RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models

Anonymous ACL submission

Abstract

Retrieval-augmented generation (RAG) has become a main technique for alleviating hallucinations in large language models (LLMs). Despite the integration of RAG, LLMs may still present unsupported or contradictory claims to the retrieved contents. In order to develop effective hallucination prevention strategies under RAG, it is important to create benchmark datasets that can measure the extent of hallucination. This paper presents RAGTruth, a corpus tailored for analyzing word-level hallucinations in various domains and tasks within the standard RAG frameworks for LLM applications. RAGTruth comprises nearly 18,000 naturally generated responses from diverse LLMs using RAG. These responses have undergone meticulous manual annotations at both the individual case and word levels, incorporating evaluations of hallucination intensity. We not only benchmark hallucination frequencies across different LLMs, but also critically assess the effectiveness of several existing hallucination detection methodologies. We show that using a high-quality dataset such as RAGTruth, it is possible to finetune a relatively small LLM and achieve a competitive hallucination detection performance when compared to the existing prompt-based approaches using state-of-the-art LLMs such as GPT-4. Furthermore, the finetuned model can effectively mitigate hallucination in LLM responses.

Question: how to prepare to get an ultrasound?

Reference: Passage 1: 1 If you are just interested in checking the gender, you can go as early as 17 weeks. 2 Any earlier, and the results may not be accurate, leading to an extra coat of paint for your bedroom. 3 To get good pictures, 3D ultrasounds are best performed between 20 and 32 weeks, and the best pictures are between 24 and 30 weeks.

...

Response: Based on the given passages, here's how to prepare for an ultrasound:

* For a gender reveal ultrasound, wait until at least 17 weeks of pregnancy to ensure accurate results.

* For 3D ultrasounds, schedule the appointment between 20 and 32 weeks of pregnancy for the best pictures ...

Annotation

Span: between 20 and 32 weeks of pregnancy for the best pictures

Type: Evident Conflict

Reason: Original: "the best pictures are between 24 and 30 weeks", Generative: "between 20 and 32 weeks of pregnancy for the best pictures"

Table 1: An example of RAGTruth data from the question answering task. It contains context, response generated by LLM with and span-level annotation.

1 Introduction

Large language models (LLMs) have achieved remarkable success in a variety of tasks, including text generation (Li et al., 2022), machine translation (Kocmi and Federmann, 2023), and question answering (Zhao et al., 2023). However, one of the key challenges in deploying LLMs in real-world applications is their tendency to hallucinate (Kaddour et al., 2023). Hallucination in the context of LLMs usually refers to a situation where the model generates content that is not based on factual or accurate information (Rawte et al., 2023). The occasional generation of outputs that appear plausible but are factually incorrect significantly undermine the reliability of LLMs in real-world scenarios, such as medical diagnoses (Pal et al., 2023) and news summarization (Shen et al., 2023).

To reduce hallucination, various methods have been developed that can be applied at different stages of LLM lifecycle, including pretraining (Brown et al., 2020), supervised finetuning (Zhou et al., 2023; Zhang et al., 2023a), RLHF (Ouyang et al., 2022; Lin et al., 2022), and inference (Dhuliawala et al., 2023; Gao et al., 2023). In terms of detection, methods are developed by examining the model's intrinsic state (Guo et al., 2017), comparing it with external data and tools (Chern et al., 2023), or leveraging the

LLM's inherent powerful capabilities for self-checking (Agrawal et al., 2023; Manakul et al., 2023). Retrieval-augmented generation (RAG) is extensively used to supply LLMs with updated, relevant knowledge, significantly mitigating hallucination (Varshney et al., 2023). Nevertheless, even with RAG and other enhancements, LLMs still produce statements that are either unfounded or contradict the information provided in the retrieved references (Shuster et al., 2021).

Despite the growing awareness of the hallucination phenomenon, the understanding of hallucination in LLMs is still in its early stages. One key challenge is the lack of high-quality, large-scale datasets specifically designed for hallucination detection. This issue is particularly acute in RAG settings. Due to the relatively low hallucination ratio, a substantial increase in annotation resources is needed. Existing datasets for LLM hallucination detection are predominantly synthesized (Li et al., 2023). For instance, in Liu and Liu (2023); Longpre et al. (2021), prompts conflicting with conventional knowledge are purposely generated to trigger hallucinations. While these approaches are efficient at generating hallucinations, the resulting artificial hallucinations can substantially differ from those that naturally occur. In Chen et al. (2023); Hu et al. (2023), hallucination datasets are developed by manual annotations of naturally produced LLM responses. However, these datasets are of limited size and are not specifically focused on the RAG scenario.

In this paper, we introduce a large-scale high-quality dataset specifically designed for word-level hallucination detection for RAG applications. Using this dataset, we have conducted an extensive benchmarking of mainstream LLMs to assess their tendency to generate hallucinations, as well as evaluate current methods for hallucination detection. Additionally, we have demonstrated superior performance in identifying hallucinations by fine-tuning LLM with RAGTruth dataset. Our key contributions are:

(i) We propose RAGTruth, a large-scale word-level hallucination evaluation dataset specifically for the RAG scenario across several common tasks. It consists of nearly 18,000 fully annotated natural responses generated from major open-source and closed-source LLMs.

(ii) We perform a comprehensive comparison of different hallucination detection methods at both the passage and word levels.

(iii) We present a baseline method of fine-tuning LLM for hallucination detection. It is shown that by fine-tuning the Llama-2-13B model on the RAGTruth training data, we can achieve results competitive to the existing prompt-based approaches using GPT-4. This shows the potential of developing better hallucination detection methods using RAGTruth.

(iv) We show that by using our finetuned hallucination detector, it is possible to significantly reduce the occurrence of hallucinations in the responses from LLMs. The improvement holds even for models with inherently low hallucination rates, such as GPT-4.

2 Related Work

2.1 Hallucination of Large Language Models

Though hallucination in traditional natural language generation (NLG) contexts has been widely studied(Ji et al., 2023), comprehending and tackling this problem in the context of LLMs presents distinct challenges(Zhang et al., 2023b). Existing research has demonstrated that incorporating up-to-date, relevant knowledge in the prompt can effectively reduce fact-conflicting hallucination (Vu et al., 2023; Lewis et al., 2021). This approach, referred to as *Retrieval-Augmented Generation* (RAG), is widely used in real-world LLM applications. For instance, Google Bard [1] and Microsoft BingChat [2] have implemented this technique.

