Beyond Facts: Evaluating Intent Hallucination in Large Language Models

Anonymous ACL submission

Abstract

When exposed to complex queries containing multiple conditions, today's large language models (LLMs) tend to produce responses that only partially satisfy the query while neglecting certain conditions. We therefore introduce the concept of *Intent Hallucination*, a phenomenon where LLMs either omit (neglecting to address certain parts) or misinterpret (responding to invented query parts) elements of the given query, leading to intent hallucinated generation. To systematically evaluate intent hallucination, we introduce FAITHQA, a novel benchmark for intent hallucination that contains 20,068 problems, covering both query-only and retrieval-augmented generation (RAG) setups with varying topics and difficulty. FAITHQA is the first hallucination benchmark that goes beyond factual verification, tailored to identify the fundamental cause of intent hallucination. By evaluating various LLMs on FAITHQA, we find that (1) intent hallucination is a common issue even for state-of-the-art models, and (2) the phenomenon stems from omission or misinterpretation of LLMs. To facilitate future research, we introduce an automatic LLM generation evaluation metric, CONSTRAINT SCORE, for detecting intent hallucination. Human evaluation results demonstrate that CONSTRAINT SCORE is closer to human performance for intent hallucination compared to baselines.

1 Introduction

The generation ability of Large Language Models (LLMs) has been widely proven for various tasks (OpenAI et al., 2024; Dubey et al., 2024; Jiang et al., 2023). Nonetheless, evaluating their generation quality is accompanied by the challenge of hallucination (Ji et al., 2023; Huang et al., 2023). Specifically, when given a complex query containing multiple conditions as shown in Fig 1, LLMs' generation may deviate from the query, leading to an unsatisfied generation result. We term such a phenomenon as "**Intent Hallucination**", which has been largely overlooked in current research (Min et al., 2023; Hou et al., 2024; Manakul et al., 2023).

Unlike factual hallucination (Li et al., 2023; Cao et al., 2021), which can be directly detected through search-based fact-checking (Sellam et al., 2020; Min et al., 2023), evaluating intent hallucination is challenging. This is because complex queries often contain duplicate intents, and LLMs may satisfy only a portion of them, making dissatisfaction hard to detect or quantify. Furthermore, as LLMs continue to be advanced, users tend to provide these stronger LLMs with more and more complicated queries, which even for human beings could be hard to understand. It demonstrates the need for LLMs to be not only factually correct but intentionally correct.

Our paper aims to address two under-explored yet crucial questions: (1) *Why do LLMs tend to have Intent Hallucination?* and (2) *How can we detect Intent Hallucination?* Answering these questions is vital for LLM applications relying on both factual accuracy and accurately addressing queries.

For the first question, we propose that LLM's **omission** (*e.g.*, ignoring query components) or **misinterpretation** (*e.g.*, responding to invented query components) over word-level meaning is the fundamental cause of intent hallucination. To further investigate, we introduce FAITHQA, the first benchmark specifically designed to address intent hallucination's two key scenarios: omission and misinterpretation. FAITHQA consists of 20,068 queries for analysis. We conducted extensive human evaluations to ensure the quality of our benchmark. FAITHQA covers a wide range of topics with different difficulty, and has proven to be challenging even for state-of-the-art models. Our benchmark reveals that increasing query complexity correlates with a higher likelihood of intent hallucination.

To address the second question, we introduce CONSTRAINT SCORE, a new evaluation metric that

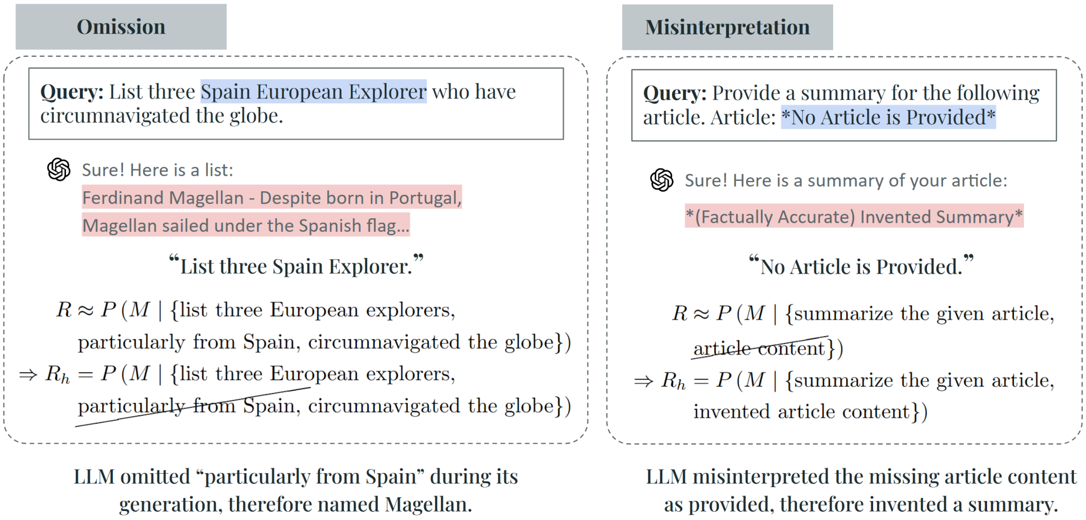


Figure 1: **Examples of two types of intent hallucination (omission and misinterpretation)**. For omission, GPT-4o omits "particularly from Spain", leading to factually accurate yet hallucinated outputs. For misinterpretation, GPT-4o misinterprets the missing article as provided, which leads to hallucinated outputs.

focuses on detecting intend hallucination. Our approach involves two major steps: (1) decomposing the query by concepts and actions then converting it into a series of short statements, each representing a specific requirement the generation must meet; and (2) assigning an importance-weighted binary label to each constraint, allowing a fine-grained evaluation. Our human evaluation shows that CONSTRAINT SCORE significantly out-performs LLM-as-the-judge (Manakul et al., 2023; Mishra et al., 2024; Sriramanan et al., 2024), as they tend to offer biased evaluation comparing with human score.

Taken together, our key contributions include:

- We propose the concept of intent hallucination beyond the existing factual hallucination.
- We developed FAITHQA, the first hallucination benchmark that focuses on the evaluation of intent hallucination. Our result shows that intent hallucination is a prevalent phenomenon even for state-of-the-art LLMs.
- We introduce CONSTRAINT SCORE, a novel evaluation metric to automatically assess LLM generation by breaking the query into intent constraints and computing a weighted score. Our analysis shows that CONSTRAINT SCORE significantly outperforms pure LLM grading baselines, which tend to be biased.

2 Related Works

Hallucinations in LLMs. In LLMs, "hallucination" refers to outputs that are nonfactual, irrelevant, or fabricated. This issue appears in tasks like question answering (Sellam et al., 2020), translation (Lee et al., 2018), summarization (Durmus et al., 2020), and dialogue (Balakrishnan et al., 2019), as noted in several studies (Ji et al., 2023; Azaria and Mitchell, 2023; Huang et al., 2023; Cao et al., 2021). To address this issue, many efforts have been made. Min et al. (2023) evaluate factual accuracy by checking core facts (atomic facts) in each sentence against reliable sources like Wikipedia. Hou et al. (2024) propose a hidden Markov tree model that breaks statements into premises and assigns a factuality score based on the probability of all parent premises. Manakul et al. (2023) detect hallucinations by sampling multiple responses and using self-consistency to identify discrepancies. Despite all these great efforts, limitations remain. Most work (1) focuses only on factual precision or in-context recall, overlooking the role of the query in generation (Li et al., 2023; Yang et al., 2023; Niu et al., 2024) (e.g., scoring both outputs equally in Fig 2), or (2) treats the query as a whole (Zhang et al., 2024a), resulting in coarse-grained evaluation.

Hallucination Benchmarks. Recent work on hallucination detection for LLMs includes HaluEval (Li et al., 2023) (synthetic and natural responses), FELM (Chen et al., 2023) (natural responses across domains), RAGTruth (Niu et al., 2024) (RAG hallucinations), and InfoBench (Qin et al., 2024) (instruction-following via query decomposition). These benchmarks mainly focus on factual hallucinations or require manual annotation. In contrast, FAITHQA is the first, to our knowledge, to assess non-factual hallucinations from a query-centric perspective. Despite discussing a similar topic, Zhang et al. (2024b)'s work is more focused on discovering intent hallucination's cause in training corpus perspective, while our paper is providing a comprehensive evaluation metric with an extensive benchmark to test with.

