The Impact of Negated Text on Hallucination with Large Language Models

Jaehyung Seo[1], **Hyeonseok Moon**[1] **and Heuiseok Lim**[1†]
[1]Department of Computer Science and Engineering, Korea University
{seojae777,glee889,limhseok}@korea.ac.kr

Abstract

Recent studies on hallucination in large language models (LLMs) have been actively progressing in natural language processing. However, the impact of negated text on hallucination with LLMs remains largely unexplored. In this paper, we set three important yet unanswered research questions and aim to address them. To derive the answers, we investigate whether LLMs can recognize contextual shifts caused by negation and still reliably distinguish hallucinations comparable to affirmative cases. We also design the NegHalu dataset by reconstructing existing hallucination detection datasets with negated expressions. Our experiments demonstrate that LLMs struggle to detect hallucinations in negated text effectively, often producing logically inconsistent or unfaithful judgments. Moreover, we trace the internal state of LLMs as they process negated inputs at the token level and reveal the challenges of mitigating their unintended effects.

1 Introduction

The rapid advancement of large language models (LLMs) continues to drive the release of diverse open-source models capable of performing a wide range of tasks (Touvron et al., 2023; Jiang et al., 2023; Team et al., 2024). As these models become more prevalent, the ability to distinguish whether generated outputs contain hallucinations is becoming increasingly critical (Magesh et al., 2024). Detecting hallucination involves identifying content that is either contextually unfaithful or contradictory to real-world facts and assessing the truthfulness of such outputs (Ji et al., 2023a; Zhang et al., 2023; Huang et al., 2023a).

Recent research on hallucination detection actively focuses on improving the reliability of LLMs by identifying their limitations and refining models based on insights into hallucinated outputs (Manakul et al., 2023; Jiang et al., 2024; Chen et al., 2024). However, there is a limited exploration of how negated text affects hallucination detection with LLMs. Negated text, which includes negation markers (e.g., "not," "never," "no," "without"), is commonly used in everyday communication (Gubelmann and Handschuh, 2022; Hossain et al., 2022). Although these markers are typically single tokens, they exert a disproportionately large influence on the overall processing factuality of a sentence, fundamentally altering its meaning (Vanek et al., 2024). However, unlike humans, LLMs struggle to effectively handle negation and infer contextual meaning (Truong et al., 2023; Ye et al., 2023). Moreover, negative knowledge can introduce hallucinations into commonsense reasoning (Chen et al., 2023; Seo et al., 2024), as LLMs tend to misrepresent negation as a faulty logical operator, leading to severe hallucinations (Bhar and Asher, 2024).

In this paper, we present an initial exploration of

Figure 1: Examples illustrating how negation can flip the hallucination label. The negated response introduces or resolves contradictions with the given knowledge.

† Corresponding Author

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 13566–13584
November 4-9, 2025

how negated text influences hallucination detection in LLMs. We propose three open research questions and work towards answering them.

- **RQ1.** Can LLMs distinguish between hallucinations and faithful statements in negated text as effectively as in affirmative text?
- **RQ2.** Can the model internally recognize differences caused by negation when detecting hallucinations?
- **RQ3.** Can targeted intervention strategies improve hallucination detection in the negated text?

To address these research questions, we investigate whether LLMs can recognize contextual shifts and reliably detect hallucinations in the presence of negated text. As illustrated in Figure 1, we conduct this analysis by reconstructing existing hallucination detection benchmarks with negated expressions and introducing NegHalu, a dataset in which hallucination labels are newly assigned to account for the effects of negation.

Our experiments show that Llama-2-7B (Touvron et al., 2023), Llama-3-8B (AI@Meta, 2024), and Mistral-7B-v0.3 (Jiang et al., 2023) exhibit performance degradation in 17 out of 18 post-negated hallucination detection cases. Furthermore, when the same content is expressed in both affirmative and negated forms, models tend to exhibit a bias toward classifying post-negated scenarios as hallucinations rather than faithful statements. We demonstrate that the influence of negated text on hallucination detection extends across multiple tasks, including question answering (QA), dialogue, summarization, and completion, indicating its task-agnostic. Moreover, this phenomenon is observed across a diverse range of domains, including 10 general topics, science, and autobiographies, suggesting its broad applicability. For in-depth analyses, we employ lens observation to examine the internal states of LLMs during hallucination detection. Furthermore, we examine approaches to alleviate the impact of negated text without resorting to unrealistic external modules or excessive parameter modifications, assessing the effectiveness of in-context learning (Brown et al., 2020), Chain-of-Thought (CoT) reasoning (Wei et al., 2022), and knowledge editing (Fang et al., 2024) as potential solutions.