2.2 Hallucination Evaluation Datasets

Extensive research has focused on hallucination benchmarks within conventional Natural Language Generation settings (Dziri et al., 2022; Zhong et al., 2021; Durmus et al., 2020; Lin et al., 2022). With the rise of LLMs, the detection of hallucinations has become increasingly challenging, necessitating the development of high-quality datasets for LLM evaluation (Chen and Shu, 2023). Contributions in this domain include HaluEval (Li et al., 2023), which introduced datasets encompassing both synthetically and naturally generated LLM responses, and FELM (Chen et al., 2023), which concentrated on naturally generated LLM responses across multiple domain tasks. RefChecker (Hu et al., 2023), a

[1] https://bard.google.com

[2] https://www.bing.com

distinctive approach, breaks down claims in LLM responses into triples and utilizes human annotation to assess the truthfulness of facts. Notably, these works primarily focus on annotating factual hallucinations in LLM responses. Distinguishing from previous research, our work centers on the evaluation of LLMs within RAG settings.

2.3 Hallucination Detection Methods

Researchers have been exploring various methods to enhance the reliability of LLMs by detecting hallucinations. In Azaria and Mitchell (2023); Xiao and Wang (2021); Malinin and Gales (2021), intrinsic model uncertainty metrics such as token-level probability and entropy are used to detect hallucinations. When direct access to output uncertainty is not feasible, as in the case with limited APIs like GPT-4, an alternative approach involves employing a fully accessible LLM as a proxy (Manakul et al., 2023). In Falke et al. (2019); Barrantes et al. (2020), natural language inference modules are adapted to check the information consistency between the articles and their summaries, and it has been shown that external knowledge is helpful for detecting factual hallucinations. (Guo et al., 2022; Mallen et al., 2022). Additionally, methods that leverage the inherent capabilities of LLMs have been proposed for self-checking, such as verbalization-based and consistency-based methods (Xiong et al., 2023; Manakul et al., 2023). These techniques aim to detect hallucinations without relying on internal states or external data and tools.

3 Construction Process of RAGTruth

We established a data generation and annotation pipeline as shown in Figure 1.

3.1 Hallucination Taxonomy

Different from open-end generation, under RAG setting, the prompt contains rich context information, and the model is generally required to generate text based on the provided context. The detection and mitigation of inconsistencies between retrieved information and responses emerge as significant sources of hallucination.

As outlined below, we categorize the hallucination in the RAG setting into four types. For concrete examples of each type, please refer to Appendix A.

Evident Conflict: for when generative content presents direct contraction or opposition to the provided information. These conflicts are easily verifiable without extensive context, often involving clear factual errors, misspelled names, incorrect numbers, etc.

Subtle Conflict: for when generative content presents a departure or divergence from the provided information, altering the intended contextual meaning. These conflicts often involve substitution of terms that carry different implications or severity, requiring a deeper understanding of their contextual applications.

Evident Introduction of Baseless Information: for when generated content includes information not substantiated in the provided information. It involves the creation of hypothetical, fabricated, or hallucinatory details lacking evidence or support.

Subtle Introduction of Baseless Information: is when generated content extends beyond the provided information by incorporating inferred details, insights, or sentiments. This additional information lacks verifiability and might include subjective assumptions or commonly observed norms rather than explicit facts.

3.2 Response Generation

Tasks and Data Sources We selected three widely recognized generation tasks with RAG settings for response generation: Question Answering, Data-to-text Writing, and News Summarization.

For the task of question answering, we conducted a random sampling from the training set of MS MARCO (Nguyen et al., 2016). To reduce the difficulty of annotation, we selected only those questions related to daily life, and preserved only three retrieved passages for each question. Then we prompted LLMs to generate answers for each question solely based on the retrieved passages.

For the data-to-text writing task, we prompted LLMs to generate an objective overview for a randomly sampled business in the restaurant and nightlife categories from the Yelp Open Dataset (Yelp, 2021). In this dataset, information pertaining to a business is represented using structured data. To streamline the annotation process, we focused only on the following business information fields: *BusinessParking*, *RestaurantsReservations*, *OutdoorSeating*, *WiFi*, *RestaurantsTakeOut*, *RestaurantsGoodForGroups*, *Music*, and *Ambience*. In addition to the structured data, we have also included up to three business-related user reviews

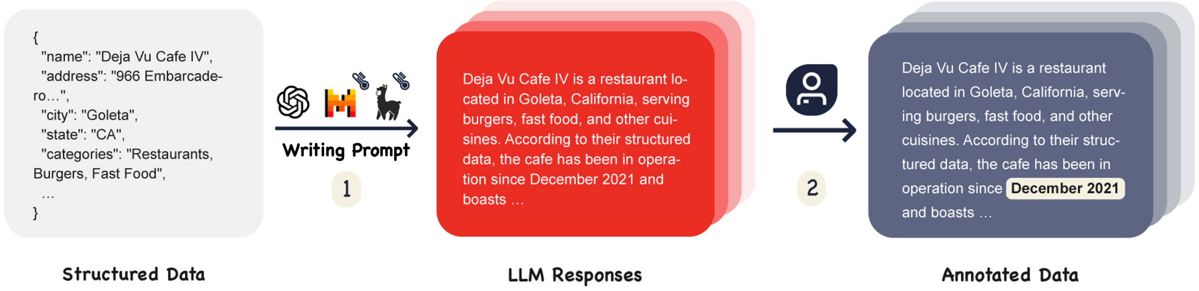

{
"name": "Deja Vu Cafe IV",
"address": "966 Embarcade-ro…",
"city": "Goleta",
"state": "CA",
"categories": "Restaurants, Burgers, Fast Food",
…
}
Structured Data
Writing Prompt
1
Deja Vu Cafe IV is a restaurant located in Goleta, California, serving burgers, fast food, and other cuisines. According to their structured data, the cafe has been in operation since December 2021 and boasts …
LLM Responses
2
Deja Vu Cafe IV is a restaurant located in Goleta, California, serving burgers, fast food, and other cuisines. According to their structured data, the cafe has been in operation since December 2021 and boasts …
Annotated Data

Figure 1: Data gathering pipeline. Taking a data-to-text writing task as an example, our data gathering pipeline includes 2 steps: 1) response generation. We generated responses with multiple LLMs and natural prompts. 2) human annotation. Human labeler annotated hallucinated spans in LLM responses.

to enrich the context information. In the prompt, these information is represented in JSON format.

