# Optimierung der Datenanalyse von deutschen Gerichtsurteilen durch RAG mit LLMs

Exposé zur Bachelorarbeit

Paul Permantier

5. Dezember 2024

# Inhaltsverzeichnis

˘

# 1 Einleitung

Large Language Models (LLMs) haben in den letzten Jahren verstärkt an Bedeutung gewonnen und finden in vielen Bereichen der Gesellschaft Anwendung [27]. Sie zeichnen sich durch ihre Fähigkeit aus, große Mengen an Textdaten zu verarbeiten und implizit Wissen in ihren Parametern zu speichern [24]. Dies erlaubt es ihnen, umfangreiche Texte zu interpretieren und komplexe Zusammenhänge zu erfassen, ohne auf externe Wissensquellen angewiesen zu sein [27]. Das befähigt sie juristischen Aufgaben zu bewältigen [2].

Trotz ihrer beeindruckenden Fähigkeiten weisen LLMs jedoch auch deutliche Schwächen auf [14]. Eine ihrer größten Einschränkungen ist die Unfähigkeit, ihr internes Wissen zu aktualisieren oder zu revidieren. Dies führt dazu, dass LLMs nicht in der Lage sind, sich auf dynamisch verändernde Wissensbestände einzustellen, was in Bereichen wie dem Recht oder der Wissenschaft, in denen sich Informationen ständig weiterentwickeln, problematisch ist. Darüber hinaus neigen LLMs dazu, sogenannte Halluzinationen zu erzeugen, also falsche Informationen, die das Modell überzeugend präsentiert, obwohl sie in der Realität nicht zutreffen Diese Eigenschaften schränken die Einsatzmöglichkeiten von LLMs erheblich ein [18]. Insbesondere in Bereichen, die ein hohes Maß an Genauigkeit und Zuverlässigkeit erfordern. Finetuning auf spezifischen Daten erhöht dabei nicht zwangsläufig ihre Fähigkeit neues Wissen zu inkorporieren [9].

Um diese Herausforderungen zu bewältigen, bietet der Ansatz der Retrieval-Augmented Generation (RAG) eine vielversprechende Lösung [15]. RAG kombiniert ein LLM mit einem externen Retrieval-System, das es dem Modell ermöglicht, während der Textgenerierung auf aktuelle und verlässliche Informationen zuzugreifen. Durch diese Integration wird das Modell in die Lage versetzt, Wissen, das nicht in seinen Parametern gespeichert ist, direkt aus externen Quellen zu beziehen. Dies minimiert das Risiko von Halluzinationen und erhöht die Flexibilität und Aktualität des Modells, ohne dass ein erneutes Training erforderlich ist [15, 1].

Ein besonders komplexes Anwendungsfeld für LLMs ist die Analyse von Gerichtsurteilen. Gerichtsurteile sind hochspezialisierte, rechtliche Dokumente, deren präzise Interpretation nicht nur tiefes Fachwissen erfordert, sondern auch ein hohes Maß an Aktualität [11].

Die Anwendung von RAG im Bereich der juristischen Textanalyse, insbesondere bei der Bearbeitung von Gerichtsurteilen, könnte zahlreiche aktuelle Herausforderungen lösen. Beispiele hierfür sind die Vorhersage von Rechtsreferenzen und die Fallvorhersage. Durch den Einsatz eines RAG-Systems könnte sichergestellt werden, dass das Modell nicht nur das in seinen Parametern gespeicherte Wissen nutzt, sondern aktiv nach den relevantesten und aktuellsten Informationen sucht [15]. Außerdem könnte es Halluzinationen von LLMs vorbeugen, die schon heute fälschlicherweise von Anwälten in Gerichtsprozessen als Beweise vorgebracht werden [23]. Dies könnte eine präzisere und zuverlässigere Analyse von Gerichtsurteilen, die den aktuellen rechtlichen Rahmenbedingungen entspricht, ermöglichen.

Diese Entwicklung besitzt sowohl wissenschaftliche als auch gesellschaftliche Relevanz. Die automatisierte Analyse von Gerichtsurteilen könnte die Effizienz und Geschwindigkeit rechtlicher Prozesse erheblich verbessern. So forscht zum Beispiel das Landesjustizministerium von Baden-Württemberg gerade daran wie KI Richtern bei der Datenverarbeitung helfen kann [12].

