Measuring the Groundedness of Legal Question-Answering Systems

Dietrich Trautmann, Natalia Ostapuk, Quentin Grail, Adrian Alan Pol, Guglielmo Bonifazi, Shang Gao and **Martin Gajek**
Thomson Reuters Labs, Zug, Switzerland
{first.last}@tr.com

Abstract

In high-stakes domains like legal question-answering, the accuracy and trustworthiness of generative AI systems are of paramount importance. This work presents a comprehensive benchmark of various methods to assess the groundedness of AI-generated responses, aiming to significantly enhance their reliability. Our experiments include similarity-based metrics and natural language inference models to evaluate whether responses are well-founded in the given contexts. We also explore different prompting strategies for large language models to improve the detection of ungrounded responses. We validated the effectiveness of these methods using a newly created grounding classification corpus, designed specifically for legal queries and corresponding responses from retrieval-augmented prompting, focusing on their alignment with source material. Our results indicate potential in groundedness classification of generated responses, with the best method achieving a macro-F1 score of 0.8. Additionally, we evaluated the methods in terms of their latency to determine their suitability for real-world applications, as this step typically follows the generation process. This capability is essential for processes that may trigger additional manual verification or automated response regeneration. In summary, this study demonstrates the potential of various detection methods to improve the trustworthiness of generative AI in legal settings.

1 Introduction

Generative AI systems are increasingly employed in high-stakes domains such as legal question-answering, where accuracy and trust are paramount (Monroy et al., 2009; Vold and Conrad, 2021; Khazaeli et al., 2021; Martinez-Gil, 2023). A significant challenge in these applications is the detection of outputs that are not grounded in the input data (context), which can compromise user trust and diminish the application's value (Maynez et al., 2020; Rawte et al., 2023). This work addresses this challenge by conducting a comprehensive benchmarking to assess the groundedness of AI-generated legal responses, thereby enhancing their reliability.

Query

When does the duty to preserve documents end? Focus on SDNY case law.

Response A

The duty to preserve evidence arises when a party reasonably anticipates litigation. This duty generally requires the party to suspend its routine document retention/destruction policy and put in place a 'litigation hold' to ensure the preservation of relevant documents. The duty to preserve evidence ends when the litigation is resolved or when the party **no longer reasonably anticipates litigation**.

Response B

The duty to preserve evidence arises when a party reasonably anticipates litigation. This duty generally requires the party to suspend its routine document retention/destruction policy and put in place a 'litigation hold' to ensure the preservation of relevant documents. The duty to preserve evidence ends when the litigation is resolved or when the party **files a motion to dismiss**.

Figure 1: Example query and corresponding LLM responses with **grounded** and **erroneous** spans (Procedural Errors). The retrieved context used for grounding the responses was omitted due to its length. The remaining sentences in both responses are identical and grounded, but not highlighted to emphasize the differences.

Our methodology investigates diverse approaches to classify responses based on their foundation in the provided source material (cf. Fig. 1). We utilize:

1. *Similarity-based techniques*, employing various text similarity metrics to quantify the alignment between the generated text and the input data at the sentence-level.
2. *Natural language inference* models to determine if the generated response sentences are entailed by or contradict the sentences in the source material.
3. Diverse *prompting strategies* for large language models (LLMs) to detect ungrounded responses. (Bubeck et al., 2023).

Proceedings of the Natural Legal Language Processing Workshop 2024, pages 176–186

We evaluate these approaches on a new corpus of legal queries and responses, annotated for their degree of groundedness.

Experimental results demonstrate the effectiveness of many methods in the detection of potentially ungrounded answers. We also discuss the trade-offs between task performance and computational efficiency, highlighting the capabilities of particular approaches to operate with minimal added latency in real-world applications.

Furthermore, we investigated the types of errors present in the responses, categorizing them into six distinct classes: *Factual Inaccuracies*, *Contextual Misinterpretations*, *Procedural Errors*, *Reasoning Errors*, *Misattributions*, and *Terminological Errors*. Our analysis reveals that factual inaccuracies are the most prevalent type of errors. Importantly, we found that the misclassification rates in the overall groundedness assessment task are not uniform across these error categories, providing valuable insights for targeted improvements in AI-generated legal responses.

Our findings underscore the potential of automated groundedness assessment tools to improve the reliability and utility of generative AI in legal settings, ensuring that the generated responses are consistently accurate and trustworthy. The error analysis further contributes to a nuanced understanding of the challenges in this domain, paving the way for more refined and effective AI systems in legal applications.

2 Related Work

2.1 Grounding of Generated Responses

Grounding and factual consistency in language model outputs, especially for summarization and question-answering tasks, have been a focal point of recent research. Kryściński et al. (2020) introduced a weakly-supervised, model-based approach to verify factual consistency between source documents and generated summaries. This method uniquely combines consistency checks with the extraction of supporting and contradictory spans.