For the news summarization task, we randomly selected documents from the training set of the well-known CNN/Daily Mail dataset (See et al., 2017) as well as recent news articles from a prestigious news platform. LLMs were prompted to generate a summary for each of the source news.

Models The following six models with strong instruction-following ability are used for response generation: GPT-3.5-turbo-0613 and GPT-4-0613 from OpenAI (OpenAI, 2023); Mistral-7b-Instruct from Mistral AI (Jiang et al., 2023); Llama-2-7B-chat, Llama-2-13B-chat and Llama-2-70B-chat (4bit quantized)[3] from Meta (Touvron et al., 2023). To ensure a fair comparison, the prompts used for response generation are kept straightforward with subtle differences among various models to optimize their performance. We provide detailed prompts in the Appendix B.

For each sample, we collected one response from each model. As a result, we got a total of 6 responses for each input sample.

3.3 Human Annotation

Identifying AI-generated hallucinations is a challenging task. It requires a strong capacity for critical thinking to understand the logical flow of various texts, along with meticulous attention to detail for spotting subtle inaccuracies and inconsistencies. Moreover, a certain level of media literacy and knowledge of current affairs is crucial to grasp the subjects discussed in news-related sample data. Therefore, we chose annotators who are proficient in English and possess a bachelor's degree in English, Communications, or relevant fields to ensure the accuracy and reliability of the annotation results. We recruited annotators from a professional vendor and paid them at a rate of $25 per hour per individual.

The annotators are invited to perform annotation tasks using Label Studio (Tkachenko et al., 2020-2022). Each labeling task is presented within one page, comprising the following components: 1) the context provided to the AI models; 2) a set of 6 responses, generated by different AI models. Our annotation interface is available in Appendix C.

Their task was to annotate the specific spans of the generated text that contains hallucinated information and categorize them into the four types. To ensure the quality of the annotations, each response is independently labeled by two annotators. The consistency rate of two annotators was 91.8% at the response level and 78.8% at the span level. In cases where there is a considerable difference between the two annotations, a third review is undertaken.

3.4 Annotations for Adaptive Evaluation

In different contexts, the definition and criteria for hallucination vary, and the annotation of hallucination is not always straightforward. In contentious cases, additional annotations are provided to accurately reflect these situations. This approach enables users to adopt various evaluation strategies tailored to their specific application circumstances. Please refer to Appendix C for more statistical information about these annotations.

Implicit Truth The extensive world knowledge and ability of LLMs is a significant advantage in open-ended generation scenarios. But in the context of this paper, which focuses on the relatively strict RAG scenarios, we have labeled information that is not mentioned in the reference but may be truthful as hallucinations. For instance, mentioning

[3] https://huggingface.co/TheBloke/Llama-2-70B-Chat-AWQ

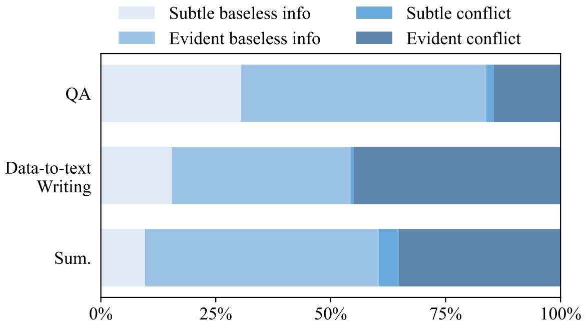


Figure 2: Frequency of different types of hallucination by task.

a local officer's name not present in the reference or claiming that a restaurant accepts credit card payments without any basis.

The decision is based on the observation that LLMs have a relatively high chance of making errors when generating detailed facts, partly because their embedded knowledge can be outdated. Therefore, RAG applications usually instruct LLMs not to generate factual content without the support of references. Besides, we provided an additional span-level annotation named *implicit_true* for these spans to accommodate different application needs.

Differences in Handling Null Value In the data-to-text writing task, certain fields sometimes are with null values. We observed that in the generated results, null is often interpreted as false by some models. Since the more common expressions for negation in our dataset are the boolean value *False* or the text *No*, we labeled these instances as hallucinations (evident introduction of baseless info) and provided a special span-level annotation named *due_to_null* for these spans. In the subsequent hallucination detection experiments, our prompts will be aligned with this standard.

4 Hallucination Benchmark Analysis

4.1 Basic Statistics

We presented detailed statistics of RAGTruth in Table 2. Compared to existing datasets for hallucination detection (Cao et al., 2023; Kamoi et al., 2023), the RAGTruth dataset is considerably large in scale. The corpus contains a total of 2,965 instances of data, which include 989 instances for question answering, 1,033 instances for date-to-text writing, and 943 instances for news summarization. Each instance comprises responses from 6 different models. As shown in Table 2, the RAGTruth dataset also features longer prompt and response lengths than existing datasets for hallucination detection (Wang et al., 2020).

4.2 Hallucination Statistics

Hallucination Types As shown in Figure 2, the generation of information baseless in the context was significantly more prevalent than the generation of information conflicting with the context, especially for the question answering tasks. Within the two major categories of *baseless info* and *conflict*, the more severe hallucinations, namely *Evident baseless info* and *Evident conflict*, respectively, account for a significant portion. This observation highlights the importance and challenges of LLMs hallucination mitigation, even in RAG settings.

Hallucination vs Tasks As shown in Table 2, across the three tasks, the date-to-text writing task exhibited the highest frequency of hallucinations in its responses. Inconsistent handling of JSON format data, especially time and attributes, contributed to a significant number of hallucinations in this task. Interestingly, the models did not show a higher rate of hallucinations for recent news compared to outdated news. This could be attributed to the shorter context length in the recent news subtask compared to the CNN/DM subtask.

Hallucination vs Models Table 3 illustrates that among the data we collected, OpenAI's two models demonstrated notably lower hallucination rates compared to others. Specifically, GPT-4-0613 exhibited the lowest hallucination frequency.

