Enhancing Reliability across Short and Long-Form QA via Reinforcement Learning

Anonymous ACL submission

Abstract

While reinforcement learning has unlocked unprecedented complex reasoning in large language models, it has also amplified their propensity for hallucination, creating a critical trade-off between capability and reliability. This work confronts this challenge by introducing a targeted RL framework designed to mitigate both intrinsic and extrinsic hallucinations across short and long-form question answering. We address extrinsic hallucinations (flawed internal knowledge) by creating a novel training set from open-ended conversions of TriviaQA. Concurrently, we tackle intrinsic hallucinations (unfaithfulness to context) by leveraging long-form texts from FineWeb in a fact-grounding reward scheme. To further bolster reliability, our framework explicitly rewards the model for refusing to answer unanswerable questions, thereby cultivating crucial cautiousness. Extensive experiments demonstrate that our methodology yields significant performance gains across a diverse suite of benchmarks, substantially reducing both hallucination types. Ultimately, this research contributes a practical framework for resolving the critical tension between advanced reasoning and factual trustworthiness, paving the way for more capable and reliable large language models.

1 Introduction

Recent advancements in reinforcement learning (RL) have empowered large language models (LLMs) to exhibit longer chain-of-thought (CoT) capabilities, significantly enhancing their complex reasoning abilities (Guo et al., 2025; Jaech et al., 2024). However, this progress comes at a cost, as these models show higher hallucination rates than their base counterparts (OpenAI, 2025; Yao et al., 2025). This heightened hallucination rate in RL-driven models may stem from an avalanche effect where long CoT chains lead to irreversible error accumulation. Additionally, existing research has prioritized reasoning enhancement over hallucination mitigation.

Hallucinations are typically categorized as intrinsic or extrinsic (Ji et al., 2023). Extrinsic hallucinations are often defined as errors in a model's internal knowledge and are frequently confused with "factuality" due to varying definitions across different studies (Bang et al., 2025; Yao et al., 2025). In this work, we define extrinsic hallucinations broadly to include both the generation of entirely fabricated knowledge and relational fallacies (e.g., temporal inaccuracies). In contrast, intrinsic hallucinations occur when a model fails to use knowledge explicitly provided by the user, such as not following instructions or ignoring given reference material. While some studies have used RL to address hallucinations (Yang et al., 2025b; Song et al., 2025; Ren et al., 2025), they often rely on highly restrictive methods, such as generating short-form outputs or simply having the model refuse to answer. These approaches are limited to mitigating internal knowledge or factuality issues, leaving a notable research gap in addressing extrinsic hallucinations—a growing concern in the wake of RL-driven LLM advancements.

In this work, we categorize question answering (QA) tasks into two main types: short-form QA and long-form QA. For short-QA, which includes tasks focused on factuality and unanswerable questions, we designed a novel RL reward method. Following the approach of prior work (Yang et al., 2025b; Song et al., 2025; Ren et al., 2025), our method successfully improves model performance while simultaneously enhancing reliability.

For long-form QA, we constructed a training set spanning two distinct scenarios: with and without reference content. For the scenario with reference content, we adopted the evaluation method of FACTS Grounding (Jacovi et al., 2025), retrieving 2,000 high-quality data samples from Fineweb (Penedo et al., 2024). We then used LLMs

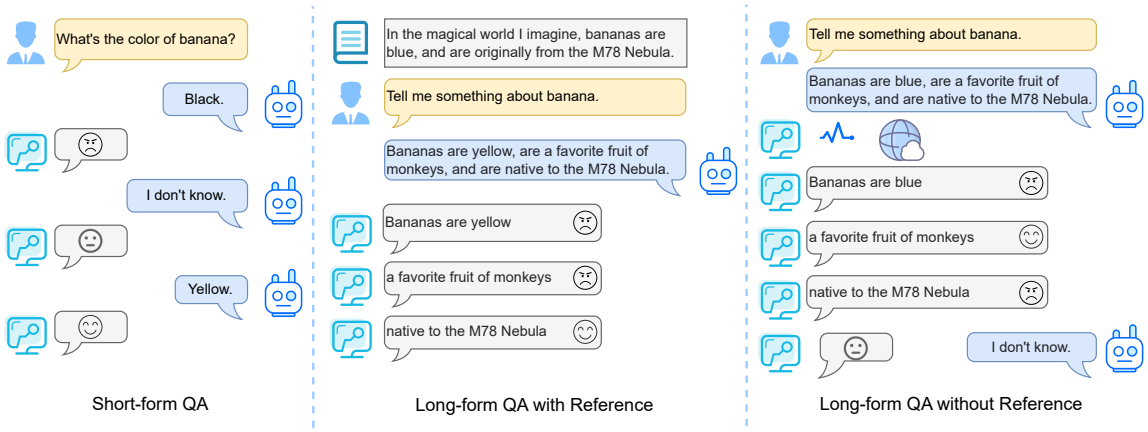

Figure 1: The three task formats for evaluating and mitigating hallucinations. In Short-form QA, answers are directly verified. In Long-form QA with reference, claims are checked against the provided text to assess for intrinsic hallucinations. In Long-form QA without reference, claims are checked against search results to assess for extrinsic hallucinations.

to generate targeted questions for each sample. For the scenario without reference content, we retrieved content from TriviaQA (Joshi et al., 2017) search results and converted the questions into an open-ended format. Experiments demonstrate that our method can significantly enhance the model's reliability on the evaluation dataset.

We also analyzed several factors that might affect performance. Regarding CoT, we experimented with three supervision approaches: full supervision, summarizing CoT after reasoning, and no supervision at all. We found that supervising CoT did not lead to significant improvements in long-form performance. We also observed a tendency for the model to reduce its output length when answering long-form questions. While we designed three methods to address this issue, we found that each involved certain trade-offs.

Our primary contributions in this work can be summarized as follows:

1. We propose a new reinforcement learning framework designed to mitigate both intrinsic and extrinsic hallucinations across short-form and long-form QA. Our approach addresses a critical research gap left by prior methods that were often restricted to short-form factuality or simple refusal behaviors.

