimodal Faithfulness/Multimodal Relevance

Da sich diese Metriken mit mehr als textuellen Daten befassen, werden diese nicht im Rahmen dieser Arbeit betrachtet.

2.2 Nvidia Metrics

Diese Metriken sind subjektiver Art und benutzten wieder eine LLM um die Bewertung zu treffen. Hier werden einzelnde Scores generiert welche keinen tieferen Einblick in die Bewertung gewähren.

2.2.1 Answer Accuracy

> Answer Accuracy misst die Übereinstimmung zwischen der Antwort eines Modells und einer Referenz (Ground Truth) für eine gegebene Frage. Dies geschieht über zwei verschiedene "LLM-as-a-judgePrompts, die jeweils eine Bewertung (0, 2 oder 4) zurückgeben. Die Metrik wandelt diese Bewertungen in eine Skala von [0,1] um und nimmt dann den Durchschnitt der beiden Bewertungen der Richter.
> (eigene Übersetzung nach [7])

Das LLM bewertet die Antwort mit der Referenz und auch die Referenz mit der Antwort. Hat Vorteile gegenüber der Answer Correctness, da es weniger Aufrufe mit weniger Tokens an LLM braucht. Es werden im Vergleich zur Answer Correctness auch robustere Bewertungen getroffen, bietet jedoch weniger Einblicke in die Bewertung. Diese Metrik wird im Rahmen dieser Arbeit betrachtet auch um einen Vergleich zu anderen Metriken zu haben.

2.2.2 Context Relevance

Diese Metrik ist sehr ähnlich zur Context Precision, als Alternative und um einen Vergleich zu haben wird diese im Rahmen dieser Arbeit betrachtet, auch wenn sie keine direkte Aussage über das zu bewertende LLM macht.

2.2.3 Response Groundedness

Wenn die Answer Accuracy eine gute Bewertung liefert, ist die Response Groundedness eine gute Bewertung für die Faktualität der Antwort. Diese Logik ist ähnlich zur Kombinierung von Context Relevancy und Context Precision. Hier wird es in den Experimenten interessant zu vergleichen wie diese Metriken zusammenhängen.

2.3 Natural Language Comparison

2.3.1 Factual Correctness

Diese Metriken basieren zu Teilen auf der Wahrheitsmatrix (Confusion matrix), welche die vier Kategorien True Positive, False Positive, False Negative und True Negative definiert.[9] Aus dieser Matrix lassen sich dann precision, recall und f1 score berechnen.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{2.2}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2.3}$$

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.4}$$

[9]

2.3.2 Semantic Similarity

This metric uses embeddings to calculate the semantic similarity between the answer and the reference. TODO: should this be used?

2.4 Non LLM String Similarity

Wie der Name schon sagt, wird die String Similarity ohne LLM berechnet. Diese Metriken sind relative einfache Metriken und werden im Rahmen dieser Arbeit keine große Rolle spielen, jedoch als Vergleich zu anderen Metriken dienen.

2.4.1 BLEU Score

Misst die Ähnlichkeit zwischen der Antwort und der Referenz. Dabei wird die Wortanzahl der Referenz berücksichtigt und eine entsrechende Bestrafung für zu kurze Antworten eingeführt.

2.4.2 ROUGE Score

Mithilfe von n-gram recall, precision, und dem F1 score wird die Ähnlichkeit zwischen der Antwort und der Referenz berechnet.

2.4.3 String Presence

Eine einfache Metrik um zu sehen, ob die Referenz in der Antwort enthalten ist.

2.4.4 Exact Match

Eine noch einfachere Metrik, die nur prüft ob die Antwort exakt der Referenz entspricht. Diese ist für einzelde Wörter sinnvoll.

2.5 General purpose

Dies sind Metriken, welche manuell konfiguriert werden müssen, aber eine gute Bewertung der Qualität eines RAGs liefern können. Die Metriken reichen von einfachen Fragen, wie ïst die Antwort schädlich oder hat die Intention des Users verletzt", bis hin zu komplexeren, einleitend definierten Scores.

- Aspect critic
- Simple Criteria Scoring
- Rubrics based Scoring
- Instance Specific Rubrics Scoring

2.6 Andere Metriken

2.6.1 Summarization

Anzahl der richtig beantworteten Fragen geteilt durch die Anzahl der Fragen. Dies ist eine sehr einfache und oberflächliche Metrik.

2.7 Irrelevante Metriken

2.7.1 SQL

SQL spezifische Metriken welche nicht im Rahmen dieser Arbeit betrachtet werden.

2.7.2 Agents or Tool use cases

Metriken zum Bewerten des Einsatzes von Agents oder Tools, dies liegt ebenso außerhalb des Themas dieser Arbeit. https://docs.ragas.io/en/stable/concepts/metrics/

Diese Metrik wird Teil dieser Arbeit sein, da sie in gewissen Nutzungsfällen, wie z.B. stark faktuale Fragen, eine gute Bewertung liefern kann.

Literatur

[1] Thorsten Honroth, Julien Siebert und Patricia Kelbert. *Retrieval Augmented Generation (RAG): Chatten mit den eigenen Daten*. Zugriff am 7. Februar 2025. Mai 2024. URL: https://www.iese.fraunhofer.de/blog/retrieval-augmented-generation-rag/.

[2] Ragas. *Context Entities Recall*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/context_entities_recall/.

[3] Ragas. *Context Precision*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/context_precision/.

[4] Ragas. *Context Recall*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/context_recall/.

[5] Ragas. *Faithfulness*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/faithfulness/.

[6] Ragas. *Noise Sensitivity*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/noise_sensitivity/.

[7] Ragas. *Nvidia Metrics*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/nvidia_metrics/.

[8] Ragas. *Response Relevancy*. Accessed: 2024. 2024. URL: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/answer_relevance/.

[9] Wikipedia. *Confusion matrix*. Accessed: 2024. 2024. URL: https://en.wikipedia.org/wiki/Confusion_matrix.

Anhang

Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer oder der Verfasserin/des Verfassers selbst entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Anmerkung: In einigen Studiengängen findet sich die Erklärung unmittelbar hinter dem Deckblatt der Arbeit.

Ort, Datum	Unterschrift