2 Related Work

Negated Text in LLMs Handling negated text has been a long-standing challenge in NLP. Minsky (1997) emphasized the importance of understanding negated expressions and meanings, highlighting the need to integrate negation into NLP systems. Building on this foundation, Morante et al. (2011) analyzed how negation operates within the text, providing the essential groundwork for subsequent studies. Further exploration aimed to understand the effects of negation on semantic structures and meaning (van Son et al., 2016; Khandelwal and Sawant, 2020). Kassner and Schütze (2020) and Hossain et al. (2022) demonstrated that models frequently overestimate their confidence in predictions, leading to errors when processing negated inputs. Arnaout et al. (2022) and Chen et al. (2023) revealed that negated knowledge can introduce biases into LLMs, further complicating their performance. Truong et al. (2023) observed that even as the model size increases, the ability to effectively handle negation does not necessarily improve. Moreover, Ye et al. (2023) found that negation can cause significant performance drops, even when advanced strategies like chain-of-thought reasoning are employed. Most recently, Bhar and Asher (2024) highlighted how negation can lead to unique types of hallucinations in tasks such as natural language inference.

Hallucination in LLMs The issue of hallucination in LLMs has gained increasing importance as these models are applied to various NLP tasks. Hallucination occurs when the output generated by an LLM either lacks logical consistency with the input or contradicts real-world facts (Ji et al., 2023a; Huang et al., 2023a). This phenomenon has been observed across a range of tasks, including machine translation (Dale et al., 2023; Guerreiro et al., 2023), summarization (Zhao et al., 2020; Choubey et al., 2023), and dialogue generation (Ji et al., 2023b). It is particularly problematic in high-stakes domains that demand high reliability, such as law (Magesh et al., 2024), medicine (Farquhar et al., 2024), and science (Dong et al., 2024b). To address hallucinations, recent research has advanced both detection and mitigation techniques (Ji et al., 2023a; Huang et al., 2023a; Zhang et al., 2023). Detection strategies range from word-level and sentence-level analysis (Huang et al., 2023b; Yang et al., 2023) to self-verification via sampling (Manakul et al., 2023) and methods em-

ploying eigen-scores (Chen et al., 2024). For mitigation, approaches include employing decoding strategies with contrasting layers (Chuang et al., 2023), leveraging knowledge graph embeddings (Ji et al., 2023b), and fine-tuning model parameters based on data quality (Choubey et al., 2023). In this paper, we explore the unanswered scope of hallucination and address the lack of research on how negated text affects hallucination phenomena in LLMs. We analyze LLM performance on pre-negated and post-negated statements to identify the underlying causes of performance shifts and investigate strategies for mitigating these effects.

3 NegHalu

Source Datasets We utilize three hallucination detection datasets: HaluEval (Li et al., 2023), BamBoo (Dong et al., 2024b), and SelfCheckGPT-WikiBio (Manakul et al., 2023). Each dataset is selected for its relevance to evaluating hallucination phenomena across various tasks. For detailed statistics and descriptions of each dataset and its subsets, please refer to Appendix C.

Post Negation To analyze the impact of negated text on hallucination detection, we introduce a post-negation transformation applied to key fields in each dataset that are crucial for determining the presence of hallucination. This transformation compels the model to re-evaluate its predictions in the context of post-negated input. The following fields are transformed into their post-negated text: the `'answer'` field for Halu-QA, the `'response'` field for Halu-Dialogue, the `'summary` field for Halu-Sum, the `'hypothesis'` field for AbsHallu and SenHallu, and the `'generated text'` field for SelfCheckGPT-WikiBio. Table 1 illustrates the prompt templates used to generate post-negated texts and corresponding new labels. To create the negated versions of the datasets, we utilize the GPT-4 omni (`gpt-4o-2024-08-06`) (OpenAI, 2023) API in a two-round process.