3 Preliminary

For a complex query containing multiple conditions, it has been reported that the model produces responses that only partially satisfy the conditions. To further investigate this, here we outline our two key insights for intent hallucination in this paper.

3.1 Intent Constraint: a Fundamental Unit

A query typically consists of multiple *concepts* and *actions*, each representing a distinct intent and carrying specific meaning within the given context. Refer to Fig 1, LLMs failed to address constraints provided in the query, leading to an intent hallucinated generation which deviates from the query.

To enable a fine-grained, query-centric evaluation, we introduce **Intent Constraint** – short statements that each express a single requirement for the generation to address (see examples in Fig 2). A query, defined by the concepts and actions within the context, can be broken down into these intent constraints, with each one representing a distinct concept or action. Addressing each of these constraints helps reduce the risk of hallucinated responses that misalign with the query's intent.

Definition 3.1 (*Intent Constraint Mapping Function*)**.** Let $\mathcal{Q}$ denote the set of all queries and $\mathcal{I}$ denote the set of all possible intent constraints. Both $q \in \mathcal{Q}$ and $c \in \mathcal{I}$ are text-based description. For any query $q \in \mathcal{Q}$, we define the intent constraint mapping function

$$C : \mathcal{Q} \to \mathcal{P}(\mathcal{I}),$$

such that

$$C(q) = C_m(q) \cup C_i(q) \cup C_o(q),$$

Where:

- $C_m(q) \subset \mathcal{I}$ is the set of *mandatory constraints* (constraints that must be addressed with the highest priority),
- $C_i(q) \subset \mathcal{I}$ is the set of *important constraints* (constraints that should be addressed after the mandatory ones), and
- $C_o(q) \subset \mathcal{I}$ is the set of *optional constraints* (constraints that are desirable but not essential).

This mapping ensures that the aggregated set $C(q)$ preserves the original meaning of the query q.

3.2 Intent Hallucination: Omission or Misinterpretation of Intent Constraints.

After establishing a fine-grained, query-centric perspective, we formally define intent hallucination as LLM's failure to address word-level concepts or actions, which expresses itself as an omission or misinterpretation of intent constraints. When LLMs either **omit** parts of the query (e.g., failing to address specific concepts or actions) or **misinterpret** it (e.g., responding to concepts or actions that were not mentioned), the generation fails to align with the original query, regardless of whether it is factually accurate.

Having intent constraint as the fundamental evaluation metric for intent hallucination is particularly important when dealing with complex, multi-condition queries. Under such cases, a language model might generate a response that only addresses the query partially while failing to address the rest. Evaluating the fulfillment of generation over intent constraint offers an approach to distinguish these nuance differences effectively.

Definition 3.2 (*Intent Hallucination*)**.** Let q be a query and P_θ be the LLM, with $y \sim P_\theta(\cdot \mid q)$ being the response. Using the intent constraint mapping function, the intend constraints extracted from q are $C(q)$. Ideally, we have

$$y \sim P_\theta(\cdot \mid q) \equiv P_\theta\Big(\cdot \mid C(q) = c_1, \dots, c_k\Big).$$

However, in practice, P_θ implicitly modifies $C(q)$ to an alternative constraint set $\hat{C}(q)$ (e.g., replacing c_i with c_i' or deleting some c_i), so that

$$y_h \sim P_\theta\Big(\cdot \mid \hat{C}(q)\Big).$$

This discrepancy between y_h and y is defined as *Intent Hallucination*.

4 Detecting Intent Hallucination

We introduce CONSTRAINT SCORE, a new evaluation metric to detect intent hallucination based on intent constraints. To operationalize the constraint mapping function $C(\cdot)$ defined earlier, we develop a multi-step process that systematically extracts and categorizes constraint set $C(q) = C_m(q) \cup C_i(q) \cup C_o(q)$ from queries. Our method has high flexibility, accommodating different queries involving RAG. The prompt template can be found in Appendix A.5.

4.1 Intent Constraint Mapping

Step 0: Preliminary assessment. The LLM first analyzes the query q to verify the presence of sufficient information for constraint extraction. This step is crucial for RAG queries to mitigate external content influence (Liu et al., 2023; Wu et al., 2024). If insufficient information is detected, the process halts and requests additional input, ensuring $C(q)$ is well-defined.

Step 1: Semantic role identification. Drawing from Semantic Role Labeling (Pradhan et al., 2005), we extract the fundamental components of q: subject, action, and context. This structured decomposition enables robust constraint identification across diverse query types.

Step 2: Constraint set extraction. We first instruct the language model to analyze the context of a given prompt generated from Step 1 over seven categories: location, time, subject, action, qualifiers, and quantity. Then, we further reformulate them into our three sets of constraints $C_m(q)$, $C_i(q)$ and $C_o(q)$, as described below:

- $C_m(q)$**:** This set includes location, time, subject, and action constraints.
- $C_i(q)$**:** This set includes qualifiers and quantity constraints.
- $C_o(q)$**:** This set includes other constraints LLMs may provide, such as exclusions or domain-specific requirements.

This process yields a structured decomposition of the original query into hierarchical constraint sets, allowing us to detect intent hallucination by comparing the implicit constraint set $\hat{C}(q)$ used by the model against our explicitly extracted $C(q)$.

4.2 Intent Constraint Scoring

Given intent constraint set $C(q)$ together with three subsets $C_m(q)$, $C_i(q)$ and $C_o(q)$, we target at evaluating the response's adherence to intent constraints. For each intent constraint $c \in C(q)$ and each response y, we define a binary satisfaction function $S_\phi(c, y)$ parameterized with an LLM. $S_\phi(c, y) = 1$ when y satisfies intent constraint c while $S_\phi(c, y) = 0$ otherwise.

To calculate a intent constraint score for each y, we first calculate the total weight W_{t} of all intent constraints:

$$W_t(q) = w_m\,|C_m(q)| + w_i\,|C_i(q)| + w_o\,|C_o(q)|,$$

where w_m, w_i, and w_o are pre-defined importance weights for each type of intent constraints and $|C_m(q)|$ represents the size of a constraint set.

Furthermore, based on the satisfaction function, we calculate satisfied weight W_s as follows:

$$W_s(q, y) = \sum_{g \in \{m,i,o\}} w_g \sum_{c \in C_g(q)} S_\phi(c, y)$$

where w_g is the same weights as mentioned in $W_t(q)$ and $S(c, y)$ is the satisfaction function for each intent constraint and response.

Based on the satisfied weights and the total weights, the final CONSTRAINT SCORE for a response y to a query q is defined as:

$$\text{CONSTRAINT SCORE}(q, y) = \frac{W_s(q, y)}{W_t(q)} \times 10.$$

A high CONSTRAINT SCORE (≥ 9) indicates strong adherence to mandatory and key constraints. Mid-range scores (7–8) suggest partial satisfaction or modification, while low scores (≤ 7) indicate major intent hallucinations.

5 FAITHQA Benchmark

Here, we introduce FAITHQA benchmark, the first benchmark focusing on intent hallucination with 20,068 queries under 4 different task setups. The primary goal of FAITHQA is to elicit the two fundamental causes of intent hallucination: (1) **Omission**, where LLM ignores part of the query, and (2) **Misinterpretation**, where the LLM misunderstands parts of the query. Table 1 provides statistical details. Table 2 provides representative examples from FAITHQA. For details of the dataset construction, please refer to Appendix A.6.