To more clearly compare the hallucination rate of different models, we calculated the hallucination density for each model across three tasks. Hallucination density is defined as the average number of hallucination spans per hundred words in the responses. In the Llama2 series, a clear negative correlation was observed between the model scale and hallucination density, aside from the data-to-text writing tasks. Despite its strong performance in various benchmarks and leaderboards (Zheng et al., 2023), the Mistral-7B-Instruct model generated the highest number of responses containing hallucinations.

Hallucination vs Length After removing the top and bottom 5% of outliers, we partitioned the data for each task type into three equal-sized groups according to the length of the context/response. We then computed the average number of hallucinated spans per response within each group. As shown

Task	# Instance	# Resp.	Context Length		Resp. Length		Hallucination		
			Mean	Max	Mean	Max	# Resp.	% Resp.	# Span
Question Answering	989	5934	243	509	119	381	1724	29.1%	2927
Data-to-text Writing	1033	6198	354	1253	159	369	4254	68.6%	9290
Summarization(CNN/DM)	628	3768	648	1749	124	632	1165	30.9%	1474
Summarization(Recent News)	315	1890	369	481	89	240	521	27.6%	598
Overall	2965	17790	381	1749	131	632	7664	43.1%	14289

Table 2: The basic statistics of RAGTruth. Here "Resp." stands for "Response".

Model	Question Answering			Data-to-text Writing			Summarization			Overall	
	# Resp.	# Span	Density	# Resp.	# Span	Density	# Resp.	# Span	Density	# Resp.	# Span
GPT-3.5-turbo-0613	75	89	0.12	272	384	0.18	54	60	0.05	401	533
GPT-4-0613	48	51	0.06	290	354	0.27	74	80	0.08	406	485
Llama-2-7B-chat	510	1010	0.59	888	1775	1.27	434	517	0.58	1832	3302
Llama-2-13B-chat	399	654	0.48	983	2803	1.53	295	342	0.41	1677	3799
Llama-2-70B-chat†	320	529	0.40	863	1834	1.15	212	245	0.26	1395	2608
Mistral-7B-Instruct	378	594	0.59	958	2140	1.51	617	828	0.86	1953	3562

Table 3: Hallucination counts and density of models. †: We used 4-bit quantized version of Llama-2-70B-chat.

CLB	Summarization	D2T Writing	QA
1	$0.29_{(176,368]}$	$1.51_{(178,273]}$	$0.50_{(131,187]}$
2	$0.36_{(368,587]}$	$1.48_{(273,378]}$	$0.51_{(187,288]}$
3	$0.44_{(587,1422]}$	$1.49_{(378,731]}$	$0.49_{(288,400]}$
RLB	**Summarization**	**D2T Writing**	**QA**
1	$0.34_{(44,87]}$	$1.20_{(93,131]}$	$0.21_{(19,93]}$
2	$0.32_{(87,119]}$	$1.59_{(131,175]}$	$0.37_{(93,138]}$
3	$0.44_{(119,245]}$	$1.69_{(175,258]}$	$0.87_{(138,257]}$

Table 4: Average number of hallucinations per response in different context length buckets (CLB) and response length buckets (RLB) for the three types of tasks. The subscript denotes the minimum and maximum length of this bucket.

in Table 4, there is a clear overall trend of an increase in the average number of hallucinations as the response length grows. Only the average number of hallucinations in news summarization tasks significantly increases with the length of the context. This may be because the contexts in the other two tasks are more structured, and an increase in length does not significantly raise the difficulty of understanding the content.

Location of Hallucinations In Figure 3, we present the heatmap of the hallucination occurrence positions. Hallucinations are significantly more likely to occur towards the end of responses in question-answering and news summarization tasks. Compared to other tasks, the data-to-text writing task has a relatively higher occurrence of hallucinations in the first half. In that bright area, hallucinations concerning business attributes frequently occur.

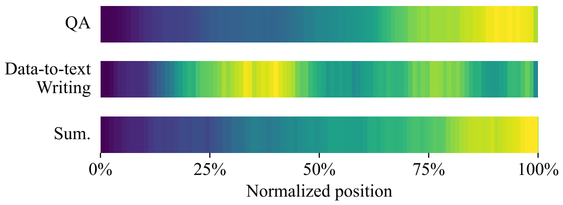


Figure 3: Heatmaps of normalized hallucination occurrence positions. The probability of hallucinations occurring is higher in brighter areas.

5 Experimental Setup

5.1 Hallucination Detection Algorithms

Using RAGTruth, we conducted experiments with the following four distinct algorithms for hallucination detection:

Hallucination Detection Prompt: Hallucination detection prompts are manually crafted to instruct LLMs (GPT-4-turbo and GPT-3.5-turbo) in assessing whether a given reference-response pair contains hallucinated content and to identify the corresponding hallucinated spans in the response. For detailed information about these prompts, please refer to Appendix D.

SelfCheckGPT (Manakul et al., 2023): SelfCheckGPT employs a zero-resource, sampling-based method to fact-check the responses of black-box models. When processing each response in RAGTruth, 3 extra responses from the same model were sampled and served as references, and GPT-3.5-turbo was used to verify consistency. We detected hallucinations sentence-by-sentence within a response, and then aggregated these results to

Methods	Question Answering			Data-to-text Writing			Summarization			Overall		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Prompt$_{\text{gpt-3.5-turbo}}$	18.8	84.4	30.8	65.1	95.5	77.4	23.4	89.2	37.1	37.1	92.3	52.9
Prompt$_{\text{gpt-4-turbo}}$	33.2	90.6	45.6	64.3	**100.0**	78.3	31.5	97.6	47.6	46.9	97.9	63.4
SelfCheckGPT$_{\text{gpt-3.5-turbo}}$	35.0	58.0	43.7	68.2	82.8	74.8	31.1	56.5	40.1	49.7	71.9	58.8
LMvLM$_{\text{gpt-4-turbo}}$	18.7	76.9	30.1	68.0	76.7	72.1	23.3	81.9	36.2	36.2	77.8	49.4
Finetuned Llama-2-13B	**61.6**	76.3	**68.2**	**85.4**	91.0	**88.1**	**64.0**	54.9	**59.1**	**76.9**	80.7	**78.7**

Table 5: The response-level hallucination detection performance for each baseline method across different tasks and different models.