As described in Table 1, in the first round (**Round 1**), we instruct the model to generate post-negated texts based on the given context and knowledge, aiming to maintain logical consistency independently. In this setting, "logical consistency" means that the insertion of a negation marker should transform the original pre-negated text, whether it is hallucinated or factual, so that the resulting meaning coherently aligns with the new label. This requirement is explicitly stated in the instruction: "*Your task is to restructure [C] into a negative statement [NEW C] by adding 'not' ONCE so that [Label] changes accordingly*," which is intended to ensure that the negated text both reverses the original meaning and remains consistent with the assigned label. Additionally, new labels are systematically assigned by considering the changes in the context of pre-negated examples, a process that requires modifying the original hallucination statuses and corresponding labels. This approach is designed to promote a definitive change in the labels of post-negated texts and to maximize the generation of examples in which the insertion of negation results in a logically valid label shift.

Data Verification To validate the plausibility of the post-negated text and the corresponding new labels, we conduct a second verification round (**Round 2**). In this step, three GPT-4 omni models with different temperatures (0, 0.7, 1.2) independently evaluate the outputs from Round 1. The evaluation prompt, shown in Table 2, is designed to control and assess data quality. Each example is evaluated on two criteria: **Logical Negation**, whether the post-negated text logically and effectively transforms the meaning of the pre-negated text, and **New Label Validity**, whether the revised text is correctly assigned the appropriate hallucination label. For each criterion, all three evaluators provide a binary judgment (`Pass` or `Fail`), yielding an outcome such as [Pass, Pass] if both aspects are satisfied. Only examples that unanimously receive "Pass" for both criteria are retained in the NegHalu dataset. This stringent process, described in more detail with qualitative examples in Appendix F, demonstrates the robustness and reliability of our logical transformation and label assignment. As a result, the final NegHalu dataset consists exclusively of high-quality, validated negated examples.

Qualitative Analysis As shown in Table 13, minimal negation operations in HaluEval effectively flip factual status: inserting "not" into a true claim about FC Bayern Munich introduces a hallucination, while adding "did not" to an erroneous claim about Michael Sheen corrects it. Similarly, in Table 14, Bamboo examples show that accurate hypotheses (e.g., the competitiveness of translated data models) become false once negated, while hallucinated statements about summary generation methods are corrected through the same operation.

In Table 15, SelfCheckGPT-WikiBio further demonstrates this precision, where a true claim

Restructuring Task Prompt | Hallucination Detection Datasets (Round 1)

##System:
You are a human annotator and an English native speaker, restructuring text according to given instructions.

##Instruction:
- The provided text is structured as [A], [B], [C], and [LABEL].
- [C] includes content that allows for determining the presence of hallucination based on [A] and [B].
- [LABEL] indicates whether [C] is "Hallucinated" if it contains hallucinations, or "True" if it does not.

Your task is to restructure [C] into a negative statement [NEW C] by adding "not" ONCE so that [Label] changes accordingly.

##Requirements:
- Do not use double negatives.
- Adding "not" only once is mandatory.
- The final result MUST align with real-world facts and commonsense.
- Generate only the text within [NEW C], omitting any other content.

##Input Format:
```
[A]: {GIVEN A TEXT}
[B]: {GIVEN B TEXT}
[C]: {GIVEN C TEXT}
[LABEL]: {HALLUCINATED OR TRUE}
[NEW C]:
[NEW LABEL]:
```

Table 1: Generalized prompt used for hallucination detection datasets to restructure new negated texts and labels.

Evaluation Task Prompt | Data Verification (Round 2)

##System:
You are a meticulous evaluator, carefully assessing if generated responses meet specific instructions and requirements.

##Evaluation Instructions:
- The text provided is structured as [A], [B], [C], [NEW C], [LABEL], and [NEW LABEL].
- Your task is to evaluate the [NEW C] and [NEW LABEL] for the following criteria:

1. **Logical Negation**: Ensure that [NEW C] negates [C] logically to change the meaning and the [LABEL] appropriately.
2. **New Label Validity**: Check that [NEW C] is appropriate for the assigned [NEW LABEL].