Die Umsetzung dieses Ansatzes erfordert die Verwendung eines spezifischen Datensatzes von deutschen Gerichtsurteilen. Über 250.000 Gerichtsurteile werden gratis von OpenLegalData zur Verfügung gestellt [19]. Die Aufbereitung der Daten stellt jedoch eine erhebliche Herausforderung dar. Gerichtsurteile variieren stark in ihrer Struktur, Länge und Komplexität, abhängig von der jeweiligen Instanz und dem Rechtsgebiet. Die Heterogenität dieser Daten erschwert die automatisierte Analyse und erfordert eine sorgfältige Vorverarbeitung. Zudem muss geprüft werden, ob ergänzende Datensätze einbezogen werden sollten, wie zum Beispiel Gesetzestexte, um präzisere Ergebnisse zu gewährleisten.

# 2 Hintergrund

## 2.1 Large Language Models

Große Sprachmodelle (LLMs): LLMs werden auf riesigen Textkorpora trainiert und besitzen mehrere Milliarden (oder mehr) Parameter, wie beispielsweise GPT-3 [3], GPT-4 [22], PaLM [4] und LLaMA [26]. Das Ziel dieser Modelle besteht darin, Maschinen in die Lage zu versetzen, menschliche Befehle zu verstehen und menschlichen Werten zu folgen. Ein besonderes Merkmal von LLMs ist der zweistufige Ansatz: Zunächst erfolgt das Training auf einem umfangreichen, allgemeinen Korpus, gefolgt von einer Ausrichtung auf menschliche Werte. Durch den erheblichen Anstieg der Modellgröße, des Datenvolumens und der Rechenleistung haben sich die Fähigkeiten der LLMs im Vergleich zu früheren Modellen deutlich verbessert, und sie zeigen Fähigkeiten, die in kleineren Modellen nicht vorhanden sind [27].

## 2.2 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) ist ein Modellansatz, der vortrainierte Sprachmodelle mit einem externen Wissensspeicher kombiniert, um die Genauigkeit und Faktentreue bei der Sprachgenerierung zu verbessern. Während herkömmliche Sprachmodelle allein auf ihrem internen Wissen basieren, greift RAG zusätzlich auf externe Informationen zu, was zu präziseren Ergebnissen führt. Diese Technik ermöglicht es, aktuelle und relevante Informationen abzurufen und in den Generierungsprozess zu integrieren, wodurch RAG-Modelle insbesondere bei wissensintensiven Aufgaben überlegen sind [15]. Dabei wird je nach Implementierung zwischen Naive-RAG, Advanced-RAG und Modular-RAG unterschieden [8].

### 2.2.1 Naive-RAG

Die naive Retrieval-Augmented Generation (RAG) folgt dem Retrieve-Read Ansatz [17]: Rohdaten werden in Segmente unterteilt und in einer Vektordatenbank als Plain-Text gespeichert, um Ähnlichkeitssuchen zu erleichtern. Bei einer Anfrage werden die relevantesten Segmente als Kontext in den Prompt integriert, den ein Sprachmodell verarbeitet.
Schwächen dieser Methode liegen in der Präzision des Abrufs und der Gefahr, dass das Modell irrelevante oder inkohärente Inhalte erzeugt, oft ohne neuen Mehrwert hinzuzufügen [8].

### 2.2.2 Advanced-RAG

Advanced RAG verbessert die Retrieval-Augmented Generation durch Optimierungen vor und nach dem Abruf, um die Grenzen der naiven RAG zu überwinden.
Im Vorabruf-Prozess wird die Indexstruktur optimiert, die Anfrage verfeinert und die Qualität der Inhalte durch Techniken wie Segmentierung und Metadatennutzung verbessert. Die Anfrage wird oft umgeschrieben oder erweitert, um präzisere Abrufe zu ermöglichen [17, 28].
Im Nachabruf-Prozess werden die abgerufenen Segmente neu angeordnet und komprimiert, um den wichtigsten Kontext hervorzuheben und Überladung zu vermeiden. So wird eine klare, relevante Antwort unterstützt, indem der Prompt auf wesentliche Informationen fokussiert bleibt [8].

### 2.2.3 Modular-RAG

Die modulare RAG-Architektur erweitert Naive und Advanced RAG durch flexible Module und neue Muster, die an spezifische Herausforderungen angepasst werden können. Neue Module wie Such- und Memory-Module sowie flexible Abrufstrategien verbessern die Abrufgenauigkeit und passen RAG an verschiedene Aufgaben an. Diese Architektur erlaubt zudem die Einbindung von Fine-Tuning und Reinforcement Learning, um die Leistung weiter zu optimieren [7].