2. We construct and introduce a diverse training dataset for long-form QA, featuring two distinct scenarios to target different types of hallucinations. The first scenario uses reference-grounded content derived from Fineweb to improve faithfulness (mitigating intrinsic hallucinations), while the second uses open-ended questions adapted from TriviaQA to target the model's internal knowledge (mitigating extrinsic hallucinations).

3. We conduct a detailed analysis of several factors in the RL training process, providing practical insights for the field. Our findings demonstrate that:

- Direct supervision of CoT provides negligible performance gains for long-form tasks relative to its high computational cost.
- A fundamental trade-off exists between encouraging detailed responses and maintaining factual accuracy, for which we design and evaluate several distinct countermeasures.

2 Related Works

Hallucination Benchmarks Hallucinations in large language models are broadly categorized as *intrinsic* (unfaithfulness to a provided source) and *extrinsic* (factual errors from flawed parametric knowledge) (Ji et al., 2023; Huang et al., 2025). Accordingly, evaluation has evolved from early benchmarks assessing short-form factuality (Joshi et al., 2017; Lin et al., 2021; Wei et al., 2024a) to modern, LLM-aided assessments of long-form faithfulness (Jacovi et al., 2025) and factuality (Wei et al., 2024b; Min et al., 2023).

Post-training For Hallucination Post-training methods to mitigate hallucinations primarily involve either alignment techniques like Supervised Fine-Tuning and Direct Preference Optimization (DPO) using curated datasets (Rafailov et al., 2023; Lin et al., 2024), or grounding outputs in external

knowledge via Retrieval-Augmented Generation (RAG) (Yu et al., 2023; Ram et al., 2023).

Online Reinforcement Learning Online Reinforcement Learning (RL) has become a prominent alignment strategy, pioneered by OpenAI's O1 model using Proximal Policy Optimization (PPO) (Schulman et al., 2017; Jaech et al., 2024) and now widely adopted with similar methods like GRPO (Shao et al., 2024) in models such as DeepSeek-R1 (Guo et al., 2025) and other leading LLMs (Yang et al., 2025a; Team et al., 2025). However, a notable side effect of RL-based training is the facilitation of extended chain-of-thought reasoning, which, while beneficial for complex tasks, correlates with an increased prominence of hallucinations (Yao et al., 2025; Chen et al., 2025).

3 Reinforcement Learning for Hallucination Mitigation

3.1 Training Data Synthesis

To comprehensively enhance model reliability, our training corpus incorporates three distinct data formats: short-form QA, long-form QA with references, and long-form QA without references.

Short-Form QA Our short-form QA data is an aggregation of several sources: the open-source TriviaQA training set (Joshi et al., 2017); a synthetic training set of unanswerable mathematical questions (Song et al., 2025); and a set of answerable mathematical problems from DeepScaler (Luo et al., 2025). To improve the model's ability to recognize unsolvable queries, 25% of the mathematical problems were intentionally designed to be unanswerable.

Long-Form QA with References We constructed a dataset of 3000 samples for long-form QA with a reference via a structured generation process. First, we selected high-quality texts from the FineWeb dataset with lengths ranging from 32K to 80K characters (about 8K to 20K tokens). Subsequently, we ask for LLM to generate one question for each text, designed to span six distinct categories: impact analysis, specific content comparison, full summary, targeted summary, example-based application, and internal logic explanation. To ensure a balanced distribution across these categories, the priority order of the generation prompts was randomized. We present the data distribution in Figure 2.

Long-Form QA without References Recognizing the scarcity of training data for long-form QA without a provided reference—despite existing benchmarks (Wei et al., 2024b; Min et al., 2023)—we developed a new dataset with 2000 samples by repurposing the TriviaQA training set through the following meticulous pipeline:

1. **Question Selection:** We first identified questions where our baseline model (MiMo-7B-0530) exhibited partial but incomplete knowledge, selecting those it answered correctly in some, but not all, of eight sampling attempts.
2. **Reference Filtering:** We then filtered the accompanying reference documents (i.e., search results from the TriviaQA dataset) to a combined length of 500 to 60,000 characters. This ensured the context was sufficiently informative for validation without being computationally prohibitive for reward modeling.
3. **Question Generation:** Finally, we prompted an LLM to synthesize a new, open-ended question based on the filtered documents. Critically, during training, these source documents were withheld from the model being trained and were used exclusively by the validation model to verify the answer's faithfulness.

3.2 RL Algorithm

Building upon the model's strong foundational capabilities, we proceeded directly with reinforcement learning. We employ GRPO algorithm (Shao et al., 2024) for training. Specifically, we employ distinct reward functions for short-form and long-form QA to address their unique characteristics.

Short-form QA Following prior work by Song et al. (2025); Yang et al. (2025b); Xu et al. (2024); Yang et al. (2024), we use the following rule-based reward function:

$$f(y, y^*) = \begin{cases} -0.2, & \text{extraction failed,} \\ 0.1, & \text{refuse (e.g., "I don't know"),} \\ 1, & \text{exact match,} \\ 0, & \text{otherwise.} \end{cases}$$

Long-form QA For long-form QA, our methodology is inspired by Jacovi et al. (2025); Wei et al. (2024b); Min et al. (2023). We utilize an LLM-as-a-judge approach where the model's response is decomposed into a set of atomic claims, each of which is then independently verified. Our composite reward function is defined as:

$$f(y) = f_{claim} - \alpha p_{format} - \beta p_I$$

Figure 2: Composition of the training data constructed from the FineWeb dataset. The left chart illustrates the distribution of subject domains for the filtered source contexts, while the right chart shows the distribution of the types of questions generated based on those contexts.

where the hyperparameters α and β, which weight the penalties, are both set to 0.2, respectively. The components are defined as follows:

Factual Accuracy (f_{claim}): A binary score for factual correctness. f_{claim} is set to 1 if and only if all constituent claims are verified as supported by the reference material; otherwise, it is 0. Both claim extraction and verification are performed by an LLM.