Building on this, Maynez et al. (2020) performed an extensive human evaluation of neural abstractive summarization systems. Their results showed a significant amount of ungrounded content in model-generated summaries and found that textual entailment measures correlate more strongly with faithfulness than standard metrics. This finding closely relates to our interest in assessing the groundedness of AI-generated legal responses.

The Chain-of-Knowledge (CoK) framework (Li et al., 2023) marks a major advance in reducing hallucinations. By dynamically incorporating grounding information from various sources, CoK enhances factual accuracy in knowledge-intensive tasks.

In essence, grounding of LLM-generated responses aims to ensure that outputs are factually consistent with input data, thereby enhancing reliability and reducing ungrounded LLM-generated content.

2.2 Hallucination Detection

Advancements in hallucination detection have been pivotal in developing more reliable and grounded LLMs, particularly for question-answering (QA) systems.

The *HaluEval-Wild* benchmark (Zhu et al., 2024) offers a novel approach to evaluating LLM hallucinations in real-world settings. By categorizing challenging user queries into five distinct types, this tool provides essential insights for enhancing LLM reliability in scenarios that mirror real-world interactions, which is crucial for QA systems.

Wang et al. (2024) contribute with *MIGRES*, a method that uses LLMs' ability to identify missing information for targeted knowledge retrieval and extraction. This approach promises to improve the groundedness of responses by ensuring comprehensive information gathering.

In long-form question answering, Rosenthal et al. (2024) introduced *ClapNQ*, a benchmark designed for retrieval-augmented generation (RAG) systems. Its emphasis on concise, cohesive answers grounded in source passages makes it particularly relevant for evaluating QA systems that require detailed, well-supported responses.

An empirical evaluation of AI-driven legal research tools (Magesh et al., 2024) challenges claims of "hallucination-free" systems, underscoring the necessity for rigorous evaluation in assessing the groundedness of legal QA systems.

Additionally, Hong et al. (2024) have launched the *Hallucinations Leaderboard*, an open initiative for measuring and comparing hallucinations across various LLMs and tasks. This resource offers a valuable opportunity for benchmarking the groundedness of QA systems against a diverse range of models and applications.

3 Grounding Definition

Grounding in legal question-answering systems refers to the extent to which an AI-generated response is firmly rooted in, supported by, and directly attributable to the provided legal source material. It ensures the model's output aligns with and accurately represents the information in the input data, avoiding fabrication, extraneous details, or misleading content. A well-grounded response should adhere closely to the facts, legal principles, and reasoning presented in the source material, without introducing unsupported claims or misrepresenting the legal context (Chandu et al., 2021).

Several key aspects ensure the reliability of AI-generated legal responses. Factual alignment and relevance are crucial, ensuring the content reflects the source documents and addresses the legal query accurately. Source attribution allows tracing information back to specific input texts, while legal interpretation fidelity ensures conclusions are substantiated by the provided materials. This involves not only accurately conveying factual information but also maintaining the integrity of legal procedures, correctly interpreting the context, and using appropriate legal terminology. The generated responses must adhere to the given context, avoiding unsupported claims or extrapolations, and preserving the nuances and complexities of legal language and concepts (Magesh et al., 2024).

The assessment of grounding in legal AI responses involves a comprehensive evaluation of how faithfully the generated content aligns with the retrieved legal context. This evaluation considers various aspects of the response, including its factual accuracy, the appropriateness of legal interpretations, the coherence of legal reasoning, and the proper use of legal terminology. Grounding is vital in legal applications to maintain the integrity of legal advice, ensure compliance with laws and precedents, and prevent misinformation. By ensuring strong grounding, legal question-answering systems can provide more reliable, trustworthy, and legally sound responses, which is crucial in the high-stakes environment of legal practice and decision-making.

4 Dataset Creation

In this section, we will describe and list all the steps involved in creating the *Groundedness Classification* dataset used in our benchmarking.

4.1 Data Source

The dataset originates from proprietary data in the *Casetext* Legal Research Skill[1]. We limited the data selection to the internal users only, primarily consisting of diverse sales demonstrations as well as domain experts and engineering-related testing sessions. All queries, however, are realistic representations of everyday research in the legal domain. Additionally, we performed a deduplication process on the input queries.

The dataset comprises input queries (e.g., questions about particular legal use cases) accompanied by LLM-generated responses and retrieved context data. During development, legal professionals verified these responses to ensure they were grounded in the context provided to the LLM (as part of the prompt). The context data is derived from a retrieval system with access to the *Casetext* database for legal research, which includes case law, statutes, regulations, and legal texts authored by internal legal experts and lawyers.

The ground truth responses (LLM-based answers) were generated using custom instructions in a prompt to *GPT-4* in the current production environment. At this stage of the dataset creation process, we had compiled a selection of legal user queries, gold responses, and their corresponding contexts.