## 2.3 OpenLegalData Datensatz

OPENLEGALDATA.IO ist eine kostenlose und offene Plattform, die rechtliche Dokumente und Informationen der Öffentlichkeit zugänglich macht. Ihr Ziel ist es, die Transparenz des Rechtssystems mithilfe von offenen Daten zu verbessern und Menschen ohne juristische Ausbildung dabei zu unterstützen, das Justizsystem besser zu verstehen [21]. Ihr Datensatz ist über ein API frei auf GitHub zugänglich und umfasst über 250.000 deutsche Gerichtsurteile und über 50.000 Gesetze [20]. Die Gesetze bestehen aus dem Gesetzestext und zusätzlich wird auf Gerichtsurteilen verwiesen von denen sie referenziert wurden. Die Gerichtsurteile bestehen üblicherweise aus einem Tenor, einem Tatbestand und einem Abschnitt über die Entscheidungsgründe.

# 3 Arbeitsverlauf

Der Arbeitsverlauf für dieses Projekt gliedert sich in mehrere aufeinanderfolgende Schritte. Zunächst wird eine umfassende Literaturrecherche zu Retrieval-Augmented Generation (RAG) sowie zur Verarbeitung juristischer Texte durchgeführt. Dabei liegt der Schwerpunkt auf dem Einsatz von RAG-Modellen in wissensintensiven Bereichen und deren potenzieller Anwendung im juristischen Sektor.
Im nächsten Schritt wird der Datensatz von OpenLegalData aufbereitet und analysiert, wobei sowohl die Qualität der Daten als auch deren Struktur und Formatierung im Vordergrund stehen. Aufgrund der heterogenen Struktur der Daten werden nur die relevantesten der 250.000 Gerichtsurteile ausgewählt. "Relevant"bedeutet in diesem Kontext, dass Gerichtsurteile ausgewählt werden, die über einen ausformulierten Tatbestand verfügen und ausreichend Rechtsreferenzen enthalten. Diese Auswahl wird durch ein Preprocessing-Verfahren getroffen, das sämtliche Daten durchläuft und die nutzbaren Urteile filtert.
Anschließend erfolgt die Implementierung und Feinabstimmung eines RAG-Modells. Dabei werden sukzessive die Naive-RAG-, Advanced-RAG- und Modular-RAG-Paradigmen eingesetzt [8]. Hierfür kommt ein Technologie-Stack bestehend aus Python, OpenAI-Modellen und Llama-Modellen in Kombination mit Langchain [13] für die Modellintegration sowie Pinecone für den Abrufmechanismus und die Datenspeicherung zum Einsatz. Die Chunk-Größe wird dabei als Hyperparameter betrachtet und optimiert [16].
Daraufhin folgen Experimente zur Evaluierung der Modellleistung, bei denen die Leistung des Modells bei der Fallvorhersage gemessen wird. Der Fokus liegt dabei darauf, wie genau das Modell den Tenor eines Falls vorhersagen kann, wenn der Tatbestand als Eingabe dient. Verschiedene Prompts werden entworfen und verglichen, um zu ermitteln, inwiefern das Modell in der Lage ist, mit Hilfe des Tatbestands den Tenor korrekt vorherzusagen. Anschließend werden verschiedene Retrieval-Methoden miteinander verglichen [17]. Als Evaluationsmetriken für das RAG-Modell kommen unter anderem Precision@k, Recall@k, ROUGE, BLEU, MRR und METEOR zum Einsatz [8]. Diese Metriken benutzen als Referenz die Entscheidungsgründe der zugehörigen Fälle. Die Entscheidungsgründe dienen als sinnvoller Standard, da sie bereits von einem Gericht als relevant für den Fall eingestuft wurden. Das RAGAS-Framework wird als effizientes Evaluierungstool genutzt [6]. Diese Experimente liefern wichtige Erkenntnisse darüber, inwieweit das RAG-Modell relevante Informationen abrufen und in die Analyse juristischer Texte integrieren kann.
Basierend auf den Experimenten wird eine Analyse der Ergebnisse durchgeführt, die Optimierungsmöglichkeiten für den Abrufmechanismus aufzeigt. Ziel dieser Optimierungen ist es, die Effizienz und Genauigkeit des Modells weiter zu steigern.
Abschließend wird der Abschlussbericht erstellt, in dem die Erkenntnisse und Ergebnisse des Projekts zusammengefasst und kritisch bewertet werden.