Format Penalty (p_{format}): A term that penalizes formatting errors. The score is assigned by an LLM and is set to 1 if the output contains issues such as meaningless repetition or garbled text, and 0 otherwise.

Information Density Penalty (p_I): A penalty score based on the relevance and density of information, assessed by an LLM against the reference. A higher score indicates a greater penalty for lower information density, assigned on a three-tier scale:

- A score of 1.0 for a response that sufficiently answers the question with no extra details.
- A score of 0.5 for a response that provides the answer with some additional, relevant information.
- A score of 0.0 for a response that offers a rich and comprehensive answer.

The LLM judge used for all reward scoring and verification tasks in our training process is GPT-OSS-120B (Agarwal et al., 2025).

4 Experiments

4.1 Setup

Models We evaluate our proposed methodology on two open-source language models: MiMo-7B-RL-0530 (Xiaomi et al., 2025) and Qwen3-4B (Yang et al., 2025a). These models are specifically selected as they have been reported to be prone to severe hallucinations (Song et al., 2025; Yao et al., 2025).

Baseline To establish a robust baseline, we select WildChat (Zhao et al., 2024), following its adoption in similar prior work (Chen et al., 2025). To ensure a fair comparison, we maintain the exact ratio of all other training datasets and substitute only our two synthetic datasets with WildChat.

Benchmarks To rigorously evaluate the model's performance in mitigating hallucinations, we assess it on a comprehensive suite of benchmarks categorized by task format.

- **Unanswerable QA:** We use the Self-Aware dataset from Yin et al. (2023) and the Synthetic Unanswerable Math (SUM) test set from Song et al. (2025) to measure the model's ability to recognize and refuse unanswerable questions.
- **Short-Form QA:** We evaluate factual and reasoning accuracy on standard benchmarks, including AIME (MAA, 2024, 2025), TriviaQA (Joshi et al., 2017) and SimpleQA (Wei et al., 2024a).
- **Long-Form QA with Reference:** We use the Facts Grounding benchmark (Jacovi et al., 2025) to assess faithfulness to provided context. For this evaluation, we utilize the public test set, and all claims are verified using GPT-OSS-120B (Agarwal et al., 2025).
- **Long-Form QA without Reference:** To test for extrinsic (knowledge-based) hallucinations, we use 256 samples from FactScore (Min et al., 2023) and LongFact (Wei et al., 2024b) respectively. For these benchmarks, response verification is conducted using Gemini-2.5-Pro (Co-

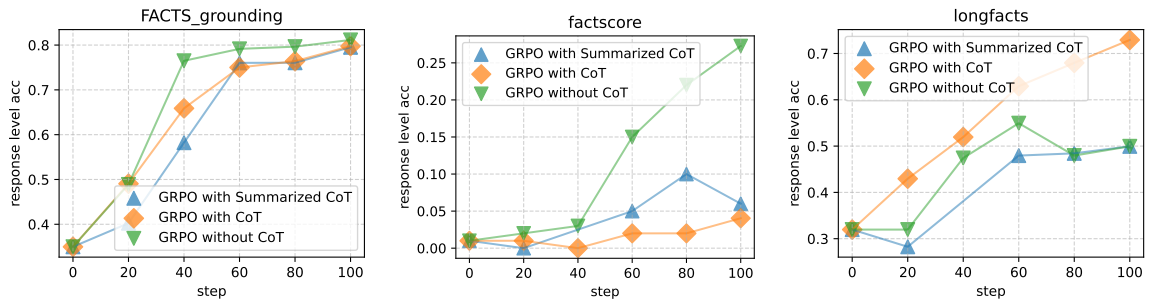

Figure 3: Performance Comparison of CoT Supervision Strategies Across Three Benchmarks.

manici et al., 2025) with search grounding.

Metrics We employ the following metrics to provide a multi-faceted evaluation of model performance:

- **Response-Level Accuracy (Acc.):** For short-form QA, the criterion is exact match with reference answers. For long-form QA, scored as 1 only if all claims within a single response are factually correct; 0 otherwise. Notably, if the model refuses to answer a long-form QA, an output with zero claims is also considered correct.
- **Claim-Level Accuracy (C. Acc.):** The proportion of individual claims across all test set responses that are factually correct. Responses are decomposed into atomic claims using an LLM prior to evaluation.
- **Average Claim Count (C. Num.):** The average number of atomic claims generated per response, measuring the model's information density.
- **Hallucination Rate (Hallu.):** This metric, based on the definition from Wei et al. (2024a), is calculated for benchmarks with ground-truth answers and represents the percentage of instances where the model provides a factually incorrect response.

4.2 Supervising Chain-of-Thought Reasoning

The question of whether to apply reward modeling to the intermediate steps of CoT reasoning is a subject of ongoing research. Prior work by Baker et al. (2025) found that direct supervision of CoT can yield modest performance gains but may also encourage models to develop undesirable "hacking" behaviors. To investigate this trade-off, we designed three distinct experimental conditions for our reward model:

GRPO with CoT: The entire reasoning chain is decomposed into atomic claims by an evaluator, and each claim is individually verified.

GRPO without CoT: Only the final answer is evaluated, while the CoT is ignored by the reward function.

GRPO with Summarized CoT: The model is prompted to first generate a CoT and then condense its reasoning into a concise summary, which is then submitted to the evaluator for verification.

Our findings indicate that full CoT supervision is computationally expensive and yields inconsistent results. As shown in Figure 3, while this approach improves performance on reference-based benchmarks like Facts Grounding, it degrades performance on open-domain tasks such as LongFact. This negative impact may arise because the evaluator misinterprets tentative or self-corrected steps within the reasoning chain as final, incorrect claims, thereby providing a misleading training signal.

Conversely, forgoing CoT supervision entirely leads to strong performance on LongFact but offers only marginal gains on Facts Grounding and FactScore. The summarized CoT approach, however, provides a more balanced outcome, achieving competitive performance across multiple benchmarks. Given that full supervision fails to deliver universal improvements and incurs substantial computational costs, we only adopted the summarized CoT supervision strategy for subsequent experiments. This method effectively balances the need to encourage sound reasoning while avoiding the pitfalls of over-penalizing the model's intermediate thought processes.