4.2 Synthetic Adaptation

The next step in our dataset creation process involved generating evoked ungrounded responses to evaluate both grounded and ungrounded outputs. We instructed *GPT-4o* to make subtle and unintrusive variations to the original grounded responses, preserving most of the meaning while introducing minor deviations from the provided context. In the prompt, we included the original query and context alongside the gold response and these instructions.

These adapted responses, which we consider partially ungrounded[2], complement our final dataset. The inclusion of both grounded and ungrounded responses allows for a more comprehensive evaluation of response quality and adherence to provided context. An example of this subtle deviation from the source material in the generated response was depicted in the leading example in Fig. 1.

[1] https://casetext.com/cocounsel/

[2] Only some sentences ended up with slight modifications, while most were kept as the original sentences.

Split	#Queries	#Responses	#Response Sentences
Training	400	1080	5671
Development	58	162	797
Testing	115	316	1516
Total	573	1558	7984

Table 1: Data Set Statistics

4.3 Data Splits

We divided the dataset into training, development, and test sets using a ratio of 70 : 10 : 20, respectively. This split ensures a representative distribution across all subsets while maintaining a sufficiently large test set for robust evaluation.

The resulting counts for each split are presented in Table 1. It is noteworthy that the number of responses is not exactly twice the number of queries. This discrepancy arises from our dataset creation process, where we retained multiple significant variations of generated responses for certain queries to enhance the diversity and coverage of our dataset.

To maintain the integrity of our evaluation, we ensured that all responses corresponding to a particular query were assigned to the same split. This approach prevents potential leakage between the training and evaluation sets, thereby providing a more accurate assessment of model performance on unseen data.

5 Benchmarking Methodologies

This section overviews the diverse methodologies employed in our benchmark study for quantifying response grounding, systematically evaluating approaches that assess adherence of generated responses to provided context.

5.1 Similarity-based Approaches

Similarity-based approaches compare each response sentence against all context sentences, allowing for detailed grounding assessment. We aggregate these sentence-level estimations for the final response-level prediction.

Semantic Similarity We embedded sentences using the *nlpaueb/legal-bert-base-uncased* model with the Sentence-Transformers library. Matching pairs were identified using cosine similarity, with an optimized threshold determined on the development set for final grounding prediction.

Quoted Information Precision Adapting the *QuIP-score* (Weller et al., 2024), we examined character n-gram overlap between LLM responses and context sentences. We optimized both the n-gram size (21 in our setup) and similarity threshold on the development set for grounding determination in the final evaluation.

5.2 Natural Language Inference

FactKB Evaluating factual consistency in natural language generation is crucial, especially for complex domains. We employed FactKB[3], an approach leveraging pre-training with facts from external knowledge bases, to address challenges in entity and relation errors (Feng et al., 2023).

FactKB has shown state-of-the-art performance in factual consistency evaluation across various domains. We used it to compute factuality scores of generated response sentences against source context sentences.

Our grounding determination process involved identifying the highest-scoring source sentence for each target sentence based on *FactKB* scores, then applying an optimized threshold to classify grounding sufficiency. This threshold, determined using our development set, balanced precision and recall in grounding classification, adapting *FactKB* to our specific task of response grounding quantification.

Hallucination Evaluation Models The Hallucination Evaluation Model (HEM), developed by Vectara (Hughes et al., 2023), is designed to detect hallucinations in LLM-generated responses. HEM is available in two versions: *V1*, a fine-tuned model based on *cross-encoder/nli-deberta-v3-base*, and *V2*, an improved version using *flan-t5-base*.

Built on research in factual consistency for summarization, HEM classifies whether a summary is factually consistent with its source. The model was fine-tuned on diverse documents to ensure robustness across content types and is publicly available on Hugging Face under the Apache 2 license.

[3] https://hf.co/bunsenfeng/FactKB

HEM evaluates LLM responses by comparing them to source documents, classifying summaries as consistent or inconsistent. For our study, we implemented a fine-grained approach, scoring individual sentences against corresponding contexts. This granular analysis provides a nuanced assessment of hallucinations at the sentence level, offering deeper insights into model performance.

5.3 Prompting Approaches

Direct Prompting One straight-forward approach for groundedness classification via prompting is asking either the same or another LLM whether a particular response for a query is grounded in a context or not (Trautmann et al., 2022). Therefore, we utilized several LLMs with a custom prompt and collected the binary classification as the prompt-based baselines. We used the specialized open access model *Lynx-v1.1* (Ravi et al., 2024) and the general purpose public LLMs *GPT-4o* and *Claude Sonnet 3.5*. All three LLMs were evaluated with the same prompt from Ravi et al. (2024).

In principle, this approach has similarities with *Reflexion* by Shinn et al. (2024), where a *Self-Reflection* LLM should reflect on a previous answer and if necessary to update its prediction. The authors showed that this was helpful, especially for more complex tasks.