# 4 Verwandte Forschung

Die Forschung im Bereich der automatisierten Analyse von juristischen Texten hat in den letzten Jahren zunehmend an Bedeutung gewonnen. Ein wesentlicher Aspekt ist die Verfügbarkeit geeigneter Datensätze und die Entwicklung von Modellen zur Verarbeitung juristischer Texte.
Darji et al. [5] präsentieren einen annotierten Datensatz für deutsche Gerichtsurteile, der speziell für Aufgaben der Rechtsreferenzvorhersage entwickelt wurde. Dieser Datensatz ermöglicht es Forschern, kontextuelle Ähnlichkeiten zwischen Fällen und Gesetzen zu analysieren und trägt zur Entwicklung von Modellen bei, die Verweise in juristischen Texten genauer vorhersagen können. Die Forscher betonen die Bedeutung gut strukturierter Datensätze, um in der juristischen Domäne valide Ergebnisse zu erzielen.
Salemi und Zamani [25] führen eine detaillierte Untersuchung zur Bewertung von Retrieval-Modellen in RAG-Systemen durch. Sie stellen das eRAG-Modell vor, das eine effizientere Methode zur Bewertung der Qualität von Retrieval-Systemen bietet, indem jedes abgerufene Dokument einzeln vom Sprachmodell verarbeitet wird. Diese Methode zeigt eine signifikante Korrelation mit der Downstream-Performance der RAG-Modelle und bietet zudem erhebliche Vorteile hinsichtlich Speicher- und Rechenzeiteffizienz im Vergleich zu herkömmlichen End-to-End-Bewertungen. Die Autoren betonen die Bedeutung einer präzisen Evaluation der abgerufenen Dokumente, um die Gesamtleistung von RAG-Systemen zu verbessern.
Hou et al. [11] stellen den CLERC-Datensatz vor, der speziell für die Fallvorhersage und die Erstellung juristischer Analysen entwickelt wurde. Dieser Datensatz dient der Evaluierung von Retrieval- und Generierungsmodellen im Bereich der juristischen Textverarbeitung. Die Ergebnisse ihrer Studie zeigen, dass aktuelle Modelle, einschließlich GPT-4, bei der Generierung von Analysen häufig Halluzinationen erzeugen und dass die Integration eines zuverlässigen Retrieval-Systems notwendig ist, um relevante rechtliche Präzedenzfälle zu identifizieren.
Glaser et al. [10] führen Forschungen zur automatisierten Zusammenfassung von deutschen Gerichtsurteilen durch. In ihrer Arbeit entwickeln sie einen Datensatz von 100.000 Urteilen und testen verschiedene Modelle für extraktive und abstrakte Zusammenfassungen. Ihre Forschung legt die Grundlage für zukünftige Arbeiten im Bereich der Zusammenfassung juristischer Texte und verdeutlicht die Herausforderungen, die bei der Verarbeitung von strukturierten rechtlichen Dokumenten wie Gerichtsurteilen auftreten.
Diese Arbeiten verdeutlichen, dass die Kombination von großen Sprachmodellen und Retrieval-Systemen vielversprechende Ansätze für die juristische Textanalyse bietet. Insbesondere in der Analyse von Gerichtsurteilen können RAG-Modelle durch den Abruf relevanter Informationen die Genauigkeit und Verlässlichkeit der Ergebnisse erhöhen.

# 5 Literaturverzeichnis

# Literatur


[1] P. Béchard und O. M. Ayala. "Reducing hallucination in structured outputs via Retrieval-Augmented Generation". In: *arXiv preprint arXiv:2404.08189* (2024).

[2] M. II Bommarito und D. M. Katz. *GPT Takes the Bar Exam*. Accessed: 2024-10-21. 2022. arXiv: 2212.14402 [cs.CL]. URL: https://arxiv.org/abs/2212.14402.

[3] T. B. Brown u. a. "Language Models are Few-Shot Learners". In: *arXiv preprint arXiv:2005.14165* (2020).

[4] A. Chowdhery u. a. "PaLM: Scaling Language Modeling with Pathways". In: *arXiv preprint arXiv:2204.02311* (2022).

[5] H. Darji, J. Mitrović und M. Granitzer. "A Dataset of German Legal Reference Annotations". In: *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*. 2023, S. 392–396.

[6] S. Es u. a. "Ragas: Automated evaluation of retrieval augmented generation". In: *arXiv preprint arXiv:2309.15217* (2023).

[7] Y. Gao u. a. *Modular RAG: Transforming RAG Systems into LEGO-like Reconfigurable Frameworks*. Accessed: 2024-10-21. 2024. arXiv: 2407.21059 [cs.CL]. URL: https://arxiv.org/abs/2407.21059.

[8] Y. Gao u. a. *Retrieval-Augmented Generation for Large Language Models: A Survey*. Accessed: 2024-10-21. 2024. arXiv: 2312.10997 [cs.CL]. URL: https://arxiv.org/abs/2312.10997.