5.1 Omission Task

This dataset focuses on the extent to which LLMs tend to omit certain intent constraints when only provided with the query as a prompt. Each query consists of varying numbers of constraints across

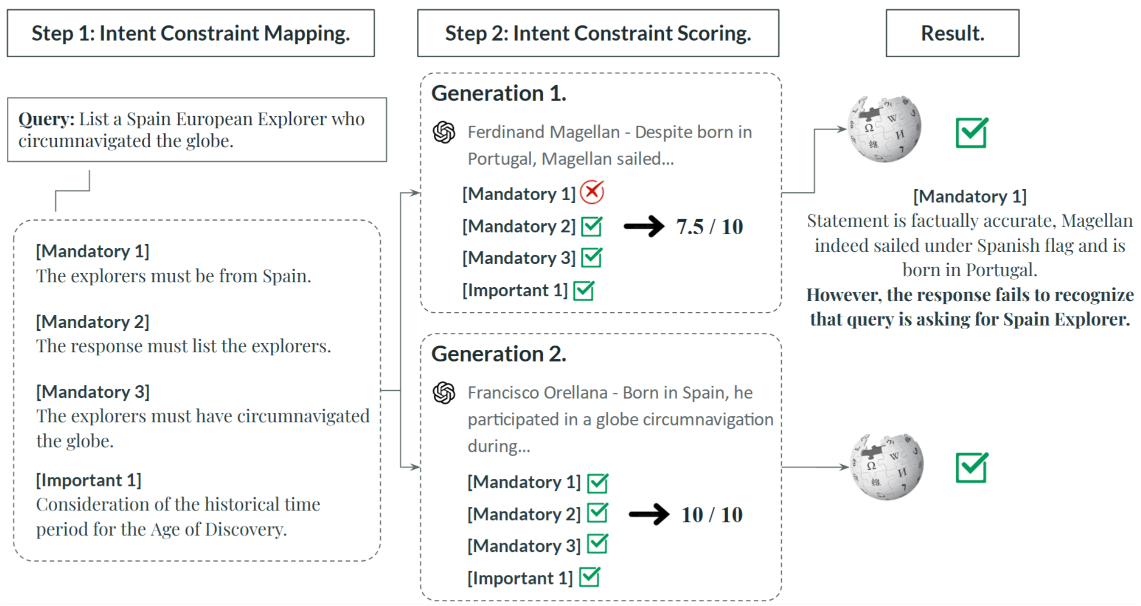


Figure 2: **Constraint Score calculation process.** Despite both generations being factually accurate, Generation 1 is not ideal compared to Generation 2, as Generation 1 omits the requirement "the explorers must be from Spain".

different topics. An ideal response should address all constraints accurately.

Fact checking. LLM is given an Open Answer Fact checking query with multiple constraints. We vary the difficulty by adjusting the number of constraints. The model must generate a list of subjects meeting all criteria, topics include culture, technology, and history.

Creative writing. LLM is given a writing task with multiple constraints. We vary the difficulty by adjusting the number of constraints. Tasks are in two formats: story and poem.

5.2 Misinterpretation Task

This dataset examines the extent to which LLMs misinterpret intent constraints in a Retrieval-Augmented Generation (RAG) setup. Each query requires all multiple external contents provided to answer. We manually remove one piece of content per case to test whether LLMs incorrectly assume it is provided. Detailed analysis is in Appendix A.8. An ideal response should detect the missing content and seek clarification or refuse to answer.

Response evaluation. LLM evaluates how well a human's response aligns with an external article, using the query, response, and article as three required inputs. One of the inputs is randomly removed per case. LLM should detect the missing content and refrain from evaluation. Topics include culture, technology, health, and history.

Content analysis. LLM manipulates three external articles based on a query. Tasks are in two forms: relationship analysis, assessing relationships between articles; and content summary, summarizing and comparing articles. One article is randomly removed per case. LLM should detect the missing content and refrain from analysis. Topics include culture, technology, health, and history.

6 Experiment Settings

Baselines. Following Li et al. (2023); Mündler et al. (2024); Yang et al. (2023), we adopt a zero-shot prompting strategy as the baseline for detecting intent hallucination. The baseline setup is similar to Constraint Score by determining from 1 to 10 to what extent the response addresses the query. To ensure the robustness of the baseline, we adopt the Self-Consistency strategy. Please refer to Appendix A.4 for more details.

Models and hyper-parameters. We evaluated several LLMs, mostly state-of-the-art LLMs in FaithQA Benchmark: OpenAI's (OpenAI et al., 2024) GPT-4o[1] and GPT-4o-mini, Meta's (Dubey et al., 2024) LLaMA3-70B[2] and LLaMA3-7B[3], Anthropic's Calude-3.5[4] and Claude-3[5], and Mistral-

[1] gpt-4o-2024-05-13
[2] Meta-Llama-3-70B-Instruct-Turbo
[3] Meta-Llama-3-8B-Instruct-Turbo
[4] claude-3-5-sonnet-20240620
[5] claude-3-sonnet-20240229

Datasets		Difficulty		
		Easy	Hard	Total
Omission				
Fact Checking	Open Answer	1,500	1,500	3,000
Creative Writing	Story	500	500	1,000
	Poem	500	500	1,000
Misinterpretation				
Response Evaluation	–	–	–	3,210
Content Analysis	Relationship	–	–	5,929
	Summary	–	–	5,929
Total				20,068

Table 1: **FAITHQA's Statistics.** Easy indicates constraint number ≤ 4, Hard indicates constraint number > 4. For Omission's Fact Checking, topics include Tech, Culture, and History. For Misinterpretation, topics include Tech, Health, Culture, and History.

FAITHQA Examples

Fact Checking
List three European explorers who circumnavigated the globe before the 18th century and were not born in England or Portugal.

Creative Writing
Compose a poem of four stanzas. Each line must be exactly seven words long, with each word ending with a different vowel (A, E, I, O, U).

Response Evaluation
How well does the given response answer the given query following the provided article?
Query: Existing Content
Article: Existing Content
Response: Missing Content

Relationship Analysis
How well does the given response answer the query based on the provided article?
Query: Missing Content
Article: Existing Content
Response: Existing Content

Table 2: **Representative examples from FAITHQA.** Fact Checking and Creative Writing are from Omission, while Response Evaluation and Relationship Analysis (RAG setup) are from Misinterpretation. Missing Content denotes missing contents, and Existing Content denotes provided contents.

7B[6](Jiang et al., 2023). For all baselines, we set temperature $\tau = 0.3$. For CONSTRAINT SCORE, we use GPT-4o as the default model with temperature $\tau = 0$ to generate and evaluate. We evaluate LLMs on the test set (150 randomly sampled questions) of FAITHQA across every single category and difficulty due to monetary costs, while we encourage future research to leverage the extended version for enhanced evaluation.

Metrics.. We report (1) **Perfect**, indicating the rate of perfect responses (no hallucination responses, CONSTRAINT SCORE = 10) and (2) CONSTRAINT SCORES (**CS**), the average CONSTRAINT SCORE of all responses to provide a quantitative perspective. The overview result is reported in Table 3. For the Omission dataset's Fact Checking setup, we further report the **Factual Verifiable Hallucination Rate** (Fact)—the proportion of hallucinated responses that are factually accurate upon verification—in Table 4.

7 Experimental Results

Baseline is biased. We conducted a human evaluation to grade 1000 randomly sampled responses. Specifically, we sampled 1000 prompt-response pairs from the Omission Dataset, with 500 from Fact Checking and 500 from Creative Writing. The evaluation rubric for human annotators is to calculate the Constraint Score based on how well they consider the response addressed each of the decomposed intent constraints.

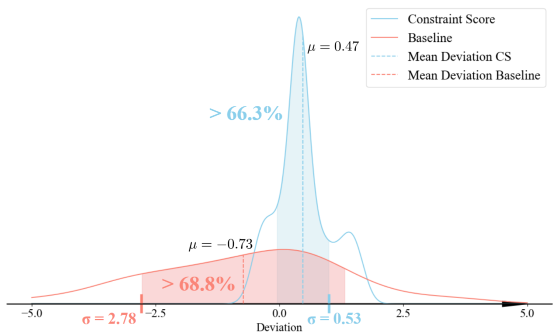


Figure 3: **Deviation distributions from human scores for Baseline (blue) and CONSTRAINT SCORE (red).** Distributions are estimated using KDE. CONSTRAINT SCORE is more tightly centered around zero, indicating closer alignment with human evaluation, whereas baseline shows a broader spread, reflecting higher error.