Amazon RefChecker RefChecker (Hu et al., 2024) introduces a framework for hallucination detection using knowledge triplets to capture fine-grained assertions. The process involves three steps: claim extraction, hallucination checking, and aggregation. This decoupled process is also known as prompt chaining (Trautmann, 2023).

An LLM identifies knowledge triplets from the response to the original query. Zero-shot checkers then predict hallucination labels for each triplet (entailment, contradiction, or neutral). Finally, these labels are integrated to compute an overall hallucination score for the response.

RefChecker's computational demands are notable: for n triplets extracted, the LLM is prompted with the entire original context n times, significantly impacting processing time and resource consumption. This approach balances granular analysis with computational intensity, offering a detailed but resource-intensive method for hallucination detection.

SelfCheckGPT We adapt the approach of Manakul et al. (2023), which assesses hallucination likelihood in LLM-generated sentences by evaluating their consistency with multiple answers from the same query. SelfCheckGPT assumes that grounded sentences should be consistent with other sampled answers.

The method generates new responses using the initial prompt with increased temperature. It then calculates a hallucination score for each sentence as the average of contradiction probabilities with these new samples. The response-level score is the maximum of sentence-level scores, with the threshold optimized on the training set.

We enhance this approach with a novel context-based evaluation (*ContextNLI*) using the *potsawee/deberta-v3-large-mnli* model. This compares each answer sentence against context sentences, identifying the minimum contradiction score as the hallucination probability. The maximum score across all sentences represents the answer's overall hallucination likelihood.

We implement two variants of this approach: *Multi-Gen*, which follows the original consistency checks, and our novel *ContextNLI*, which incorporates the context-based evaluation, thus providing complementary methods for assessing the groundedness of LLM-generated content.

DeepEval: Claims Extraction and Verification We adapt the *Faithfulness* metrics from Ip (2023) to detect contradictions between source documents and generated answers. This approach divides the task into two subtasks: claims extraction and claim verification (prompt chains, Trautmann (2023)).

First, we use an LLM to extract claims independently from both source documents and generated answers using a custom prompt. Then, a second LLM call with another custom prompt identifies claims from the generated answer not factually supported by the source document claims. If any generated claim contradicts a source claim, we consider the answer inaccurate.

This method requires three LLM calls in total: two for claims extraction and one for comparison. We utilize *Claude Sonnet 3.5* for all these calls, balancing task complexity reduction with comprehensive analysis.

5.4 Fine-Tuning

In addition to our primary methods, we fine-tuned a Cross-Encoder classifier (*DeBERTa v3* as the base

model) specifically tailored to our dataset. To ensure the integrity of our evaluation, we meticulously prepared a specialized training and evaluation corpus based on the initial data splits, thereby avoiding any potential contamination between sets.

Our fine-tuning approach focused on the nuanced differences between grounded and ungrounded responses. For each pair of such responses, we isolated the sentences that differed between them. This selective process allowed us to concentrate on the most informative elements for distinguishing between grounded and ungrounded content.

To establish ground truth for the grounded responses, we employed a semantic similarity measure (as described in Section 5.1). For each sentence in the grounded response, we identified the most semantically similar sentence from the context and assigned it the corresponding cosine similarity score. These scores typically ranged from 0.8 to 0.99, indicating high levels of semantic alignment.

Conversely, for the ungrounded responses, we paired each sentence with the same context sentence used for its grounded counterpart. However, we assigned these pairs a score of 1 minus the cosine similarity, effectively inverting the grounding measure. This approach provided a balanced representation of both grounded and ungrounded examples in our training data.

Through this methodology, we compiled a balanced dataset comprising 558 samples for training and 75 for development. This carefully curated dataset served as the foundation for our fine-tuning process, enabling the Cross-Encoder to learn the subtle distinctions between grounded and ungrounded content within our specific corpus.

The outcomes of our fine-tuning efforts (after hyper-parameter optimization), are comprehensively presented (macro averaged) in Tab. 2.

Model Name	M-Prec	M-Rec	M-F1	Acc
deberta-v3-base	0.459	0.466	0.450	0.493
deberta-v3-large	0.736	0.739	0.733	0.733

Table 2: DEV set metrics for DeBERTa models

Following the fine-tuning stage, we integrated this grounding classification (GC) model into our benchmark, employing a methodology analogous to that used for the NLI approaches described in Section 5.2.

6 Experimental Set-Up

Our benchmarking study aimed to evaluate various methods for classifying LLM responses as grounded or ungrounded relative to a given context and query.

Methodology Despite the varied granularity of approaches (response-level vs. sentence-level), we standardized outputs to binary classifications for consistent comparison. We developed each method on the training set, optimized parameters on the development set, and conducted final evaluations on the test set.

Performance Metrics We assessed classification accuracy (including macro-averaged f1, precision, and recall) and computational efficiency through latency measurements. Latency was computed as the average processing time across all samples in the development set. These metrics provide insights into each approach's practical applicability.