Methods	Question Answering			Data-to-text Writing			Summarization			Overall		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Prompt Baseline$_{\text{gpt-3.5-turbo}}$	7.9	25.1	12.1	8.7	45.1	14.6	6.1	33.7	10.3	7.8	35.3	12.8
Prompt Baseline$_{\text{gpt-4-turbo}}$	23.7	52.0	32.6	17.9	**66.4**	28.2	14.7	**65.4**	24.1	18.4	**60.9**	28.3
Finetuned Llama-2-13B	**55.8**	**60.8**	**58.2**	**56.5**	50.7	**53.5**	**52.4**	30.8	**38.8**	**55.6**	50.2	**52.7**

Table 6: The span-level detection performance for each baseline method across different tasks and different models.

provide a response-level detection outcome.

LMvLM (Cohen et al., 2023): LMvLM is an approach that employs a multi-turn interaction between two Language Models that aim to discover inconsistencies through cross-examination.

LLM Finetuning: Llama-2-13B has been fine-tuned using the training set from RAGTruth. The model takes the context-response pair with proper instructions as the input and treats the hallucinate span as the targeted generation output. We employed full training with an initial learning rate of 2e-5, and limiting the training to 1 epochs, all conducted on 4 A100 GPUs.

5.2 Data Split

All detection algorithms are tested on the same RAGTruth test set, which consists of 450 instances in total, derived by randomly selecting 150 instances from each task type. The rest of the data is used to fine-tune the LLama-2-13B model, as previously mentioned.

5.3 Evaluation Metrics

It is a more challenging and significant task to identify the locations of hallucinations within the response than only determining whether a response contains hallucinations. We assess hallucination detection at both the response and span levels.

Response-level Detection We report precision, recall, and F1 score for each detection algorithm and its variants across different tasks.

Span-level Detection We calculate the overlap between the detected span and human-labeled span and report the precision, recall, and f1 score at the char-level.

6 Experimental Results

6.1 Response-level Detection

The results in Table 5 reveal that hallucination detection remains a significant challenge in the context of RAG for all existing detection methods. Even when reference information is available, the responses generated may still include hallucinations, which current LLMs cannot reliably identify. The most advanced LLM, GPT-4-turbo, achieves only an average F1 score of 63.4%. For another notable baseline, SelfCheckGPT also shows unsatisfactory performance in this regard, achieving an average F1 score of 58.8% with GPT-3.5-turbo.

By utilizing our high-quality training set, a fine-tuned Llama-2-13B can achieve the best performance with an average 78.7% f1 score. This shows the effectiveness of our data in improving the model's hallucination detection ability.

6.2 Span-level Detection

RAGTruth, as a hallucination corpus with fine-grained span labels, enables us to present experimental results for span-level detection, serving as a baseline for future research. As shown in Table 6, the overall performance of the current detection method is sub-optimal, highlighting the challenges in span-level detection. Even the advanced GPT-4-turbo tends to incorrectly classify many non-hallucinated contents with a low precision of 18.4%. While our fine-tuned model shows improved capability in identifying hallucinated spans by achieving an averaged f1 score of 52.7%, it still falls short of perfect detection, emphasizing the inherent difficulties of this task.

We also report the detection performance across

Group	Selection Strategy	Valid Response Num	Hallucination Rate
Llama-2-7B-chat (51.8) Mistral-7B-Instruct (57.6)	Random	450	52.4(-)
	Select the response with fewer detected hallucination spans	450	41.1(↓**21.6%**)
	Select the response with no detected hallucination spans	328†	19.3(↓**63.2%**)
GPT-3.5-Turbo-0613 (10.9) GPT-4-0613 (9.3)	Random	450	9.8(-)
	Select the response with fewer detected hallucination spans	450	5.6(↓**42.9%**)
	Select the response with no detected hallucination spans	448†	4.8(↓**51.0%**)

Table 7: Utilizing the finetuned hallucination detector to sample from two responses can significantly reduce the rate of hallucinations. The numbers within the brackets in the group column represent the model's hallucination rate. †: Some instances did not have responses that met the required criteria.

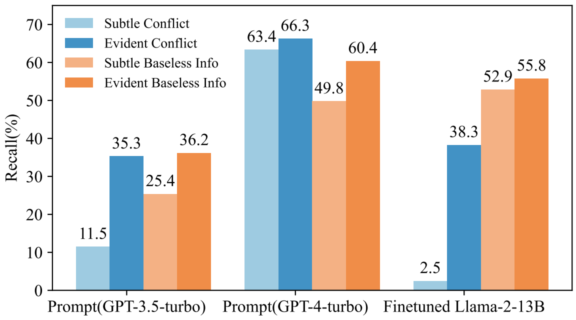


Figure 4: The span-level recalls of different models on four types of hallucinations.

four different types of hallucination spans. In the current stage, as we have not differentiated the types of detected hallucinations, we only report the char-level recall for different types of hallucinations. As indicated in Figure 5, the detection of evident hallucinations proves more effective compared to that of subtle hallucinations.

6.3 Hallucination Suppression

We tested the effectiveness of hallucination suppression using our finetuned hallucination detection model. For the 450 instances in the test set, we employed two strategies to select a final output from two responses generated by two different models with similar hallucination densities. The first strategy involved selecting the response with fewer predicted hallucination spans. The second strategy, more stringent, mandated that the selected response have no detected hallucination spans. When the number of hallucination spans detected in both candidate responses is the same, one will be chosen at random. Due to limited response candidates, not all instances have a response that conforms to the second strategy. In practical scenarios, this issue can be addressed by increasing the number of candidate responses. We employed random selection as a simple baseline for comparison.

The results shown in Table 7 indicate that with the help of the hallucination detector, both strategies can significantly reduce the hallucination rate. For the relatively small Llama-2-7B-chat and Mistral-7B-Instruct models, compared to random selection, the first strategy reduced the hallucination rate by 21.6%, while the second strategy achieved a reduction of 63.2%. Even for models with a low hallucination rate, specifically GPT-3.5-Turbo and GPT-4, employing the finetuned hallucination detector for sampling can still further reduce the rate of hallucinations. The two strategies yielded a reduction in hallucination rates of 42.9% and 51.0%, respectively. These results demonstrate the potential of an efficient hallucination detection model in developing trustworthy RAG LLMs.