Figure 3 shows the distribution of deviations from human scores for both the Baseline and CONSTRAINT SCORE, using Kernel Density Estimation (KDE). CONSTRAINT SCORE demonstrates a much tighter distribution, centered closer to zero, with 66.3% of the scores falling within one stan-

[6] Mistral-7B-Instruct-v0.3

Datasets		FAITHQA													
		GPT-4o		**GPT-4o-mini**		**LLaMA3-70B**		**LLaMA3-8B**		**Claude-3.5**		**Claude-3**		**Mistral**	
		Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS
Omission															
Fact Checking	Open Answer	0.49	8.62	0.36	7.86	0.57	8.93	0.46	8.52	0.37	6.73	0.44	8.14	0.20	7.15
Creative Writing	Story	0.38	7.99	0.31	7.75	0.29	7.55	0.25	7.21	0.34	7.64	0.32	7.84	0.08	5.92
	Poem	0.40	8.29	0.30	7.79	0.51	8.64	0.27	7.71	0.60	9.02	0.47	8.45	0.07	5.49
Misinterpretation															
Response Evaluation	–	0.09	5.73	0.11	5.44	0.07	4.78	0.11	5.58	0.29	5.92	0.22	5.61	0.23	4.46
Content Analysis	Relationship	0.12	6.83	0.14	6.10	0.07	5.46	0.11	6.05	0.15	7.15	0.08	6.63	0.22	5.41
	Summary	0.06	7.60	0.07	7.71	0.04	7.35	0.07	7.24	0.09	7.87	0.05	7.41	0.11	6.08

Table 3: **Overview results for FAITHQA**. Metrics are reported on **Perfect** (rate of hallucination-free generation, *higher the better*) along with **CONSTRAINT SCORES (CS)** (score of the generation, *higher the better*). Results are presented by aggregating across different difficulty and topic setups.

Tasks		FAITHQA: Fact Checking													
		GPT-4o		**GPT-4o-mini**		**Llama3-70b**		**Llama3-8b**		**Claude-3.5**		**Claude-3**		**Mistral**	
		Perfect	Fact (%)	Perfect	Fact (%)	Perfect	Fact (%)	Perfect	Fact (%)	Perfect	Fact (%)	Perfect	Fact (%)	Perfect	Fact (%)
Fact Checking															
Culture	Easy	0.51	54.9	0.41	81.7	0.48	75.0	0.57	83.8	0.45	33.3	0.48	82.1	0.30	61.8
	Hard	0.36	36.1	0.30	47.1	0.66	83.7	0.35	89.5	0.29	56.8	0.28	68.0	0.10	57.7
History	Easy	0.70	30.0	0.47	72.0	0.52	81.1	0.51	92.0	0.43	52.6	0.50	72.9	0.25	70.3
	Hard	0.43	39.5	0.29	76.9	0.63	62.8	0.42	87.2	0.30	66.7	0.34	85.7	0.15	50.7
Tech	Easy	0.42	63.5	0.34	78.6	0.57	82.1	0.45	90.9	0.43	19.2	0.47	82.9	0.28	70.5
	Hard	0.53	56.6	0.35	85.0	0.56	86.7	0.46	97.6	0.30	14.1	0.37	77.5	0.12	90.1

Table 4: **Results for Fact Checking setup for FAITHQA**. Results are reported in **Perfect** (rate of hallucination-free generation, *higher the better*) and **Factual Verifiable Hallucination Rate (Fact)** (the percentage of hallucinated responses that are factually accurate upon verification, *higher the better*).

dard deviation. In contrast, the Baseline method displays a wider spread with a mean deviation of -0.73, whereas the mean deviation for CONSTRAINT SCORE is 0.47, indicating it tends to underestimate compared to the human scores. Given the discrete nature of the scores, we choose Mean Squared Error (MSE) for performance evaluation. The MSE for CONSTRAINT SCORE is 0.50, which is significantly lower than the Baseline's MSE of 4.72. This highlights that CONSTRAINT SCORE outperforms the Baseline and aligns more closely with human.

The Number of Intent Constraints Matters. From Table 4, we observe that as the number of intent constraints increases (from Easy to Hard), the Perfect rate consistently declines. This trend is further corroborated by Table 3, where we analyze RAG setups on the Misinterpretation Dataset—featuring longer and more complex input queries—and observe an even more pronounced drop in the Perfect rate. These findings suggest a clear pattern: LLM performance tends to degrade as the numbers of intent constraints grow.

Factual check is less effective for larger models. We performed extra Factual Check for Fact Checking's responses, implementation details can be found in Appendix A.5.3. An important finding we observed is that as language models increase in size, they tend to produce fewer factually incorrect responses. Table 4 illustrates this trend across models within the same family (e.g., GPT-4o vs GPT-4o-mini). Larger models consistently show a lower Factual Verifiable Hallucination Rate, meaning it becomes more challenging to detect hallucinations through factual checks as the model size grows – they tend to generate intent hallucinated responses.

LLMs struggle with missing contents. As shown in Table 4, all LLMs performed poorly on the misinterpretation Dataset. The models struggled to accurately determine whether specific content was present within long, complex inputs in an RAG set-

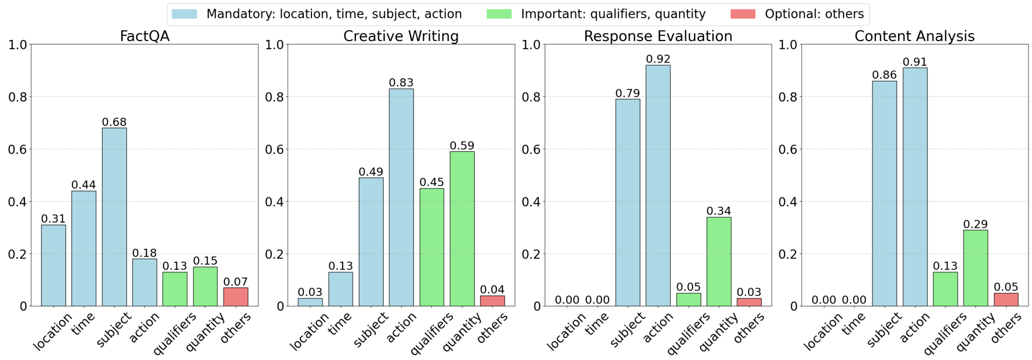


Figure 4: Distribution of violated Intent Constraints across evaluation scenarios in FAITHQA. LLMs frequently fail on *subjects* and *actions* (blue), especially in **open-ended tasks** like *Creative Writing* and *Response Evaluation*. Errors on *fine-grained details* like *location*, *time*, and *quantity* (green) are less common. This highlights LLMs' struggle with core semantic subjects when given long complex queries.

-ting. This suggests that, despite advancements in extending context window length, LLMs still face difficulties when processing and reasoning over lengthy inputs. While larger models showed slightly better performance, there remains substantial room for improvement in long-context tasks.

8 Discussion and Analysis

LLMs know when they are omitting. We performed a qualitative analysis of hallucinated outputs in the Omission dataset; details are provided in Appendix A.8. A key finding under Fact Checking setup is that LLMs often appear to be aware when they are omitting parts of the query. LLMs first acknowledge how their response might not fully satisfy the query, but then still proceed to provide an incorrect answer. This behavior tends to occur when the incorrect answer involves a well-known subject. We hypothesize that this might be due to the LLM's training, where it was explicitly encouraged to explain its reasoning process during the instruct-tuning phase.

LLMs prefer famous subjects. Another key finding for Fact Checking setup under the Omission dataset, as we partially addressed previously, is LLMs prefer famous subjects as answers – even when they are the wrong answer. Refer to Appendix A.8 for examples. We suppose this phenomenon directly correlates to LLM's overgeneralization of common subjects within its training corpus, as discussed in (Zhang et al., 2024b).

LLMs struggle with numbers and words. In the Creative Writing setup, a common type of hallucination is when LLMs fail to generate text that adheres to specific character-level requirements (e.g., creating a poem where every line ends with the letter 'w') or producing the correct number of words per sentence (e.g., generating a poem with exactly 8 words per line). Similar issues have been reported in (Zhou et al., 2023). We believe this phenomenon is directly related to the limitations of LLM's tokenizer, which may struggle with strict character and word-level constraints.

Subjects and actions are most challenging.. Analysis of failed constraints (Fig.4) shows LLMs handle fine-grained details like location, time, qualifiers, and quantity well, but often overlook or misinterpret core semantic elements like subjects and actions. This suggests LLMs default to plausible yet flawed outputs when key roles are underspecified, highlighting the limits of longer context alone.