Computational Resources Local approaches utilized Amazon EC2 G5 Instances (8xlarge)[4]. Prompting-based methods were executed via Azure OpenAI Services[5], AWS Bedrock (Anthropic's Claude)[6], and Anthropic's API directly, ensuring diverse and robust evaluation environments.

7 Groundedness Classification Results

Our benchmark evaluation of groundedness classification approaches revealed insightful performance trade-offs, as shown in Tab. 3. The metrics include classification precision, recall, F1-score, and accuracy, providing a comprehensive view of each method's applicability.

The multi-stage prompt chaining approach, *DeepEval Claims Verify*, achieved top classification metrics, but with high latency (26.1 seconds per request). In contrast, *direct prompting* with *GPT-4o* achieved the second-highest scores with significantly lower latency (2.2 seconds), as illustrated in Fig. 2.

A clear speed-performance trade-off emerged across methods. Similarity-based approaches (*COS_SIM* and *QUIP*) were fastest but struggled with ungrounded response identification. NLI methods showed improved performance at the cost

[4] https://aws.amazon.com/ec2/instance-types/g5/

[5] https://azure.microsoft.com/en-us/products/ai-services/openai-service

[6] https://aws.amazon.com/bedrock/claude/

#	Model Name	Development Set				Test Set			
		Precision	Recall	Macro-F1	Accuracy	Precision	Recall	Macro-F1	Accuracy
1	COS_SIM	0.525	0.520	0.494	0.520	0.497	0.497	0.493	0.497
2	QUIP	0.648	0.533	0.421	0.533	0.560	0.509	0.379	0.509
3	HEM V1	0.640	0.640	0.640	0.640	0.598	0.595	0.592	0.595
4	HEM V2	0.580	0.580	0.580	0.580	0.564	0.563	0.562	0.563
5	FACT_KB	0.527	0.527	0.526	0.527	0.510	0.510	0.508	0.510
6	GC-large	0.694	0.667	0.655	0.667	0.628	0.620	0.615	0.620
7	LYNX v1.1	0.764	0.460	0.571	0.460	0.792	0.503	0.597	0.503
8	Sonnet 3.5	0.728	0.727	0.726	0.727	0.724	0.715	0.712	0.715
9	GPT-4o	0.783	0.773	0.771	0.773	**0.802**	0.763	0.755	0.763
10	RefChecker (Haiku)	0.511	0.506	0.450	0.506	0.514	0.507	0.435	0.508
11	RefChecker (Sonnet 3)	0.500	0.500	0.366	0.500	0.500	0.500	0.386	0.500
12	DeepEval Claims Verify	**0.801**	**0.800**	**0.800**	**0.800**	0.779	**0.774**	**0.774**	**0.775**
13	SCGPT (Multi-Gen)	0.627	0.627	0.627	0.627	0.679	0.667	0.661	0.667
14	SCGPT (ContextNLI)	0.620	0.620	0.620	0.620	0.610	0.604	0.600	0.604

Table 3: Performance comparison of different models on Development and Test sets

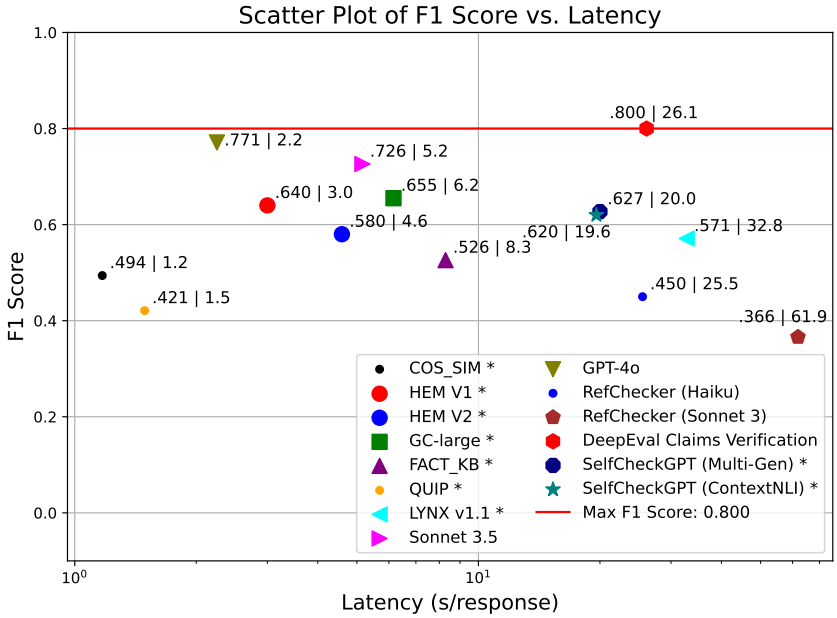

Figure 2: Development set results for our benchmark. We report the F1-scores (y-axis) for each method and the corresponding latency (x-axis) in seconds per response. Approach names denoted with * were run on an AWS *ml.8xlarge* instance.

of increased latency. Within NLI, *HEM V1* outperformed *HEM V2*, and fine-tuning on our corpus further improving results.