[9] Z. Gekhman u. a. *Does Fine-Tuning LLMs on New Knowledge Encourage Hallucinations?* Accessed: 2024-10-21. 2024. arXiv: 2405.05904 [cs.CL]. URL: https://arxiv.org/abs/2405.05904.

[10] I. Glaser, S. Moser und F. Matthes. "Summarization of German Court Rulings". In: *Proceedings of the Natural Legal Language Processing Workshop 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics, 2021, S. 180–189. DOI: 10.18653/v1/2021.nllp-1.19. URL: https://aclanthology.org/2021.nllp-1.19.

[11] A. B. Hou u. a. "CLERC: A Dataset for Legal Case Retrieval and Retrieval-Augmented Analysis Generation". In: *arXiv preprint arXiv:2406.17186* (2024).

[12] Landtag Baden-Württemberg. *Landtage überlegen Technologie bundesweit einzusetzen*. Accessed: 2024-09-19. Sep. 2024. URL: https://www.landtag-bw.de/home/aktuelles/dpa-nachrichten/2024/September/KW37/Donnerstag/b8016d95-1f57-47c8-8364-b49f9cb1.html.

[13] Langchain. *Retrieval-Augmented Generation (RAG) Tutorial*. Accessed: 2024-10-21. 2024. URL: https://python.langchain.com/docs/tutorials/rag/.

[14] S. Lappin. "Assessing the Strengths and Weaknesses of Large Language Models". In: *Journal of Logic, Language and Information* 33.1 (2024), S. 9–20. DOI: 10.1007/s10849-023-09409-x. URL: https://doi.org/10.1007/s10849-023-09409-x.

[15] P. Lewis u. a. "Retrieval-augmented generation for knowledge-intensive NLP tasks". In: *Advances in Neural Information Processing Systems* 33 (2020), S. 9459–9474.

[16] LlamaIndex. *Evaluating the ideal chunk size for a RAG system using LlamaIndex*. Accessed: 2024-10-30. Okt. 2023. URL: https://www.llamaindex.ai/blog/evaluating-the-ideal-chunk-size-for-a-rag-system-using-llamaindex-6207e5d3fec5.

[17] X. Ma u. a. *Query Rewriting for Retrieval-Augmented Large Language Models*. Accessed: 2024-10-21. 2023. arXiv: 2305.14283 [cs.CL]. URL: https://arxiv.org/abs/2305.14283.

[18] G. Marcus. "The next decade in AI: four steps towards robust artificial intelligence". In: *arXiv preprint arXiv:2002.06177* (2020).

[19] Open Legal Data. *Open Legal Data - Freie juristische Datenbank*. Accessed: 2024-10-20. 2024. URL: https://de.openlegaldata.io/.

[20] Open Legal Data. *Open Legal Data GitHub Repository*. Accessed: 2024-10-21. 2024. URL: https://github.com/openlegaldata/.

[21] Open Legal Data. *Über uns - Open Legal Data*. Accessed: 2024-10-21. 2024. URL: https://openlegaldata.io/about/.

[22] OpenAI. "GPT-4 Technical Report". In: *arXiv preprint arXiv:2303.08774* (2023).

[23] V. Patel. *Avianca Airline Lawsuit Undermined by ChatGPT*. Accessed: 2024-10-20. Mai 2023. URL: https://www.nytimes.com/2023/05/27/nyregion/avianca-airline-lawsuit-chatgpt.html.

[24] A. Roberts, C. Raffel und N. Shazeer. "How much knowledge can you pack into the parameters of a language model?" In: *arXiv preprint arXiv:2002.08910* (2020).

[25] A. Salemi und H. Zamani. "Evaluating Retrieval Quality in Retrieval-Augmented Generation". In: *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: Association for Computing Machinery, 2024, S. 2395–2400. ISBN: 9798400704314. DOI: 10.1145/3626772.3657957. URL: https://doi.org/10.1145/3626772.3657957.

[26] H. Touvron u. a. "LLaMA: Open and Efficient Foundation Language Models". In: *arXiv preprint arXiv:2302.13971* (2023).

[27] Z. Wang u. a. "History, development, and principles of large language models: an introductory survey". In: *AI and Ethics* (Okt. 2024). DOI: 10.1007/s43681-024-00583-7. URL: https://doi.org/10.1007/s43681-024-00583-7.

[28] H. S. Zheng u. a. *Take a Step Back: Evoking Reasoning via Abstraction in Large Language Models*. Accessed: 2024-10-21. 2024. arXiv: 2310.06117 [cs.LG]. URL: https://arxiv.org/abs/2310.06117.