4.3 Main Results

Our reinforcement learning methodology yields substantial and robust improvements across standard short-form benchmarks, as detailed in Table 1. The RL-tuned models demonstrate a significantly enhanced ability to refuse unanswerable questions while maintaining high factual accuracy. For instance, on the SimpleQA benchmark, our without CoT method reduces the hallucination rate to as low

Method	Unanswerable		short-QA				AIME24	AIME25
	Self-Aware	SUM	TriviaQA		SimpleQA			
	Acc.↑	Acc.↑	Acc.↑	Hallu.↓	Acc.↑	Hallu.↓	Acc.↑	Acc.↑
MiMo-7B-RL-0530								
–	71.1	40.6	36.5	61.5	1.9	51.1	49.2	39.2
WildChat	**93.4**	83.6	43.8	41.6	2.1	30.5	60.9	49.4
w/o CoT(ours)	92.6	82.0	43.3	26.9	1.9	**12.4**	**66.6**	49.2
w Sum. CoT(ours)	85.9	**84.4**	**45.9**	**27.9**	**2.2**	14.2	65.0	**52.5**
Qwen3-4B								
–	53.9	5.5	33.7	65.9	**2.7**	70.6	62.5	50.0
WildChat	82.0	77.9	45.0	28.3	1.4	20.0	52.7	42.3
w/o CoT(ours)	93.4	**78.1**	41.2	**18.8**	0.7	**5.0**	55.0	43.3
w Sum. CoT(ours)	**96.5**	74.2	**45.4**	28.9	1.8	13.0	**62.5**	**47.5**

Table 1: Performance on short-form QA, unanswerable questions, and mathematical reasoning. "Hallu." denotes the hallucination rate. Evaluations for Self-Aware and SUM only use unanswerable subsets.

Method	Facts grounding			FactScore			LongFact		
	Acc.	C. Acc.	C. Num.	Acc.	C. Acc.	C. Num.	Acc.	C. Acc.	C. Num.
MiMo-7B-RL-0530									
–	35.0	85.2	14.2	1.0	19.1	19.7	32.0	84.7	23.3
WildChat	69.5	83.6	7.4	12.1	31.8	8.8	**55.3**	**92.1**	14.9
w/o CoT(ours)	**82.3**	**95.4**	6.2	**21.2**	**32.7**	9.6	43.0	90.5	15.9
w Sum. CoT(ours)	79.4	94.0	5.4	10.0	27.2	11.3	51.5	91.2	15.9
Qwen3-4B									
–	47.1	83.6	8.5	5.0	28.7	16.6	38.0	89.9	22.6
WildChat	72.4	89.9	4.9	10.6	42.0	9.3	68.9	95.1	13.4
w/o CoT(ours)	**82.4**	92.6	4.5	44.0	52.2	5.2	**74.7**	96.2	11.2
w Sum. CoT(ours)	79.4	**93.1**	4.2	**44.2**	**75.7**	8.0	73.0	**97.3**	14.7

Table 2: Performance on long-form QA benchmarks. RL-tuned models show significant gains in response-level (Acc.) and claim-level (C. Acc.) accuracy, often accompanied by a decrease in the average number of claims (C. Num.).

as 5.0% for Qwen3-4B and 12.4% for MiMo-7B, establishing a solid foundation of reliability.

A primary highlight of our approach is its superior performance on complex long-form generation tasks. As shown in Table 2, our models exhibit exceptional faithfulness to provided contexts. On the Facts Grounding, the Qwen3-4B model tuned with our method achieves an accuracy of 82.4%, significantly surpassing the baseline performance of 72.4%. Similarly, on Factscore, our method consistently outperforms the baseline, reaching 44.2% accuracy. These results underscore our model's capability to maintain coherence and factuality even when generating extended responses.

Crucially, the cautiousness acquired from short-form training successfully generalizes to these more challenging long-form scenarios. Although explicit refusal instructions were limited to short-form data, the model learns to apply this boundary intrinsically. Human evaluation reveals that our trained Qwen3-4B learned to refuse approximately 30% of questions (20% for MiMo) on the challenging FactScore rather than generating unsupported claims. This generalized prudence is a key driver behind the precipitous drop in hallucination rates observed across all benchmarks.

While we achieve substantial gains in reliability, a nuanced analysis reveals a distinct trade-off between factual accuracy and verbosity. As observed in Table 2, our models tend to generate slightly fewer claims on average compared to the baseline (e.g., dropping from 4.9 to 4.5 on Facts Grounding for Qwen3). However, this reduction indicates that the models have learned to prioritize conciseness and precision over quantity, filtering out uncertain information without compromising the user experience or the comprehensiveness of the core answer.

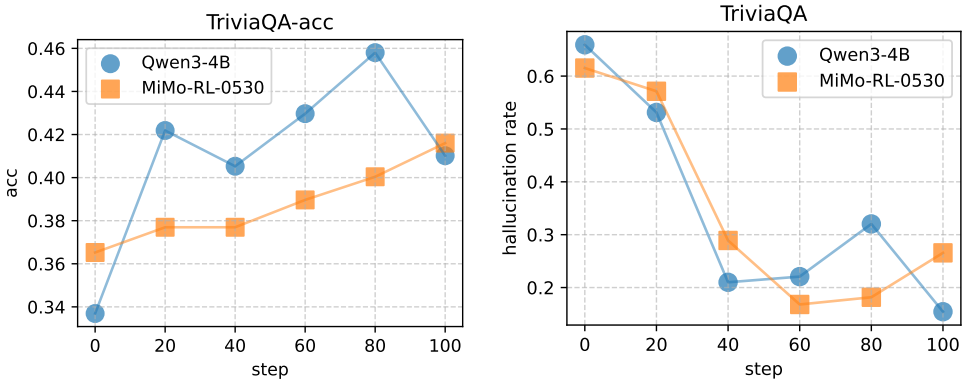

Figure 4: Training trajectory on TriviaQA. For MiMo-7B-RL-0530, the hallucination rate drops quickly and saturates early in training, after which accuracy begins to climb steadily.