Unexpectedly, complex prompt chaining approaches like *RefChecker* and *SelfCheckGPT* underperformed, highlighting challenges in developing universally effective methods across diverse contexts.

These findings emphasize the importance of balancing task performance and computational efficiency when selecting a groundedness classification approach, with optimal choices depending on specific application requirements and resource constraints.

8 Error Analysis

We conducted a detailed investigation into the types of response errors present in our benchmark dataset to gain deeper insights into ungrounded content.

Through examination of error spans in the training set, we identified six distinct error types. The models were instructed to select from our predefined error types (Tab. 5, App. A.2).

Focusing on the development set, our analysis revealed interesting patterns. The LLMs achieved exact agreement on the hallucination type in 29% of cases, with at least one overlapping error type for each response. *GPT-4o* typically predicted a

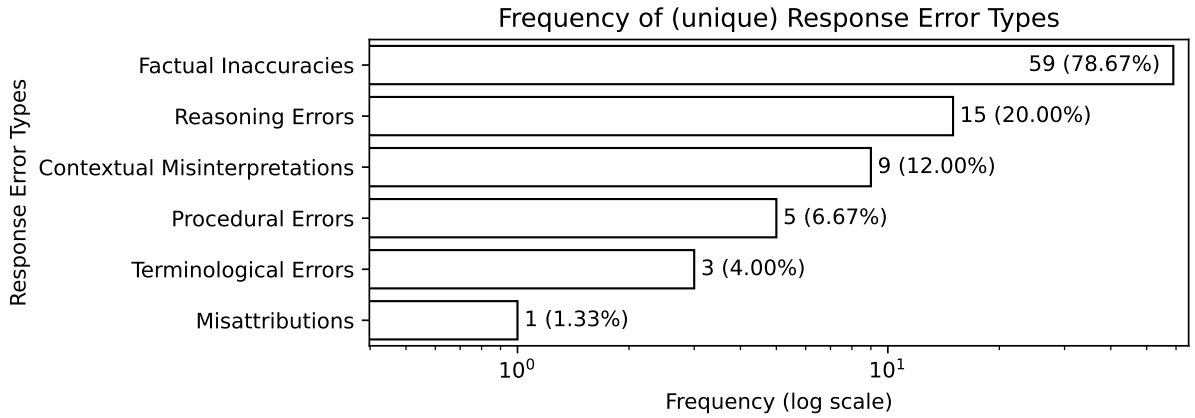

Figure 3: Counts of unique error types in the development set. Some responses contained up to three different error types. The frequency axis is in log-scale.

Error Type	Misclassified	Total	Percentage
Terminological Errors	2	3	66.7%
Factual Inaccuracies	12	59	20.3%
Procedural Errors	1	5	20.0%
Reasoning Errors	2	15	13.3%
Contextual Misinterpretations	1	9	11.1%
Misattributions	0	1	0.0%

Table 4: Development set misclassification of the best performing model by error types.

single error type, while *Claude-3.5-Sonnet* often suggested multiple types per response.

We aggregated predictions where both LLMs agreed. The distribution of unique error types is visualized in Fig. 3, with per-response occurrences in Fig. 4 (App. A.1). *Factual Inaccuracies* were most common, followed by *Reasoning Errors*. All initially defined error types were represented, validating our classification scheme.

This analysis provides valuable insights into response error types and ungrounded content in language model outputs, crucial for developing targeted strategies to improve response generation.

Misclassification Analysis We conducted a misclassification analysis on our best-performing model, *DeepEval Claims Verify*, to gain deeper insights into its performance across different error types. As summarized in Tab. 4, *Terminological Errors* showed the highest misclassification rate (67%), despite their low frequency, followed by *Factual Inaccuracies* (20%) and *Procedural Errors* (20%). These findings reveal the varying challenges posed by different error categories and highlight areas for potential improvement in groundedness classification models, particularly in handling less common but difficult-to-classify error types.

9 Conclusion

Our comprehensive benchmark study on groundedness classification of legal question-answering systems has revealed significant insights into performance and efficiency trade-offs. The multi-stage prompt chaining approach, *DeepEval Claims Verify*, emerged as the top performer with an F1 score of 0.80, closely followed by direct prompting using *GPT-4o* at 0.77, which demonstrated lower latency. These results highlight the potential of advanced prompting techniques in achieving high accuracy.

Similarity-based and natural language inference methods, while less accurate, offered fast processing times. Our response error type classification identified *Factual Inaccuracies* and *Reasoning Errors* as the most prevalent types of ungrounded content, providing direction for future improvements.