##Output Format for Evaluation:
After evaluating each criterion, rate it as "Pass" or "Fail." If a criterion fails, provide a brief reason. The final output should use the following format:

##Evaluation Criteria:
- Logical Negation: Pass / Fail
- NEW Label Validity: Pass / Fail

##Output Format:
```
[RESULT]: [Pass, Pass]
```

##Input Format:
```
[RESULT]:
```

Table 2: Generalized prompt for evaluating logical negations and the validity of new negated text and labels.

about Lee Hsien Loong is rendered false by negation, while a false claim about Admiral Aylmer is corrected. These cases across dialogue, QA, and summarization confirm that our negation strategy reliably induces or removes hallucinations with minimal intervention, ensuring robustness and low noise across datasets.

Human Evaluation In addition to automated verification, the authors manually inspected all 1,950 generated examples to ensure the overall plausibility and factual consistency of the NegHalu dataset. Human evaluation served as a precautionary step to prevent the inclusion of nonsensical or factually incorrect content and to ensure alignment with real-world knowledge and commonsense. For example, if a question asks, "*Which industry do Richard Hawley and Chicago's Catherine belong to?*" and the original hallucinated answer is "*Richard Hawley is a chef*," then the negated output "*Richard Hawley is not a chef*" would contradict known facts about Richard Hawley. As a result of this manual review, we identified and revised 13 nonsensical or factually incorrect examples in HaluEval, 11 in SelfCheckGPT-WikiBio, and 3 in BamBoo. Additionally, for BamBoo–SenHallu, 2 cases involving double negation were rephrased as single negation to maintain contextual clarity.

Dataset	Post Negation (Pre/Post)	After Verification (Pre/Post)
NegHalu	6,257 / 6,257	1,950 / 1,950
⊢HaluEval - QA	1,500 / 1,500	400 / 400
⊢HaluEval - Dialogue	1,500 / 1,500	400 / 400
⊢HaluEval - Sum	1,500 / 1,500	400 / 400
⊢BamBoo - AbsHallu	200 / 200	152 / 152
⊢BamBoo - SenHallu	200 / 200	136 / 136
⊢SelfCheckGPT-WikiBio	1,357 / 1,357	462 / 462

Table 3: Dataset Overview for NegHalu and Its Subsets. SelfCheckGPT-WikiBio represents the number of sentences obtained by splitting 238 paragraphs.

Data Statistics and Label Balancing Table 3 presents the statistics for NegHalu, summarizing the results after the two-round process of generation and verification. The NegHalu dataset consists of 1,950 samples, reconstructed from the three original hallucination detection datasets with post-negated text and corresponding new labels. Each subset within NegHalu preserves the evaluation framework and methodology of its respective source benchmark. To mitigate label imbalances, we maintain an equal ratio of hallucinated to faithful samples in HaluEval and SelfCheckGPT-WikiBio, while BamBoo is excluded from this ad-

justment because of its comparatively smaller size. Additionally, during the data verification process, we include only post-negated examples where the meaning of the original text has been altered, resulting in a label change. This ensures that the ratio of labels between pre- and post-negated examples remains equal in the final dataset. This balanced distribution makes it easier to analyze the model's preference for specific labels when presented with pre- or post-negated examples.

4 Experimental Setup

Models To achieve diverse and representative coverage in our experiments, we focus on selecting LLMs that are well-regarded in the open-source community and frequently serve as benchmarks in follow-up research. Our experimental framework is designed to investigate whether these models exhibit consistent patterns or distinctive behaviors. The models chosen for this study—Llama2 (Touvron et al., 2023) (*meta-llama/Llama-2-7b-chat-hf*), Llama3 (AI@Meta, 2024) (*meta-llama/Meta-Llama-3-8B-Instruct*), Mistral (Jiang et al., 2023) (*mistralai/Mistral-7B-Instruct-v0.3*), and Qwen3 (Bai et al., 2023) (*Qwen/Qwen3-4B*)—are known for their ability to generate high-quality outputs from provided instructions and for their adherence to the exact match metric, making them well-suited for our evaluation tasks.