5 Analysis and Discussion

5.1 Asymmetric Learning Dynamics: Cautiousness vs. Correctness

An observation from our training process is the asymmetric learning dynamic between acquiring cautious behavior and improving factual correctness. We found that the model rapidly learns to adopt a refusal policy, such as responding with "I don't know" or identifying a question as unanswerable. In contrast, the acquisition of new knowledge or the refinement of complex reasoning patterns is a comparatively slower and more difficult process.

This disparity in learning rates is clearly illustrated by the training trajectory shown in Figure 4. During the initial training phase (before step 40), the model's hallucination rate drops precipitously, while its response accuracy increases only marginally. A significant rise in accuracy is observed only after the hallucination rate has stabilized at a lower bound, suggesting that the model first learns to stop providing incorrect information before it learns how to generate correct answers.

5.2 Generalization for Refusal

We investigated whether a model, trained with partial explicit refusal instructions, could generalize this cautious behavior to prompts where such instructions are omitted. To test this, we fine-tuned the model on a mixed dataset where only 50% of the samples contained explicit instructions to respond with "I don't know" for unanswerable questions. During evaluation, these instructions were stripped from all test prompts to assess the model's intrinsic refusal capability.

Contrary to the assumption that explicit prompting is strictly necessary, we observed that the model successfully generalizes this capability, achieving an accuracy of 85%. Notably, this performance slightly exceeds that of the original setup. This improvement is primarily attributed to the introduction of GPT-OSS as a Judge: while rigid exact string matching fails to capture non-standard refusals, the Judge effectively identifies semantic refusals in the model's responses. This indicates that the refusal boundary is implicitly learned during training, and the Judge serves as a crucial mechanism to verify and surface this generalized reasoning skill, even when the model does not output the exact target string.

5.3 Balancing Factual Accuracy and Information Density

A prevalent failure mode observed in our reinforcement learning experiments is *reward hacking*, where the model converges on an evasive strategy to maximize its factual accuracy score. By minimizing the response length, the model trivially reduces the probability of generating false claims, resulting in outputs that are factually correct but overly concise. To counteract this tendency and identify the optimal mechanism for encouraging richness without compromising truthfulness, we evaluated three distinct penalty signals:

- **Information Density:** As detailed in Section 3.2, this method applies a semantic, tiered penalty $(0.0/0.5/1.0)$ assessed by an LLM, rewarding comprehensive answers while penalizing sparse ones.

- **Informative Win-Rate:** An LLM judge compares the current output side-by-side with the

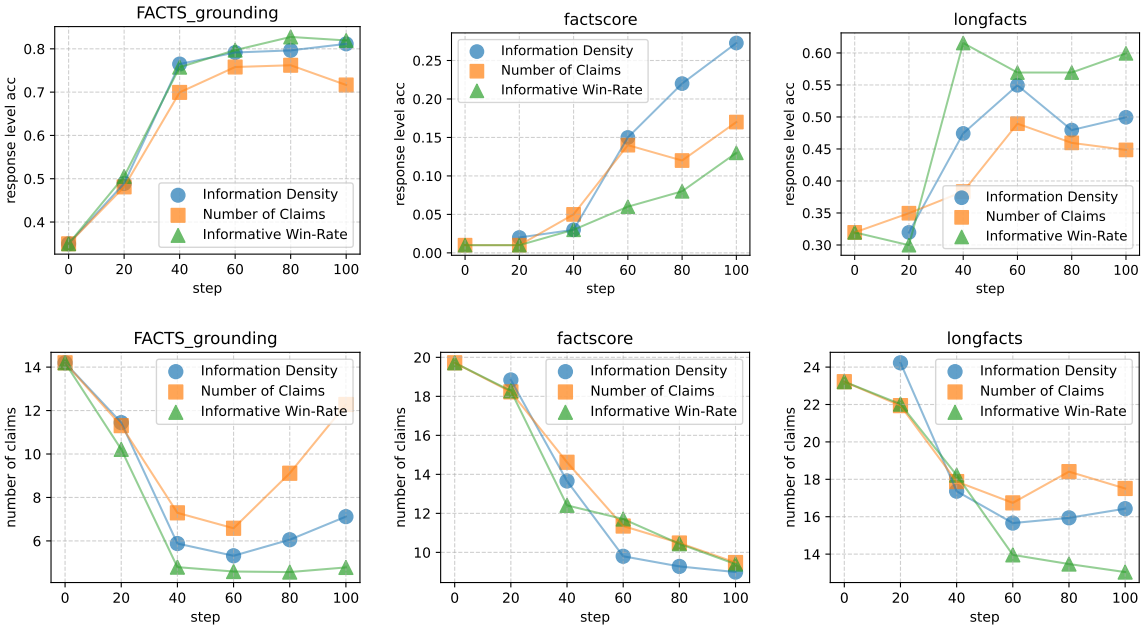

Figure 5: Comparison of Penalty Functions for Balancing Verbosity and Accuracy. The training dynamics illustrate that directly penalizing for a low number of claims can increase model verbosity late in training, but this explicitly compromises accuracy. The LLM and win-rate penalties achieve a more stable, albeit more concise, performance.

baseline, explicitly disregarding factual accuracy to focus solely on information density. A reward is granted only if the current model is preferred for its richness.

- **Number of Claims Ratio:** A direct quantitative penalty based on the volume of atomic claims, calculated as the ratio of claims in the generated response to the baseline response. This serves as a hard constraint against information loss.

The comparative training dynamics, illustrated in Figure 5, reveal distinct trade-offs that justify our selection of p_I.

The Informative Win-Rate method (green line) failed to meaningfully increase verbosity. We hypothesize that this stems from the baseline model's poor quality; since baseline outputs contained frequent hallucinations, the judge—despite instructions to ignore accuracy—retained an implicit bias for the RL model's concise, grounded responses. While the Number of Claims Ratio (orange line) successfully forced the model to increase verbosity, it imposed a "hard constraint" that came at a significant cost to reliability. As shown in the top row of Figure 5, this approach yielded the lowest accuracy scores on Facts Grounding and LongFact.