The study underscores the critical balance between task performance, computational efficiency, and ease of implementation when selecting groundedness classification methods. With top-performing methods achieving F1 scores of 0.80, this benchmark represents a significant advancement in the reliable assessment of AI-generated content across diverse applications.

Limitations

While our study offers valuable insights into the performance of various groundedness classification approaches, it is essential to acknowledge several limitations inherent in our experimental setup and the methods we evaluated.

Firstly, our dataset, though carefully curated, is limited in size and domain scope. The responses were generated using specific language models and may not fully represent the diverse range of hallucinations or ungrounded content that could occur across different models or domains. This limitation potentially affects the generalizability of our findings to broader contexts or more specialized applications.

Secondly, the binary classification of responses as either grounded or ungrounded may oversimplify the nuanced nature of language model outputs. In reality, responses often contain a mix of grounded and ungrounded elements, and a more granular assessment might provide deeper insights into model behavior.

Our evaluation metrics, while standard in the field, may not capture all aspects of response quality or usefulness. For instance, a response that is technically grounded but irrelevant or poorly structured might still receive a high rating within our current framework.

The computational resources required for some of the more complex approaches, particularly those involving multiple API calls or large language models, pose scalability challenges. This limitation may restrict the practical applicability of these methods in real-time or resource-constrained environments.

Additionally, our error type classification, while informative, relies on the agreement between two specific language models. This approach may introduce biases or limitations based on the particular characteristics of these models.

Lastly, the rapid pace of development in language model technology means that our findings may quickly become outdated as new models and techniques emerge. The performance gaps we observed between different approaches may shift with the introduction of more advanced models or refined methodologies.

Future work should address these limitations by expanding the dataset to include a broader range of domains and increasing its size. Developing more nuanced classification frameworks that can capture the complexity of language model outputs would also be beneficial. Furthermore, exploring scalable methods that can be applied in real-time or resource-constrained environments, as well as continuously updating the evaluation framework to reflect the latest advancements in language model technology, will be crucial for the ongoing relevance of this research.

Ethics Statement

This study on groundedness classification methods aims to improve the reliability and trustworthiness of AI-generated content, which has significant ethical implications. By developing more accurate methods to detect ungrounded or hallucinated information, we contribute to the broader goal of mitigating the spread of misinformation and enhancing the integrity of AI-assisted communication. Our work aligns with the principles of beneficence and non-maleficence, as it seeks to maximize the benefits of language models while minimizing potential harms associated with inaccurate or misleading information.

We acknowledge that the development and deployment of these classification methods may have broader societal impacts. We emphasize the importance of transparent and responsible use of these methods, respecting principles of fairness and privacy. Furthermore, we encourage ongoing dialogue and collaboration within the NLP community to address the ethical challenges associated with AI-generated content and its evaluation.

References

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. arxiv. *arXiv preprint arXiv:2303.12712*.

Khyathi Raghavi Chandu, Yonatan Bisk, and Alan W Black. 2021. Grounding 'grounding' in NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4283–4305, Online. Association for Computational Linguistics.

Shangbin Feng, Vidhisha Balachandran, Yuyang Bai, and Yulia Tsvetkov. 2023. Factkb: Generalizable factuality evaluation using language models enhanced with factual knowledge. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 933–952.

Giwon Hong, Aryo Pradipta Gema, Rohit Saxena, Xiaotang Du, Ping Nie, Yu Zhao, Laura Perez-

Beltrachini, Max Ryabinin, Xuanli He, and Pasquale Minervini. 2024. The hallucinations leaderboard–an open effort to measure hallucinations in large language models. *arXiv preprint arXiv:2404.05904*.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. 2024. Refchecker: Reference-based fine-grained hallucination checker and benchmark for large language models. *arXiv preprint arXiv:2405.14486*.

Simon Hughes, Minseok Bae, and Miaoran Li. 2023. Vectara Hallucination Leaderboard.

Jeffrey Ip. 2023. Deepeval: A tool for deep learning model evaluation. GitHub repository.

Soha Khazaeli, Janardhana Punuru, Chad Morris, Sanjay Sharma, Bert Staub, Michael Cole, Sunny Chiu-Webster, and Dhruv Sakalley. 2021. A free format legal question answering system. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 107–113.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346.

Xingxuan Li, Ruochen Zhao, Yew Ken Chia, Bosheng Ding, Shafiq Joty, Soujanya Poria, and Lidong Bing. 2023. Chain-of-knowledge: Grounding large language models via dynamic knowledge adapting over heterogeneous sources. In *The Twelfth International Conference on Learning Representations*.

Varun Magesh, Faiz Surani, Matthew Dahl, Mirac Suzgun, Christopher D Manning, and Daniel E Ho. 2024. Hallucination-free? assessing the reliability of leading ai legal research tools. *arXiv preprint arXiv:2405.20362*.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017.