7 Conclusion

In this paper, we introduce RAGTruth, a large-scale corpus of naturally generated hallucinations, featuring detailed word-level annotations tailored for RAG scenarios. Our work includes an in-depth analysis of the interplay between hallucinations and various factors, such as task types, models being used, and contextual settings.

Additionally, we conduct empirical benchmarks of several hallucination detection approaches using our corpus. We show that fine-tuning Llama with RAGTruth leads to competitive performance. This implies that by using a high-quality dataset such as RAGTruth, it is possible to develop specialized hallucination detection models that are highly effective when compared to prompt-based methods using general models such as GPT-4.

Simultaneously, our findings reveal that identifying hallucinations in RAG contexts, particularly at the span level, remains a formidable challenge, with current methods still falling short of reliable detection. We hope that RAGTruth, can assist the development of hallucination detection techniques for retrieval augmented generation.

8 Limitations

The study of hallucination in large language models is a rapidly advancing field, characterized by the continuous evolution of application scenarios, sources of hallucination, and techniques for detecting and preventing them. While our work represents the first attempt to benchmark hallucination in the RAG setting, there may be situations not addressed by this research that are nonetheless significant for certain practical applications.

9 Ethical considerations

This work is in full compliance with the Ethics Policy of the ACL. We acknowledge that responses generated by LLMs in this study may contain inaccuracies. Aside from this, to the best of our knowledge, there are no additional ethical issues associated with this paper.

References

Ayush Agrawal, Mirac Suzgun, Lester Mackey, and Adam Tauman Kalai. 2023. Do language models know when they're hallucinating references?

Amos Azaria and Tom Mitchell. 2023. The Internal State of an LLM Knows When its Lying.

Mario Barrantes, Benedikt Herudek, and Richard Wang. 2020. Adversarial nli for factual correctness in text summarisation models. *arXiv preprint arXiv:2005.11739*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Zouying Cao, Yifei Yang, and Hai Zhao. 2023. Autohall: Automated hallucination dataset generation for large language models. *ArXiv*, abs/2310.00259.

Canyu Chen and Kai Shu. 2023. Can llm-generated misinformation be detected? *arXiv preprint arXiv:2309.13788*.

Shiqi Chen, Yiran Zhao, Jinghan Zhang, I-Chun Chern, Siyang Gao, Pengfei Liu, and Junxian He. 2023. Felm: Benchmarking factuality evaluation of large language models.

I-Chun Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, and Pengfei Liu. 2023. Factool: Factuality detection in generative ai – a tool augmented framework for multi-task and multi-domain scenarios.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. Lm vs lm: Detecting factual errors via cross examination.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2023. Chain-of-verification reduces hallucination in large language models.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. 2022. Evaluating attribution in dialogue systems: The begin benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083.

Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. 2019. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy. Association for Computational Linguistics.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks.

Zhijiang Guo, Michael Schlichtkrull, and Andreas Vlachos. 2022. A survey on automated fact-checking.

Xiangkun Hu, Dongyu Ru, Qipeng Guo, Lin Qiu, and Zheng Zhang. 2023. Refchecker for fine-grained hallucination detection.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. 2023. Challenges and applications of large language models.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. Wice: Real-world entailment for claims in wikipedia. In *Conference on Empirical Methods in Natural Language Processing*.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality. *arXiv preprint arXiv:2302.14520*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models.

Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2022. Pretrained language models for text generation: A survey. *arXiv preprint arXiv:2201.05273*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Alisa Liu and Jiacheng Liu. 2023. The memotrap dataset. `https://github.com/liujch1998/memo-trap`.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, Hannaneh, and Hajishirzi. 2022. When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2023. Med-halt: Medical domain hallucination test for large language models.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A Survey of Hallucination in Large Foundation Models.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Jiaming Shen, Jialu Liu, Dan Finnie, Negar Rahmati, Michael Bendersky, and Marc Najork. 2023. " why is this misleading?": Detecting news headline hallucinations with explanations. *arXiv preprint arXiv:2302.05852*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2022. Label Studio: Data labeling software. Open source software available from https://github.com/heartexlabs/label-studio.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Neeraj Varshney, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. 2023. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. 2023. Freshllms: Refreshing large language models with search engine augmentation.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries.

Yijun Xiao and William Yang Wang. 2021. On hallucination and predictive uncertainty in conditional language generation.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms.

Yelp. 2021. Yelp open dataset. https://www.yelp.com/dataset. Accessed: 2023-11-03.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi R Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. 2023a. R-tuning: Teaching large language models to refuse unknown questions. *arXiv preprint arXiv:2311.09677*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, Shuming Shi, and Tencent AI Lab. 2023b. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921, Online. Association for Computational Linguistics.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

A Hallucination Examples

EVIDENT CONFLICT
Example 1 Original content: They've sent about 500 of the most seriously injured people for treatment in Kathmandu already – the majority by road. Generative content: The team airlifted 500 injured people… to Kathmandu for medical treatment. **Example 2** Original content: "RestaurantsReservations": false Generative content: The restaurant offers outdoor seating and takes reservations
SUBTLE CONFLICT
Example 1 Original content: This is the section that lets the government deny trademark protection to a mark that is "immoral," "scandalous" or "disparaging." Generative content: …violates Section 2(a) of the trademark act, which prohibits marks that are "immoral, scandalous, or disqualified." **Example 2** Original content: It was quite a hassle to find a table since this place is crowded all the time, but well, nothing can stand between me and that absolutely delightful lobster roll Generative content: The restaurant is located in a crowded area, but the delicious lobster rolls make it worth the hassle to find a table.
EVIDENT INTRODUCTION OF BASELESS INFORMATION
Example 1 Original content: The couple divorced in 1968… She had no contact with the surviving members of the band until meeting up at the 2006 Las Vegas premiere of "The Beatles Love." Generative content: The couple divorced in 1968, but later reconciled and attended the 2006 Las Vegas premiere of "The Beatles Love." **Example 2** Original content: "RestaurantsTakeOut": true Generative content: Customers can also enjoy takeout and delivery options.
SUBTLE INTRODUCTION OF BASELESS INFORMATION
Example 1 Original content: Amazing happy hour and great people. Generative content: One of the standout features of Killer Shrimp is their happy hour, which is available daily from 4 pm to 7 pm. **Example 2** Original content: Place mashed potatoes in a medium bowl. Stir in onions, cheese, salt, pepper, and bacon. Generative content: Place your mashed potatoes in a medium bowl. Stir in cooked onions, cheese, salt, pepper, and bacon.