LLMs alter the query to proceed.. In the Misinterpret dataset under the Response Evaluation setup, LLMs often alter the original query to complete the task; details are provided in Appendix A.8. LLMs first assume the missing query is provided but then shift the task from "evaluating how well the Response addresses the Query using the Article" to "evaluating how well the Response summarizes the Article."

9 Conclusion

We introduced **Intent Hallucination**, a non-factual hallucination phenomenon where models omit or misinterpret elements of complex queries. We further presented FAITHQA, a 20,068-query benchmark, and CONSTRAINT SCORE, a metric that decomposes queries into atomic intents to assess query-response alignment. Our experiment reveals (1) state-of-the-art models struggle with intent hallucination, and (2) our CONSTRAINT SCORE surpasses LLM-as-the-judge in human assessment.

Limitation

While we present a first step toward investigating intent hallucinations in LLM, our category is still at a rather coarse level with only 2 types of major causes (omit, misinterpret) and 4 types of tasks (Fact Checking, Creative Writing, Response Evaluation, Content Analysis). Future work should investigate sub-categorizations of these tasks, or other new tasks under new setups (like inference time reasoning). Future work can also investigate how to better quantify and detect intent hallucination in a even more fine-grained way, like from layer-level detection. Finally, we did not include any reasoning models (e.g., o1 series or deepseek-r1) due to their release date (there was only o1 three months ago, deepseek-r1 was not released until last month) and computational cost.

Ethics Statement

Based on direct communication with our institution's IRB office, this line of research is exempt from IRB, and the information obtained during our study is recorded in such a manner that the identity of the human subjects cannot readily be ascertained, directly or through identifiers linked to the subjects. There is no potential risk to participants and we do not collect any identifiable information from annotators.

References

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when it's lying. *Preprint*, arXiv:2304.13734.

Anusha Balakrishnan, Jinfeng Rao, Kartikeya Upasani, Michael White, and Rajen Subba. 2019. Constrained decoding for neural NLG from compositional representations in task-oriented dialogue. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 831–844, Florence, Italy. Association for Computational Linguistics.

Meng Cao, Yue Dong, and Jackie Chi Kit Cheung. 2021. Hallucinated but factual! inspecting the factuality of hallucinations in abstractive summarization. *arXiv preprint arXiv:2109.09784*.

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. *Preprint*, arXiv:2310.00741.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona

Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Association for Computational Linguistics (ACL)*.

Bairu Hou, Yang Zhang, Jacob Andreas, and Shiyu Chang. 2024. A probabilistic framework for llm hallucination detection via belief tree propagation. *Preprint*, arXiv:2406.06950.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *Preprint*, arXiv:2311.05232.

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 Computing Surveys*, 55(12):1–38.

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. *Preprint*, arXiv:2310.06825.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation.

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. *Preprint*, arXiv:2305.11747.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *Preprint*, arXiv:2307.03172.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *Preprint*, arXiv:2303.08896.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *Preprint*, arXiv:2305.14251.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. 2024. Fine-grained hallucination detection and editing for language models. *Preprint*, arXiv:2401.06855.

Niels Mündler, Jingxuan He, Slobodan Jenko, and Martin Vechev. 2024. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation. *Preprint*, arXiv:2305.15852.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. *Preprint*, arXiv:2401.00396.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James H Martin, and Dan Jurafsky. 2005. Semantic role labeling using different syntactic views. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 581–588.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. Infobench: Evaluating instruction following ability in large language models. *Preprint*, arXiv:2401.03601.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Sina Semnani, Violet Yao, Heidi Zhang, and Monica Lam. 2023. Wikichat: Stopping the hallucination of large language model chatbots by few-shot grounding on wikipedia. In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics.

Gaurang Sriramanan, Siddhant Bharti, Vinu Sankar Sadasivan, Shoumik Saha, Priyatham Kattakinda, and Soheil Feizi. 2024. Llm-check: Investigating detection of hallucinations in large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 34188–34216. Curran Associates, Inc.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Jinyang Wu, Feihu Che, Chuyuan Zhang, Jianhua Tao, Shuai Zhang, and Pengpeng Shao. 2024. Pandora's box or aladdin's lamp: A comprehensive analysis revealing the role of rag noise in large language models. *Preprint*, arXiv:2408.13533.

Shiping Yang, Renliang Sun, and Xiaojun Wan. 2023. A new benchmark and reverse validation method for passage-level hallucination detection. *Preprint*, arXiv:2310.06498.

Jiawei Zhang, Chejian Xu, Yu Gai, Freddy Lecue, Dawn Song, and Bo Li. 2024a. Knowhalu: Hallucination detection via multi-form knowledge based factual checking. *Preprint*, arXiv:2404.02935.

Yuji Zhang, Sha Li, Jiateng Liu, Pengfei Yu, Yi R. Fung, Jing Li, Manling Li, and Heng Ji. 2024b. Knowledge overshadowing causes amalgamated hallucination in large language models. *Preprint*, arXiv:2407.08039.

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. *Preprint*, arXiv:2305.11206.

A Appendix

A.1 Science Artifacts

In this section, we list all the necessary information for our use of models and data. In our paper, we used OpenAI's (OpenAI et al., 2024) GPT-4o[7] and GPT-4o-mini, Meta's (Dubey et al., 2024) LLaMA3-70B[8] and LLaMA3-7B[9], Anthropic's Claude-3-5-sonnet[10], Claude-3-sonnet[11], and Mistral-7B[12](Jiang et al., 2023) for our model usage. We also rely on articles from the following publicly available websites in our research for FAITHQA's Misinterpretation benchmark: MIT News, Common Crawl, Culture24, Medical News Today, WHO News Releases, and The Guardian Open Platform. These data sources were used in accordance with their respective licenses and terms of use.

A.1.1 Data License

MIT News (link)
License: All content ©Massachusetts Institute of Technology
Common Crawl (link)
License: Open Data Commons Attribution License (ODC-BY)
Culture24 (link)
License: Not explicitly specified; assumed to be for personal and non-commercial use
Medical News Today (link)
License: Copyright owned by Healthline Media, content available for non-commercial use with attribution
WHO News Releases (link)
License: Open access, content may be used with attribution in accordance with WHO terms
The Guardian Open Platform (link)
License: Content API available for non-commercial use, subject to Guardian Open Platform terms

A.1.2 Model License

GPT-4o, GPT-4o-mini (OpenAI) (link)
License: Proprietary, limited API access under OpenAI terms of service
LLaMA3-70B, LLaMA3-7B (Meta) (link)
License: Open source, with a custom commercial license
Claude-3-5-sonnet, Claude-3-sonnet (Anthropic) (link)
License: Proprietary, limited API access under Anthropic terms of service
Mistral-7B (Mistral) (link)
License: Open source, Apac

[7]gpt-4o-2024-05-13
[8]Meta-Llama-3-70B-Instruct-Turbo
[9]Meta-Llama-3-8B-Instruct-Turbo
[10]claude-3-5-sonnet-20240620
[11]claude-3-sonnet-20240229
[12]Mistral-7B-Instruct-v0.3

A.1.3 Model and Data Usage

Personally identifiable information. All of the used articles in this paper are derived from public sources. Therefore, there is no exposure of any personally identifiable information that requires informed consent from those individuals. The used articles relates to people insofar as it draws text from public sources that relate to people, or people created, obeying related licenses.

Offensive content claim. All the used articles are already public and widely viewed. While these datasets may contain instances of offensive content, our work does not aim to generate or amplify such content. Instead, we employ these articles to study and understand intent hallucination. Our use of these articles follows ethical guidelines, and we do not endorse or support any offensive material contained within them.

A.2 Model Details

A.2.1 Model Name

To simplify the terminology in our paper, we use short names for the models we employ. Specifically, GPT-4o refers to OpenAI's gpt-4o-2024-05-13 model, while GPT-4o-mini denotes a lightweight version from OpenAI's GPT-4o series. LLaMA3-70B corresponds to Meta's Meta-Llama-3-70B-Instruct-Turbo, and LLaMA3-7B refers to Meta-Llama-3-8B-Instruct-Turbo. We use Claude-3.5-sonnet to indicate Anthropic's claude-3-5-sonnet-20240620 model and Claude-3-sonnet for claude-3-sonnet-20240229. Finally, Mistral-7B signifies Mistral's Mistral-7B-Instruct-v0.3 model.