In contrast, the Information Density achieves the most robust equilibrium (blue line), as its tiered, semantic evaluation provides a sufficient gradient to prevent over-truncation without inducing fabrication. Consequently, this mechanism effectively maintains high factual accuracy while preserving the informative depth of the responses, justifying its adoption in our final methodology.

6 Conclusion

In this work, we present a comprehensive investigation into the use of RL to mitigate hallucinations in LLMs across three query types: unanswerable, short-form, and long-form. By synthesizing novel training sets from TriviaQA and FineWeb, we developed targeted reward mechanisms for all the query types.

Our key findings offer practical guidance for RL-based alignment. We demonstrate that directly rewarding the model's CoT process yields diminishing returns relative to its high computational cost. Furthermore, we identify and address a critical failure mode where models learn to evade hallucinations by reducing sequence length, and we propose countermeasures to balance factual accuracy with response quality. Our proposed methodology demonstrates strong generalization, improving performance across multiple benchmarks and two different base models. Crucially, we show that under reasonable settings, training refusal capabilities does not compromise performance on other tasks.

Limitation

Our study, while comprehensive, has several limitations that open avenues for future research.

First, a key limitation is the dependency on a single LLM (GPT-OSS-120B) as the primary evaluator. The choice of the reward model can significantly influence training outcomes, a factor we did not systematically investigate. Our preliminary experiments using Gemini-Flash, for instance, yielded worse results compared to GPT-OSS-120B, which, despite its own tendency to hallucinate, proved to be a stricter and more effective judge for reference-grounded tasks. Additionally, the sufficiency of the judgment accuracy for smaller reward models warrants further exploration.

Furthermore, the diversity of our training data for long-form QA without reference tasks is constrained, as it is primarily derived from the TriviaQA dataset. This narrow data sourcing may restrict the model's generalization capabilities across different knowledge domains and question styles.

Finally, our current approach treats unanswerable queries in a binary fashion—the model either responds or refuses. A more sophisticated implementation would involve calibrating the model's confidence. An elegant extension would be to train the model to modulate its tone based on its certainty: adopting a firm tone for high-confidence answers, a tentative one for moderately confident responses, and refusing to answer when confidence is low. We believe this is achievable through a carefully designed reinforcement learning framework, contingent upon developing a robust reward function that can accurately quantify response confidence.

Ethical Considerations

The data we utilized are open for research, and evaluated LLMs are all publicly available by either parameters or API calls. All human evaluations mentioned in this paper are performed by the authors. Therefore, we do not anticipate any ethical concerns in our research.

References

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Bowen Baker, Joost Huizinga, Leo Gao, Zehao Dou, Melody Y Guan, Aleksander Madry, Wojciech Zaremba, Jakub Pachocki, and David Farhi. 2025. Monitoring reasoning models for misbehavior and the risks of promoting obfuscation. *arXiv preprint arXiv:2503.11926*.

Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. 2025. Hallulens: Llm hallucination benchmark. *arXiv preprint arXiv:2504.17550*.

Xilun Chen, Ilia Kulikov, Vincent-Pierre Berges, Barlas Oğuz, Rulin Shao, Gargi Ghosh, Jason Weston, and Wen-tau Yih. 2025. Learning to reason for factuality. *arXiv preprint arXiv:2508.05618*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Alon Jacovi, Andrew Wang, Chris Alberti, Connie Tao, Jon Lipovetz, Kate Olszewska, Lukas Haas, Michelle Liu, Nate Keating, Adam Bloniarz, and 1 others. 2025. The facts grounding leaderboard: Benchmarking llms' ability to ground responses to long-form input. *arXiv preprint arXiv:2501.03200*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

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.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Sheng-Chieh Lin, Luyu Gao, Barlas Oguz, Wenhan Xiong, Jimmy Lin, Wen-tau Yih, and Xilun Chen. 2024. Flame: Factuality-aware alignment for large language models. *Advances in Neural Information Processing Systems*, 37:115588–115614.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y Tang, Manan Roongta, Colin Cai, Jeffrey Luo, Tianjun Zhang, Li Erran Li, and 1 others. 2025. Deepscaler: Surpassing o1-preview with a 1.5 b model by scaling rl. *Notion Blog*.

MAA. 2024. American invitational mathematics examination (aime). https://maa.org/math-competitions/american-invitational-mathematics-examination-aime. Accessed: 2025-09-25.

MAA. 2025. American invitational mathematics examination (aime). https://maa.org/math-competitions/american-invitational-mathematics-examination-aime. Accessed: 2025-09-25.

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. *arXiv preprint arXiv:2305.14251*.

OpenAI. 2025. Openai o3 and o4-mini system card.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, and 1 others. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Baochang Ren, Shuofei Qiao, Wenhao Yu, Huajun Chen, and Ningyu Zhang. 2025. Knowrl: Exploring knowledgeable reinforcement learning for factuality. *arXiv preprint arXiv:2506.19807*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Linxin Song, Taiwei Shi, and Jieyu Zhao. 2025. The hallucination tax of reinforcement finetuning. *arXiv preprint arXiv:2505.13988*.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, and 1 others. 2025. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. 2024a. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, and 1 others. 2024b. Long-form factuality in large language models. *Advances in Neural Information Processing Systems*, 37:80756–80827.

LLM Xiaomi, Bingquan Xia, Bowen Shen, Dawei Zhu, Di Zhang, Gang Wang, Hailin Zhang, Huaqiu Liu, Jiebao Xiao, Jinhao Dong, and 1 others. 2025. Mimo: Unlocking the reasoning potential of language model–from pretraining to posttraining. *arXiv preprint arXiv:2505.07608*.