Jorge Martinez-Gil. 2023. A survey on legal question–answering systems. *Computer Science Review*, 48:100552.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Alfredo Monroy, Hiram Calvo, and Alexander Gelbukh. 2009. Nlp for shallow question answering of legal documents using graphs. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 498–508. Springer.

Selvan Sunitha Ravi, Bartosz Mielczarek, Anand Kannappan, Douwe Kiela, and Rebecca Qian. 2024. Lynx: An open source hallucination evaluation model. *arXiv preprint arXiv:2407.08488*.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.

Sara Rosenthal, Avirup Sil, Radu Florian, and Salim Roukos. 2024. Clapnq: Cohesive long-form answers from passages in natural questions for rag systems. *arXiv preprint arXiv:2404.02103*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2024. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36.

Dietrich Trautmann. 2023. Large language model prompt chaining for long legal document classification. *arXiv preprint arXiv:2308.04138*.

Dietrich Trautmann, Alina Petrova, and Frank Schilder. 2022. Legal prompt engineering for multilingual legal judgement prediction. *arXiv preprint arXiv:2212.02199*.

Andrew Vold and Jack G Conrad. 2021. Using transformers to improve answer retrieval for legal questions. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, pages 245–249.

Keheng Wang, Feiyu Duan, Peiguang Li, Sirui Wang, and Xunliang Cai. 2024. Llms know what they need: Leveraging a missing information guided framework to empower retrieval-augmented generation. *arXiv preprint arXiv:2404.14043*.

Orion Weller, Marc Marone, Nathaniel Weir, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. 2024. "according to . . . ": Prompting language models improves quoting from pre-training data. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2288–2301, St. Julian's, Malta. Association for Computational Linguistics.

Zhiying Zhu, Zhiqing Sun, and Yiming Yang. 2024. Halueval-wild: Evaluating hallucinations of language models in the wild. *arXiv preprint arXiv:2403.04307*.

A Response Error Types

A.1 Dev Set Error Types

A.2 Description and Examples

See the table 5 for our six response error types with their descriptions and examples.

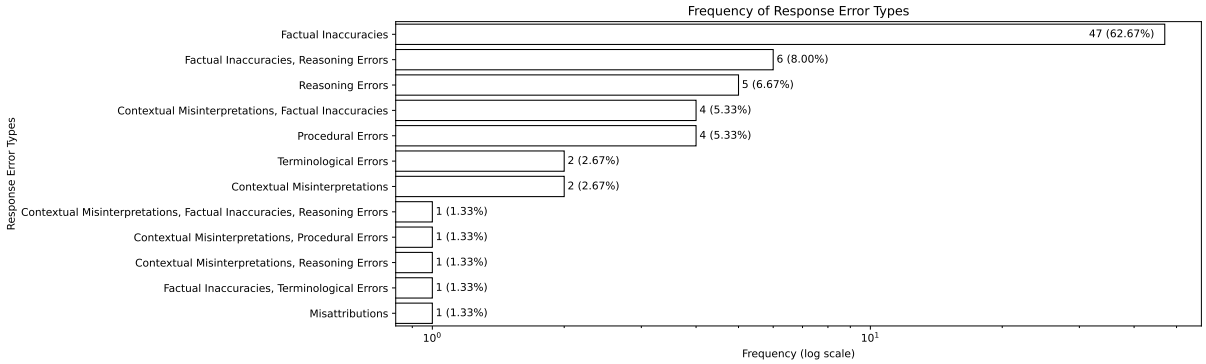

Figure 4: Counts of response error types in the development set. The frequency axis is in log-scale.

Error Type	**Short Description**	**Examples**
Factual Inaccuracies	Misrepresentation of established facts, dates, or details	1. Brown v. Board of Education was decided in 1964. 2. The First Amendment protects only written speech.
Contextual Misinterpretations	Misapplication of legal principles or inappropriate analogies	1. Applying Miranda rights to a civil tax dispute. 2. Using Roe v. Wade precedent in a Second Amendment case.
Procedural Errors	Mistakes in describing legal procedures or processes	1. A case goes directly from district court to the Supreme Court, skipping the appellate court. 2. Claiming that jury selection occurs after opening statements in a trial.
Reasoning Errors	Flawed arguments or unsupported legal conclusions	1. Since the Fourth Amendment protects against unreasonable searches, all warrantless searches are unconstitutional. 2. Because the Supreme Court ruled on abortion in Roe v. Wade, states cannot pass any abortion laws.
Misattributions	Incorrect assignment of opinions, quotes, or actions	1. Justice Scalia wrote the majority opinion in Obergefell v. Hodges. 2. The phrase "separate but equal" originated from Brown v. Board of Education.
Terminological Errors	Misuse or misinterpretation of legal terms or concepts	1. "Habeas corpus" refers to the right to a speedy trial. 2. "Strict scrutiny" means that a law is automatically unconstitutional.

Table 5: Response error types with a description and examples