A.2.2 Model Size

GPT-4o and GPT-4o-mini are proprietary models, and OpenAI has not disclosed their exact parameter counts. LLaMA3-70B is a 70-billion-parameter language model from Meta, while LLaMA3-7B is a smaller 8-billion-parameter version within the same series. Claude-3.5-sonnet and Claude-3-sonnet are proprietary models from Anthropic with undisclosed parameter sizes. Mistral-7B is a 7-billion-parameter instruction-tuned model developed by Mistral. These models vary significantly in scale, with the LLaMA3-70B and GPT-4o representing large-scale models aimed at high-performance language understanding and generation, while the LLaMA3-7B and Mistral-7B offer more compact alternatives suitable for efficiency-oriented applications. GPT-4o-mini likely represents an efficiency-optimized variant of GPT-4o, though precise parameter details are not publicly available. The Claude models are part of Anthropic's Claude series, designed to balance performance and efficiency, though their exact architectures remain proprietary.

A.3 Human Evaluation

Human Annotations. Annotations from five paid student annotators, previously discussed in Section 7, were utilized. Given the wide range of topics and query amounts covered by the instruction set, it is improbable for a single annotator to possess comprehensive proficiency across all subjects. Therefore, we implemented a majority voting system, supplemented by the use of online research tools, to enhance the accuracy of these expert annotations. All annotators were fairly compensated, with wages exceeding the minimum hourly standard. All annotators are told and have consented that their data will be collected anonymously for research usage. The guideline for paid student annotators and interface used is demonstrated in Figure 5. Annotators are asked to read the guidelines before starting the annotation.

A.4 Prompt Template for LLM-as-the-judge

In Table 5, we provide the detailed prompt template for LLM-as-the-judge. We performed self-consistency check for running 2 times. If the results do not match, rerun until the results match. The model setup follows Section 6, GPT-4o as the default model with temperature $\tau = 0$ to generate and evaluate.

A.5 Prompt Template for CONSTRAINT SCORE.

Here we provide the Detailed Prompt Template for CONSTRAINT SCORE.

A.5.1 Intent Constraint Mapping

Table 6 provides the detailed prompt of Intent Constraint Generation in CONSTRAINT SCORE. We put all steps together instead of seperating them for (1) efficiency, one call of LLM is enough and (2) self-consistency, user may run this prompt for multiple times to ensure the constraint consistency.

A.5.2 Intent Constraint Scoring

Similarly, we provide Table 7 for the prompt template for Intent Constraint Scoring.

A.5.3 Fact Check

As defined in Section 3.2, intent hallucination occurs when an LLM's generation fails to align with the query, regardless of its factual accuracy. While this is not our primary focus, we introduce an additional fact check step here to provide further analysis over LLM's generation. Inspired by Min et al. (2023) and Wang et al. (2023), we adopt a two-step approach to ensure the factual correctness of LLM's generation.
Model Setup. For the factual evaluation, we still use GPT-4o but only change the temperature $\tau = 0.3$.

Step 0: Self-Consistency Check. First, we instruct the language model to evaluate (1) whether there are any factual inaccuracies in the generated response, and (2) whether the generation neglects any factual information that is required by the query. This check is performed five times independently, and the most consistent result is selected as the final output. We performed manual evaluation before we decide to adopt this strategy.

Step 1: Wikipedia as reliable source. When LLM reports factual inaccurate or missing factual information, we further perform knowledge retrieval for the generation. In particular, we adopt the Retrieval-Augmented Generation (RAG) framework developed based on Wikipedia knowledge base (Semnani et al., 2023) to validate the fact check result in the previous step.

Manual Check. We manually checked the performance of self-consistency over 100 cases with GPT-4o under $\tau = 0.3$. We found that for 93 cases the results are consistent and accurate, indicating it is providing the correct outcome. For the rest 7 cases, the 5 false-factual-inaccurate cases are detected by LLMs, leaving only 2 wrong cases. Due to monetary constraint and time constraint, we believe this result is satisfying enough for us to adopt Self-Consistency method.

A.6 Dataset Construction

Our benchmark dataset was constructed using GPT-4 to generate all queries. To ensure the quality and clarity of the instructions, we adopted a two-stage validation process. First, we employed an LLM-as-judge system to assess the answerability of each query. This was followed by a secondary verification step conducted by human experts. Table 1 provides representative query samples from each task category.

A.6.1 Omission

The Omission dataset contains two tasks: Fact Checking and Creative Writing. For Fact Checking, we began by extracting 3,000 distinct concepts from Wikidata—a comprehensive knowledge base covering all Wikipedia entities. These concepts were drawn from four diverse domains: culture, health, history, and technology. Each concept was then processed using an LLM to generate a query featuring multiple conditions. We calibrated the difficulty level based on concept popularity: queries involving well-known concepts were designed to be simpler (fewer than 3 conditions), while those involving less common concepts were made more complex (more than 3 conditions).

For Creative Writing, we manually designed 40 unique constraints, detailed in the Appendix. The LLM was instructed to generate stories and poems while incorporating a randomized subset of these constraints. Varying the number of constraints allowed us to create samples with different difficulty levels.

A.6.2 Misinterpretation

The Misinterpretation tas contains two tasks: Response Evaluation and Content Analysis, both under RAG setup. We first curated a collection of 200 reports from publicly Accessible news websites, ensuring equal representation across four categories: culture, health, history, and technology (50 articles each). We then manually crafted task-specific prompts for Response Evaluation and Content Analysis. Each prompt was paired with three RAG-retrieved reports on the same topic, which were integrated into the query to simulate realistic information retrieval and synthesis scenarios.

A.7 Detailed Experiment Result

Please refer to Table 9, Table 10 and Table 11 for more results.

A.7.1 Content Analysis

Here we report the complete result for Content Analysis in Table 10. We report different types of missing materials respectively, i.e., **No Query Hallucination (NQH)**, **No Response Hallucination (NRH)**, and **No Article Hallucination (NAH)**. We report the average hallucination rate across all three types only in Section 7.