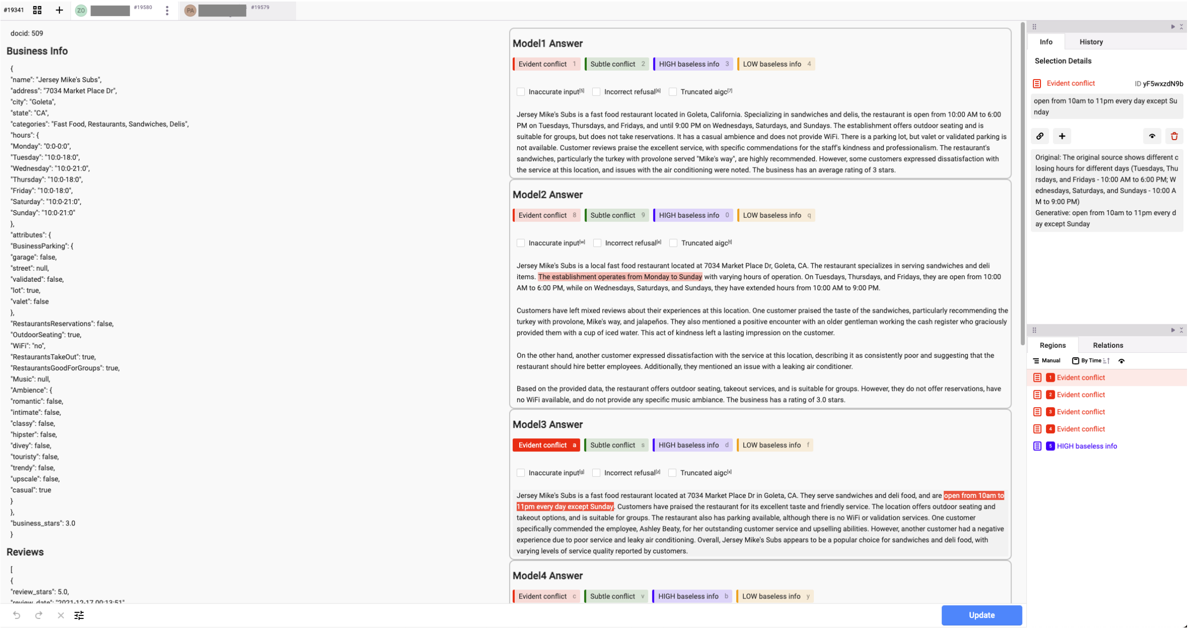
Table 8: Examples of the four types of hallucinations.

B Response Generation Prompts

QUESTION ANSWERING
Answer the following question: {question} Bear in mind that your response should be strictly based on the following 3 passages: {passages} In case the passages do not contain the necessary information to answer the question, please reply with: "Unable to answer based on given passages."
DATA-TO-TEXT WRITING
Instruction: Write an objective overview about the following local business based only on the provided structured data in the JSON format. You should include details and cover the information mentioned in the customers' review. The overview should be 100 - 200 words. Don't make up information. Structured data: {json_data} Overview:
SUMMARIZATION
Summarize the following news within {word_num} words: {news} output:

Table 9: Prompts for generating responses for the three types of tasks. word_num is min(200, word_num_of_news//4). The word count requirement is only to control the length of the generated summarization, it will not serve as the basis for hallucination annotation.

C Annotation Details



#19341 ZO [redacted] #19580 PA [redacted] #19579

docid: 509

Business Info

```
{
"name": "Jersey Mike's Subs",
"address": "7034 Market Place Dr",
"city": "Goleta",
"state": "CA",
"categories": "Fast Food, Restaurants, Sandwiches, Delis",
"hours": {
"Monday": "0:0-0:0",
"Tuesday": "10:0-18:0",
"Wednesday": "10:0-21:0",
"Thursday": "10:0-18:0",
"Friday": "10:0-18:0",
"Saturday": "10:0-21:0",
"Sunday": "10:0-21:0"
},
"attributes": {
"BusinessParking": {
"garage": false,
"street": null,
"validated": false,
"lot": true,
"valet": false
},
"RestaurantsReservations": false,
"OutdoorSeating": true,
"WiFi": "no",
"RestaurantsTakeOut": true,
"RestaurantsGoodForGroups": true,
"Music": null,
"Ambience": {
"romantic": false,
"intimate": false,
"classy": false,
"hipster": false,
"divey": false,
"touristy": false,
"trendy": false,
"upscale": false,
"casual": true
}
},
"business_stars": 3.0
}
```

Reviews

```
[
{
"review_stars": 5.0,
"review_date": "2021-12-17 00:13:51"
```

Model1 Answer

Evident conflict 1 | Subtle conflict 2 | HIGH baseless info 3 | LOW baseless info 4

□ Inaccurate input[5] □ Incorrect refusal[6] □ Truncated aigc[7]

Jersey Mike's Subs is a fast food restaurant located in Goleta, California. Specializing in sandwiches and delis, the restaurant is open from 10:00 AM to 6:00 PM on Tuesdays, Thursdays, and Fridays, and until 9:00 PM on Wednesdays, Saturdays, and Sundays. The establishment offers outdoor seating and is suitable for groups, but does not take reservations. It has a casual ambience and does not provide WiFi. There is a parking lot, but valet or validated parking is not available. Customer reviews praise the excellent service, with specific commendations for the staff's kindness and professionalism. The restaurant's sandwiches, particularly the turkey with provolone served "Mike's way", are highly recommended. However, some customers expressed dissatisfaction with the service at this location, and issues with the air conditioning were noted. The business has an average rating of 3 stars.

Model2 Answer

Evident conflict 8 | Subtle conflict 9 | HIGH baseless info 0 | LOW baseless info q

□ Inaccurate input[w] □ Incorrect refusal[e] □ Truncated aigc[t]

Jersey Mike's Subs is a local fast food restaurant located at 7034 Market Place Dr, Goleta, CA. The restaurant specializes in serving sandwiches and deli items. The establishment operates from Monday to Sunday with varying hours of operation. On Tuesdays, Thursdays, and Fridays, they are open from 10:00 AM to 6:00 PM, while on Wednesdays, Saturdays, and Sundays, they have extended hours from 10:00 AM to 9:00 PM.