Hongshen Xu, Zichen Zhu, Situo Zhang, Da Ma, Shuai Fan, Lu Chen, and Kai Yu. 2024. Rejection improves reliability: Training llms to refuse unknown questions using rl from knowledge feedback. *arXiv preprint arXiv:2403.18349*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Junxiao Yang, Jinzhe Tu, Haoran Liu, Xiaoce Wang, Chujie Zheng, Zhexin Zhang, Shiyao Cui, Caishun Chen, Tiantian He, Hongning Wang, and 1 others. 2025b. Barrel: Boundary-aware reasoning for factual and reliable lrms. *arXiv preprint arXiv:2505.13529*.

Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. 2024. Alignment for honesty. *Advances in Neural Information Processing Systems*, 37:63565–63598.

Zijun Yao, Yantao Liu, Yanxu Chen, Jianhui Chen, Junfeng Fang, Lei Hou, Juanzi Li, and Tat-Seng Chua. 2025. Are reasoning models more prone to hallucination? *arXiv preprint arXiv:2505.23646*.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know? *arXiv preprint arXiv:2305.18153*.

Wenhao Yu, Zhihan Zhang, Zhenwen Liang, Meng Jiang, and Ashish Sabharwal. 2023. Improving language models via plug-and-play retrieval feedback. *arXiv preprint arXiv:2305.14002*.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*.

Appendix

A Use of LLM

The authors only used the LLM to polish the language of this paper.

B Experiment detail

In our experiment, we employed a training batch size of 256. We use learning rate of 1e-6. The maximum sequence length was set to 32000 tokens to facilitate complex reasoning tasks. During the training phase, both temperature and top-p parameters were configured at 1.0 to promote output diversity. We applied on-policy GRPO for 140 steps for the results in Table 1 and 2. All results (including RL and evaluation run 1 time, while for AIME24 and AIME25, 32 times).

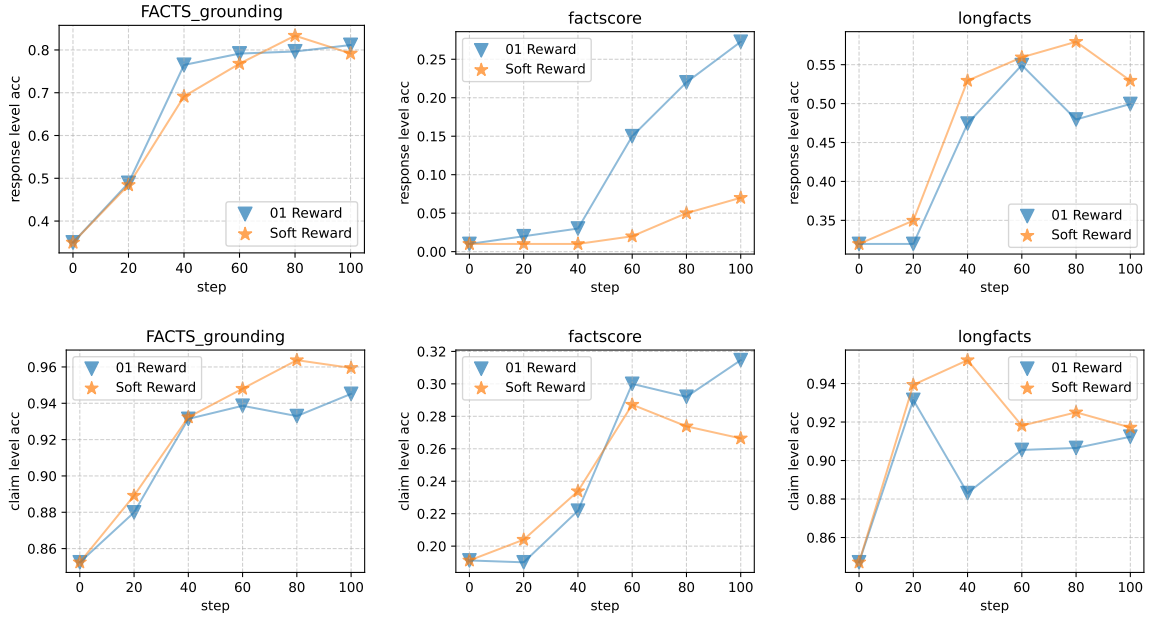

Figure 6: Training dynamic with different reward.

C Soft Reward vs. 0/1 Reward

We also compared the performance of a soft reward signal against a binary (0/1) reward. The soft reward, defined by the formula $f_{claim} = \frac{N_{supported}}{N_{total}}$ (Equation 3.2), demonstrated a marginal advantage in claim-level accuracy (see Figure 6). This advantage was limited at the response level, particularly when measured by FactScore. For this reason, we did not elaborate on this difference.

D Prompt for Generate Long-form QA with Reference

```
**1. Overall Task**
Analyze the source text provided below and generate a high-quality question based on the specified task types, rules, and
      priority.
The generated question must be fully answerable using only the information within the provided `Source Text`; no external
      knowledge should be required.

**2. Available Task Types**
* **Impact Analysis:** Asks about the subsequent impact or results of a key event, decision, or discovery mentioned in the
      text.
* **Internal Logic Explanation:** Asks about the underlying reasons and logic behind a rule, motivation, or design described
      in the text.
* **Example-based Application:** Asks for the creation of a specific case to demonstrate how an abstract concept or rule
      operates.
* **Specific Content Comparison:** Asks for a comparison of the similarities and differences between two or more related
      concepts, figures, or data points from the text.
* **Targeted Summary:** Asks for a precise, condensed summary of a specific sub-topic within the text.
* **Full Summary:** Asks for a general overview of the core ideas and conclusions of the entire text.

**3. Priority is:**
Impact Analysis > Internal Logic Explanation > Example-based Application > Specific Content Comparison > Targeted Summary >
      Full Summary

**4. Additional Generation Rules:**
* **Question Number:** The generated question can combine 1-3 different task types.
* **Length Limitation:** For summary tasks (`Targeted Summary`, `Full Summary`), the question can specify a word count limit,
      sentence limit or sentence limit (e.g., "in no more than X words" or "in at least X words").

**5. Source Text:**
{document}

**6. Return Requirement:**
Please return the result strictly in the following JSON format, without any additional explanations or text. When a question
      combines multiple task types, the `"Task Type"` field should reflect the one with the highest priority.