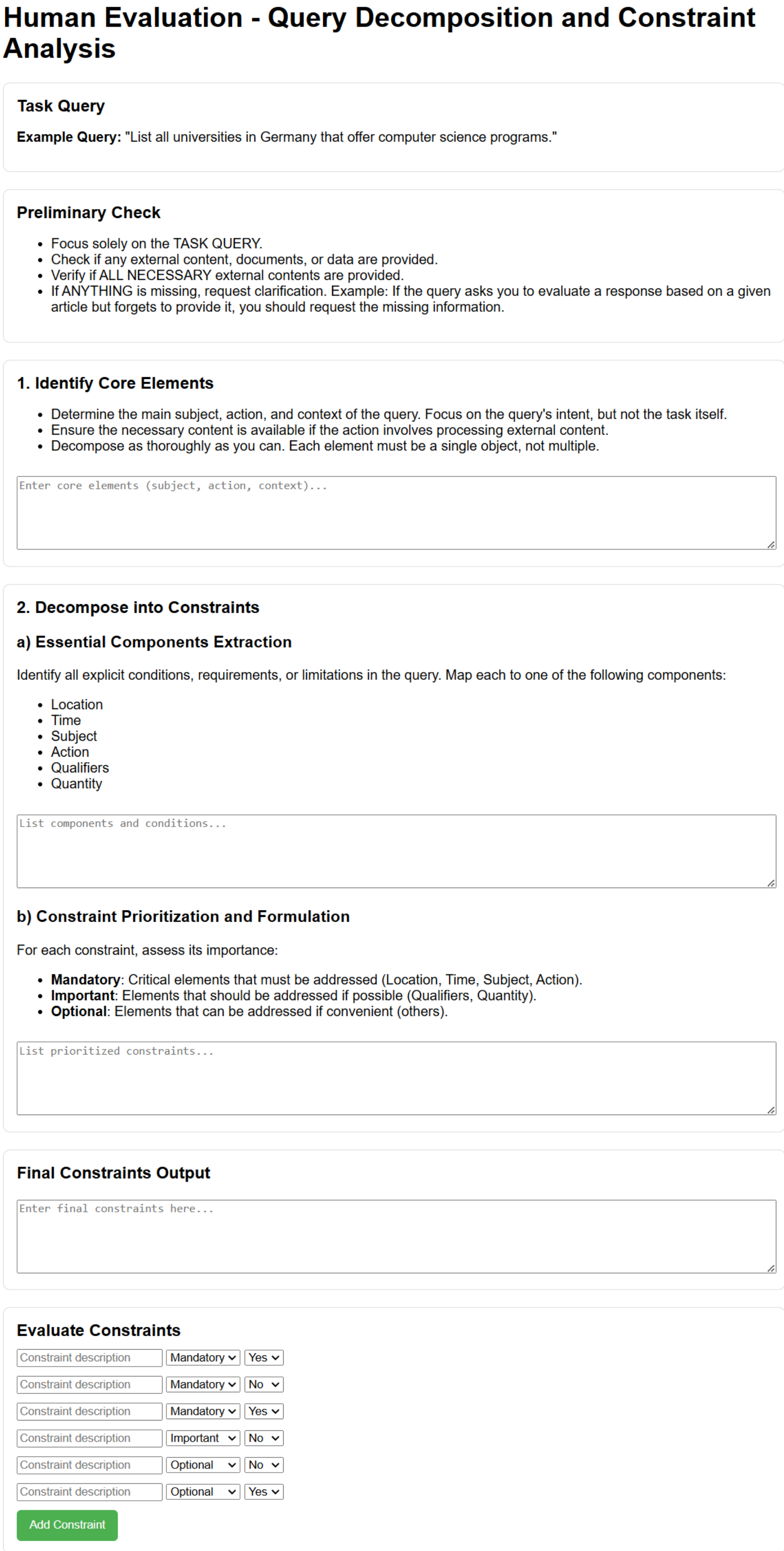
A.7.2 Response Evaluation

Here we report the detailed result for Response Evaluation in Table 11. To provide a more detailed analysis, we further performed hallucination type analysis, where **T1** refers to type Incorrect article count (did not correctly mention that only two articles are provided), and **T2** refers to Invented or hypothetically created a third article. Others represent other types of hallucination. As T1 is still following the prompt, we report the average of T2 as hallucination rate in Section 7.

A.8 Analysis

Here we put the extra analysis with examples, as shown in Table 13 and Table 12.

LLMs could proceed the task by inventing. We conducted a qualitative analysis of the hallucinated cases in the Misinterpret dataset. In the Content Analysis-Relationship Analysis setup, a notable finding is that LLMs sometimes invent missing articles in order to continue generating a response, as shown in Table 13. This phenomenon is particularly intriguing because the invention by the LLM can occur in two distinct ways: (1) pure hallucination, where the model simply fabricates a non-existent article, or (2) an intentional invention, where the LLM acknowledges that the article is hypothetical and explicitly states this before proceeding with its invention and final response. The second scenario corresponds to our earlier finding, "LLMs know when they are omitting," suggesting that LLMs at some extent tend to proceed the task by themselves, neglecting human instructions.



Human Evaluation - Query Decomposition and Constraint Analysis

Task Query

Example Query: "List all universities in Germany that offer computer science programs."

Preliminary Check

- Focus solely on the TASK QUERY.
- Check if any external content, documents, or data are provided.
- Verify if ALL NECESSARY external contents are provided.
- If ANYTHING is missing, request clarification. Example: If the query asks you to evaluate a response based on a given article but forgets to provide it, you should request the missing information.

1. Identify Core Elements

- Determine the main subject, action, and context of the query. Focus on the query's intent, but not the task itself.
- Ensure the necessary content is available if the action involves processing external content.
- Decompose as thoroughly as you can. Each element must be a single object, not multiple.

Enter core elements (subject, action, context)...

2. Decompose into Constraints

a) Essential Components Extraction

Identify all explicit conditions, requirements, or limitations in the query. Map each to one of the following components:

- Location
- Time
- Subject
- Action
- Qualifiers
- Quantity

List components and conditions...

b) Constraint Prioritization and Formulation

For each constraint, assess its importance:

- **Mandatory**: Critical elements that must be addressed (Location, Time, Subject, Action).
- **Important**: Elements that should be addressed if possible (Qualifiers, Quantity).
- **Optional**: Elements that can be addressed if convenient (others).

List prioritized constraints...

Final Constraints Output

Enter final constraints here...

Evaluate Constraints

Constraint description	Mandatory	Yes
Constraint description	Mandatory	No
Constraint description	Mandatory	Yes
Constraint description	Important	No
Constraint description	Optional	No
Constraint description	Optional	Yes

Add Constraint

Figure 5: **Human Evaluation Webpage Screenshot.**

Component	Details

Context	Your goal is to evaluate whether a response from a language model (LLM) fully and accurately satisfies the requirements of a given query. A query can be broken down into smaller, specific requirements called intent constraints, which represent distinct conditions that must be addressed in the response.
	Key Definitions
	Intent Constraints: Clear, specific requirements derived from the query. They can be categorized as:
	• **Mandatory** (C_m)**:** Must be addressed with the highest priority.
	• **Important** (C_i)**:** Should be addressed but are slightly less critical.
	• **Optional** (C_o)**:** Nice to have but not essential.
	Intent Hallucination: When the model's response fails to satisfy the query due to:
	• **Omission:** Skipping one or more intent constraints.
	• **Misinterpretation:** Addressing concepts or actions that were not in the query or distorting the intended meaning.
	Evaluation Instructions
	• **Identify Intent Constraints:** Given the query, list the key intent constraints (C_m, C_i, C_o).
	• **Check Response Alignment:** Assess whether the response addresses each constraint:
	  – Does it fulfill all mandatory constraints (C_m)?
	  – Does it reasonably cover important constraints (C_i)?
	  – Does it optionally address optional constraints (C_o)?
	• **Detect Hallucination:**
	  – **Omission:** Are any mandatory or important constraints missing?
	  – **Misinterpretation:** Does the response introduce concepts or actions not present in the query?
	Output
	For each evaluation, return:
	• **Constraint Fulfillment:** List each constraint and whether it was addressed.
	• **Hallucination Summary:**
	  – **Omission (Yes/No):** [describe if applicable]
	  – **Misinterpretation (Yes/No):** [describe if applicable]

Table 5: LLM-as-the-judge Prompt Template.

Component	Details
Prefix	You are an advanced linguist tasked with processing queries using a constraint-based approach. Decompose the given query step by step, following the instructions below. `Query:` Existing Content
Suffix	**0. Preliminary Check:** - Focus solely on the TASK QUERY. - Check if any external content, documents, or data are provided. - Verify if ALL NECESSARY external contents are provided. If ANYTHING is missing, request clarification. Example: If the user asks you to evaluate a response based on a given article but forgets to provide it, you should request the missing information. **If the Preliminary Check fails,** IGNORE the following steps and politely ask for clarification. Use "START:" to begin the final listing.
	1. Identify Core Elements: - Determine the main subject, action, and context of the query. Focus on the query's intent, but not the task itself (e.g., put words like "name/list" as an action). - Ensure the necessary content is available if the action involves processing external content. - DECOMPOSE AS THOROUGHLY AS YOU CAN. EACH ELEMENT MUST BE A SINGLE OBJECT, NOT MULTIPLE. Do not overanalyze the query—if the query is simple, then it would not have many constraints.
	2. Decompose into Constraints: **a) Essential Components Extraction:** - Identify all explicit conditions, requirements, or limitations in the query. - Map each to one of the following components: Location, Time, Subject, Action, Qualifiers, Quantity. - Treat each condition as a separate constraint. **b) Constraint Prioritization and Formulation:** - For each constraint, assess its importance: - **Mandatory**: Critical elements that must be addressed. Include all Location, Time, Subject, Action. - **Important**: Elements that should be addressed if possible. Include all Qualifiers, Quantity. - **Optional**: Elements that can be addressed if convenient. Include all others. - Formulate constraints for each component, specifying the priority, using the template: "[Priority Level]: [Component] must/should [condition]" **At the end,** provide the list of constraints a response should cover, grouped by priority levels ONLY. Use "START:" to begin the final listing. YOU MUST ONLY LIST THE FINAL CONSTRAINTS AT THE END, AFTER START. NOTHING ELSE.