Customers have left mixed reviews about their experiences at this location. One customer praised the taste of the sandwiches, particularly recommending the turkey with provolone, Mike's way, and jalapeños. They also mentioned a positive encounter with an older gentleman working the cash register who graciously provided them with a cup of iced water. This act of kindness left a lasting impression on the customer.

On the other hand, another customer expressed dissatisfaction with the service at this location, describing it as consistently poor and suggesting that the restaurant should hire better employees. Additionally, they mentioned an issue with a leaking air conditioner.

Based on the provided data, the restaurant offers outdoor seating, takeout services, and is suitable for groups. However, they do not offer reservations, have no WiFi available, and do not provide any specific music ambiance. The business has a rating of 3.0 stars.

Model3 Answer

Evident conflict a | Subtle conflict s | HIGH baseless info d | LOW baseless info f

□ Inaccurate input[g] □ Incorrect refusal[z] □ Truncated aigc[x]

Jersey Mike's Subs is a fast food restaurant located at 7034 Market Place Dr in Goleta, CA. They serve sandwiches and deli food, and are open from 10am to 11pm every day except Sunday. Customers have praised the restaurant for its excellent taste and friendly service. The location offers outdoor seating and takeout options, and is suitable for groups. The restaurant also has parking available, although there is no WiFi or validation services. One customer specifically commended the employee, Ashley Beaty, for her outstanding customer service and upselling abilities. However, another customer had a negative experience due to poor service and leaky air conditioning. Overall, Jersey Mike's Subs appears to be a popular choice for sandwiches and deli food, with varying levels of service quality reported by customers.

Model4 Answer

Evident conflict c | Subtle conflict v | HIGH baseless info b | LOW baseless info y

Info | History

Selection Details

Evident conflict ID yF5wxzdN9b

open from 10am to 11pm every day except Su nday

Original: The original source shows different c losing hours for different days (Tuesdays, Thu rsdays, and Fridays - 10:00 AM to 6:00 PM; W ednesdays, Saturdays, and Sundays - 10:00 A M to 9:00 PM)
Generative: open from 10am to 11pm every d ay except Sunday

Regions | Relations

Manual | By Time

1 Evident conflict
2 Evident conflict
3 Evident conflict
4 Evident conflict
5 HIGH baseless info

Figure 5: Annotation interface. For privacy reasons, we have masked the full names of the annotators in the screenshot.

Task	Model	# Hallucination Span	*implicit_true*		*due_to_null*	
			# Span	% Span	# Span	% Span
Question Answering	GPT-3.5-turbo-0613	89	33	0.371		
	GPT-4-0613	51	15	0.294		
	Llama-2-7B-chat	1010	251	0.249		
	Llama-2-13B-chat	654	215	0.329		
	Llama-2-70B-chat	529	168	0.318		
	Mistral-7B-Instruct	594	164	0.276		
Data-to-text Writing	GPT-3.5-turbo-0613	384	52	0.135	69	0.180
	GPT-4-0613	354	24	0.068	209	0.590
	Llama-2-7B-chat	1775	195	0.110	230	0.130
	Llama-2-13B-chat	2803	260	0.09	439	0.157
	Llama-2-70B-chat	1834	274	0.149	272	0.148
	Mistral-7B-Instruct	2140	102	0.048	423	0.198
Summarization	GPT-3.5-turbo-0613	60	14	0.233		
	GPT-4-0613	80	10	0.125		
	Llama-2-7B-chat	517	44	0.085		
	Llama-2-13B-chat	342	28	0.082		
	Llama-2-70B-chat	245	27	0.110		
	Mistral-7B-Instruct	828	52	0.063		
Overall		14289	1928	0.135	1642	0.115

Table 10: Detailed statistical information for the labels *implicit_true* and *due_to_null*. The majority of implicit truths appear in two types of tasks: question answering and data-to-text writing. About 17.7% hallucination spans in the data-to-text writing tasks are related to null values in the JSON data.

D Hallucination Detection Prompts

SUMMARIZATION
Below is the original news: {article} Below is a summary of the news: {summary} Your task is to determine whether the summary contains either or both of the following two types of hallucinations: 1. conflict: instances where the summary presents direct contraction or opposition to the original news; 2. baseless info: instances where the generated summary includes information which is not substantiated by or inferred from the original news. Then, compile the labeled hallucinated spans into a JSON dict, with a key "hallucination list" and its value is a list of hallucinated spans. If there exist potential hallucinations, the output should be in the following JSON format: {"hallucination list": [hallucination span1, hallucination span2, ...]}. Otherwise, leave the value as a empty list as following: {"hallucination list": []}. Output:
QUESTION ANSWERING
Below is a question: {question} Below are related passages: {passages} Below is an answer: {answer} Your task is to determine whether the answer contains either or both of the following two types of hallucinations: 1. conflict: instances where the answer presents direct contraction or opposition to the passages; 2. baseless info: instances where the answer includes information which is not substantiated by or inferred from the passages. Then, compile the labeled hallucinated spans into a JSON dict, with a key "hallucination list" and its value is a list of hallucinated spans. If there exist potential hallucinations, the output should be in the following JSON format: {"hallucination list": [hallucination span1, hallucination span2, ...]}. Otherwise, leave the value as a empty list as following: {"hallucination list": []}. Output:
DATA-TO-TEXT WRITING
Below is a structured data in the JSON format: {business info} Below is an overview article written in accordance with the structured data: {overview} Your task is to determine whether the overview contains either or both of the following two types of hallucinations: 1. conflict: instances where the overview presents direct contraction or opposition to the structured data; 2. baseless info: instances where the generated overview includes information which is not substantiated by or inferred from the structured data. In JSON, "null" or "None" represents an unknown value rather than a negation. Then, compile the labeled hallucinated spans into a JSON dict, with a key "hallucination list" and its value is a list of hallucinated spans. If there exist potential hallucinations, the output should be in the following JSON format: {"hallucination list": [hallucination span1, hallucination span2, ...]}. Otherwise, leave the value as a empty list as following: {"hallucination list": []}. Output:

Table 11: Prompts for detecting hallucination for the three types of tasks. In the prompt for data-to-text writing, we clarified that null or None in JSON should be treated as unknown rather than a negation.