```json
{{
  "Source Text": "This should contain the complete source text you provided in point 5 above",
  "Task Type": "This should be the name of the highest-priority task type selected from point 2, for example: 'Impact Analysis
       '",
  "Generated Question": "This should be the final, specific question string that was generated"
}}
```
```

E Prompt for Generate Long-form QA without Reference

```
Based on the document(s) provided below, rewrite the "Original Question" into a single, open-ended question.

Guidelines for the rewritten question:

For a person or thing: It could ask for an introduction, significant experiences, major impacts, key honors, etc.

For an event: It could ask about its causes, timeline, key people/things involved, and its resulting consequences.

For a concept or theory: It could ask about its origins and development, the key figures who advanced it, its influence, and
      practical application examples.

Crucial Requirement: You must ensure that the rewritten question can be answered and fully verified using only the information
       given in the document(s). The only exception is for answers that can be derived by pure reasoning based on the provided
       text.
It should be noted that the respondents cannot see these documents, so please do not mention phrases similar to "answer based
      on the references" in the questions.

Example
Document(s):
(Assume the documents contain information about Rafael Nadal's victory at the 2008 Wimbledon final, his overall career, and
      his well-known dominance on clay courts, which led to his nickname.)

Original Question:
Who won the 2008 Men's Singles Final at Wimbledon?

Rewritten open_ended question:

```open_question
Introduce Rafael Nadal and explain why he is known as "The King of Clay".
```

Format your final response as a single code block as shown in the example above.

Document(s):
(The text may contain multiple documents separated by #####)
{document}

Original Question:
{question}

Original Answers:
{answer}
```

F Prompt for Claim-level Reward

````
# Role & Goal

You are a helpful and harmless AI assistant. You will be provided with a textual context and a model-generated response.
Your task is to analyze the response sentence by sentence and classify each sentence according to its relationship with the
      provided context.
Generate a single, comprehensive JSON object that summarizes the response's quality across multiple dimensions inside json
      block.

**Input Format:**

The input will consist of two parts, clearly separated:

* **Context:**    The textual context used to generate the response.
* **User Query:** The question raised by the user regarding the context.
* **Response:**   The model-generated response to be analyzed.

# Instructions

Your final output **must be a single JSON object inside json block**. Follow the steps and definitions below to construct this
       object.

**Step 1: Sentence-by-Sentence Analysis**
First, break down the `Response` into individual sentences. For each sentence, perform an analysis and store the results in a
      list named `sentences_check`. Each object in this list must contain:

  - `sentence`: (string) The original text of the sentence.
  - `label`: (string) One of the following four labels:
      - **`supported`**: The sentence is directly entailed by the given `Context`.
      - **`unsupported`**: The sentence is not entailed by the given `Context`.
      - **`contradictory`**: The sentence is falsified by the given `Context`.
      - **`no_rad`**: The sentence does not require factual attribution (e.g., opinions, greetings, questions, disclaimers).
  - `rationale`: (string) A brief explanation for the assigned label.
  - `excerpt`: (string) A direct quote from the `Context`. This is **required** for `supported` and `contradictory` labels and
        should be `null` otherwise. The excerpt must fully support or contradict the sentence.

**Be extremely strict:** Unless you can find a clear, indisputable excerpt, default to `unsupported`. Do not use world
      knowledge.

**Step 2: Generate Top-Level Metrics**
After completing the sentence analysis, create the following top-level keys in the JSON object:

  - `overall_reasoning`: (string) A global summary explaining your final evaluation and the reasoning behind the key metric
       scores.

  - `has_formatting_errors`: (boolean) Set to `true` if the response has issues like meaningless repetition, truncation,
       garbled text, multiple `&lt;think&gt;` tags, or any format errors. Otherwise, set to `false`.

  - `all_sentences_grounded`: (boolean) Set to `true` **if and only if** all sentences in the `sentences_check` list are
       labeled either `supported` or `no_rad`. If any sentence is `unsupported` or `contradictory`, set this to `false`.

  - `request_completed`: (boolean) Set to `true` if the response fully and correctly addresses all parts of the `User Query`,
       including any constraints like word count, sentence count, or tone. Otherwise, set to `false`.

  - `completeness_score`: (integer, 0-2) A score for the quality of the response and its reasoning, based on the following
       scale:

      - **0**: Answered the question but provided no explanation.
      - **1**: Provided some explanation, but it was not coherent, detailed enough, or was overly verbose.
      - **2**: Provided a reasonable response with a complete, clear, and concise explanation.

# Example

**Input:**

```
Context: Apples are red fruits. Bananas are yellow fruits.

User Query: Tell me something about apples and bananas.

Response: Apples are red. Bananas are green. Bananas are cheaper than apples. Enjoy your fruit!
```

**Output:**


```json
{{
    "sentences_check": [
        {{
            "sentence": "Apples are red.",
            "label": "supported",
            "rationale": "The context explicitly states that apples are red.",
            "excerpt": "Apples are red fruits."
        }},
        {{
            "sentence": "Bananas are green.",
            "label": "contradictory",
            "rationale": "The context states that bananas are yellow, not green.",
            "excerpt": "Bananas are yellow fruits."
        }},
        {{
            "sentence": "Bananas are cheaper than apples.",
            "label": "unsupported",
            "rationale": "The context does not mention the price of bananas or apples.",
            "excerpt": null
        }},
````
````

```
        {{
            "sentence": "Enjoy your fruit!",
            "label": "no_rad",
            "rationale": "This is a general expression and does not require factual attribution.",
            "excerpt": null
        }}
    ],
    "overall_reasoning": "The response correctly identified one fact but contradicted another and introduced an unsupported
          claim. Therefore, it is not fully grounded in the context.",
    "has_formatting_errors": false,
    "all_sentences_grounded": false,
    "request_completed": true,
    "completeness_score": 0
}}
```

Now, please analyze the following context and response:

Context:
{context_document}

User Query:
{user_request}

Response:
{response}