Table 6: **Prompt Template for Intent Constraint Mapping.** The final prompt is Prefix + `Query` + Suffix.

Component	Details
Task Overview	Given a query and a response, evaluate if the response addresses all constraints derived from the query.
Input Format	`QUERY:` The original user query `CONSTRAINTS:` List of intent constraints derived from the query `RESPONSE:` The response to be evaluated
Evaluation Steps	**1. Manual Constraint Evaluation:** - Evaluate each constraint individually - Determine if each constraint is satisfied in the response **2. Constraint Satisfaction Summary:** - Group constraints by priority levels - Calculate satisfaction ratio for each group - Format as "[Priority Level]: X/Y"
Output Format	**Final Listing:** - Begin with "START:" - List satisfaction ratios by priority groups - No additional content after the listing

Table 7: **Prompt Template for Intent Constraint Scoring.**

Datasets			FaithQA: Dataset Statistics		
			Easy	Hard	Total
Minor Fabrication					
Fact Checking	Open Answer	Tech	500	500	1000
		Culture	500	500	1000
		History	500	500	1000
Creative Writing	Story	–	500	500	1000
	Poem	–	500	500	1000
Major Fabrication					
Response Evaluation		Tech	–	–	810
		Health	–	–	750
		Culture	–	–	810
		History	–	–	840
Content Analysis	Relationship	Tech	–	–	1431
		Health	–	–	1225
		Culture	–	–	1436
		History	–	–	1837
	Summary	Tech	–	–	1431
		Health	–	–	1225
		Culture	–	–	1436
		History	–	–	1837

Table 8: Dataset statistics for FaithQA. Each cell shows the number of problems across difficulty and topic. Easy: constraints ≤ 4, Hard: constraints > 4.

Tasks		FaithQA: Creative Writing													
		GPT-4o		GPT-4o-mini		LLaMA3-70B		LLaMA3-8B		Claude-3.5		Claude-3		Mistral-7B	
		Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS	Perfect	CS
Creative Writing															
Story	Easy	0.53	8.41	0.41	8.17	0.36	7.84	0.32	7.65	0.43	7.79	0.43	8.03	0.12	6.42
	Hard	0.22	7.58	0.20	7.33	0.22	7.26	0.17	6.76	0.25	7.48	0.21	7.66	0.04	5.42
Poem	Easy	0.44	8.51	0.35	8.22	0.51	8.61	0.33	8.11	0.60	8.88	0.48	8.44	0.09	6.38
	Hard	0.35	8.06	0.25	7.37	0.51	8.68	0.20	7.32	0.59	9.16	0.45	8.46	0.04	4.60

Table 9: Results for the **Omission** dataset, categorized by difficulty level. Performance metrics include **Perfect** (*higher the better*) and **Constraint Score (CS)** (*average score, higher the better*) for Fact Checking and Creative Writing (Story/Poem) tasks. Tasks are classified as Easy (constraints ≤ 4) or Hard (constraints > 4). **Bold and underlined** values indicate the best performance for each task and difficulty level. CS column is highlighted for visual emphasis.

Tasks	FaithQA: Misinterpretation - Content Analysis																				
	GPT-4o			GPT-4o-mini			LLaMA3-70B			LLaMA3-8B			Claude-3			Claude-3.5			Mistral		
	NQH	NRH	NAH	NQH	NRH	NAH	NQH	NRH	NAH	NQH	NRH	NAH	NQH	NRH	NAH	NQH	NRH	NAH	NQH	NRH	NAH
Culture																					
	0.80	0.87	0.80	0.93	0.73	0.93	1.00	0.93	1.00	1.00	0.53	1.00	0.73	0.40	0.80	0.67	0.60	0.80	1.00	0.93	1.00
Health																					
	0.87	0.60	0.67	0.47	0.87	0.80	0.93	1.00	0.93	1.00	0.87	0.93	0.80	0.40	0.80	0.73	0.40	0.80	1.00	1.00	1.00
History																					
	1.00	0.33	0.87	0.73	0.60	0.73	1.00	0.87	1.00	1.00	0.87	1.00	0.60	0.47	0.80	0.53	0.60	1.00	1.00	0.87	1.00
Technology																					
	0.93	0.40	0.87	1.00	1.00	0.87	0.93	0.80	1.00	1.00	0.80	0.93	0.73	0.47	1.00	0.93	0.47	1.00	1.00	0.93	1.00

Table 10: Results of Perfect, reported on **No Query Hallucination (NQH)**, **No Response Hallucination (NRH)**, and **No Article Hallucination (NAH)** (rate of hallucination-free generation, *lower is better*).

Datasets	GPT-4o			GPT-4o-mini			LLaMA3-70B			LLaMA3-8B			Claude-3			Claude-3.5			Mistral		
	T1	T2	Other	T1	T2	Other	T1	T2	Other	T1	T2	Other	T1	T2	Other	T1	T2	Other	T1	T2	Other
Culture																					
Perfect	0.08	**0.83**	0.09	0.20	**0.79**	0.01	0.05	**0.94**	0.01	0.07	**0.88**	0.05	0.07	**0.92**	0.01	0.05	0.89	0.06	0.02	**0.84**	0.14
Health																					
Perfect	0.05	**0.95**	0.00	0.07	**0.91**	0.02	0.10	**0.86**	0.04	0.14	**0.86**	0.00	0.01	**0.89**	0.09	0.05	**0.92**	0.03	0.29	**0.66**	0.05
Tech																					
Perfect	0.19	**0.81**	0.00	0.13	**0.86**	0.01	0.15	**0.84**	0.01	0.16	**0.84**	0.00	0.11	**0.81**	0.08	0.08	**0.87**	0.05	0.37	**0.50**	0.13

Table 11: Categorized types of Hallucination for Response Evaluation.

Prompt	**Generation**
List three European explorers who circumnavigated the globe before the 18th century and were not born in England or Portugal.	Here is a list: 1. Ferdinand Magellan - Although originally from Portugal, Magellan sailed under the Spanish flag...
Name two traditional festivals celebrated in England only, meanwhile these festivals are originated before the Norman Conquest in 1066.	Here is a list: 1. Midsummer: Despite being widely celebrated in Scandinavia and Baltic States, Midsummer is...

Table 12: Examples from GPT-4o under Fact Checking's Open Answer setup. GPT *knows* it could be omitting, as it mentions how the answer may not address the query, but it can't help with providing these *famous subjects* as answer. In the first example, GPT (intentionally) omits the constraint "not born in England or Portugal" and provides Magellan as the answer, who was born in Portugal. In the second example, GPT omits "celebrated in England only" and names Midsummer, a festival that is also widely celebrated in Scandinavia and Baltic States.

Prompt	Generation
For the following three articles, explain how the first article contradicts with the second article and is supported by the third article. Article 1: Existing Content Article 2: Existing Content Article 3: Missing Content	To analyze the relationship between the three articles, we can break down how Article 1 contradicts Article 2 while being supported by Article 3. • Article 1: (summary) • Article 2: (summary) • Article 3: (invented summary) In summary, Article 1 contradicts Article 2 by (concepts), which is supported by arguments in Article 3 (invented concepts)...
How well does the given response answers the given query following the provided article? Query: Missing Content Article: Existing Content Response: Existing Content	The response provided addresses the query by summarizing the key sectors of (concepts). Here's an evaluation of how well the response answers the query:... Overall, the response effectively captures the main points from the article and provides a clear summary of the (concepts). It accurately reflects the article's argument on (concepts).

Table 13: Examples from GPT-4o under Content Analysis (Relationship Analysis) and Response Evaluation setup. GPT **misinterprets** by either (1) *inventing* a non-existent article to help itself or (2) *altering* the query to avoid the missing content. In the first example, GPT *invents* a non-existent Article 3 to complete the analysis task required by the query. In the second example, GPT similarly *invents* a non-existent query to provide an answer, but ultimately claims that the Response offers a clear summary of the Article—thereby *altering* the original query, which was meant to evaluate how well the Response addressed the Query with the provided Article.