Datasets and Evaluation We adopt standard hallucination detection benchmarks: HaluEval, BamBoo, and SelfCheckGPT-WikiBio. Evaluations use pairs of pre-negated and post-negated examples, with original and updated labels, under consistent settings. **HaluEval** and **BamBoo** use binary classification to assess whether model outputs contain hallucinations, reporting Accuracy (HaluEval) and F1 score (BamBoo). **SelfCheckGPT-WikiBio** measures sentence-level accuracy against human annotations, treating any inaccuracy as hallucination. For all datasets, we compare model performance before and after negation to analyze label shifts.

Lens Observation This method examines intermediate representations and attention distributions within the model to uncover how negated text influences the generation process. We employ a lens observation method to analyze LLMs' processing of pre- and post-negated text across layers. The Logit Lens (nostalgebraist, 2020; Dar et al., 2023) projects internal model states onto the vocabulary space, tracking prediction changes layer by layer. Lens observation helps pinpoint token-wise changes across layers, offering a comprehensive understanding of how negation affects predictions. We focus on observing the final token of the input and the first token generated by the model to measure the judgment in response to negated input.

In-Context Learning We assess model adaptability to negation by providing pre- and post-negated examples together in the input context (Brown et al., 2020; Dong et al., 2024a). Our experiments include zero-shot, two-shot, and four-shot settings, each offering different combinations of faithful and hallucinated examples as shown in Table 12. The two-shot setting serves as our default.

Chain-of-Thought We use step-by-step reasoning to evaluate whether it reduces hallucinations with negated text (Wei et al., 2022). Models are prompted to explain their reasoning before making hallucination judgments. Table 12 shows the CoT prompt template.

Knowledge Editing In our experiments, we adopt the approach of AlphaEdit (Fang et al., 2024) to examine and address hallucination issues in negated text scenarios. To prevent disruptions to parametric knowledge, AlphaEdit projects parameter updates onto the null space of the preserved knowledge, ensuring minimal interference with existing factual associations. Our study focuses on using the null-space constraint to preserve affirmative knowledge while updating negated knowledge. We approximate the covariance matrix of preserved knowledge and use causal tracing to identify layers for editing. By leveraging 100,000 (subject, relation, object) triplets from Wikipedia (Meng et al., 2023), we construct a basis for preserved knowledge and target layers critical for encoding it, ensuring edits address negated knowledge without unwanted parameter updates. Training details are provided in Appendix B.

5 Results

In this section, we present experimental findings and analyses that provide answers to the research questions raised in this study. A detailed summary of the answers to the research questions is provided in the Appendix A.

A1. LLMs Exhibit Degradation and Bias in Hallucination Detection for Negated Text

Models	HaluEval-QA (Acc)		HaluEval-Dialogue (Acc)		HaluEval-Sum (Acc)		BamBoo-AbsHallu (P/R/F1)		BamBoo-SenHallu (P/R/F1)	
	Pre	Post	Pre	Post	Pre	Post	Pre	Post	Pre	Post
Llama-2-7B	0.4825	**0.4950**	**0.6150**	0.4975	**0.4750**	0.4625	**59.8/73.6/66.0**	38.4/54.1/44.9	**68.7/77.3/72.7**	34.3/47.9/40.0
Llama-3-8B	**0.7650**	0.5525	**0.7825**	0.4500	**0.6600**	0.5175	**59.5/96.7/72.7**	38.9/80.3/52.4	**69.9/97.7/81.5**	31.1/58.3/40.6
Mistral-7B-v0.3	**0.5900**	0.5200	**0.7050**	0.5100	**0.5950**	0.5125	**61.1/100/75.8**	32.6/49.2/39.2	**77.3/96.6/75.9**	31.1/47.9/37.7
Qwen3-4B	**0.4425**	0.2775	**0.7425**	0.5150	**0.5625**	0.4675	**64.5/100/78.4**	50.0/21.3/29.9	**82.8/93.2/87.7**	46.7/14.6/22.2

Table 4: Performance comparison of models across HaluEval and Bamboo subsets in the NegHalu. **Acc** represents accuracy, and **P/R/F1** denotes precision, recall, and F1-score. Bold text indicates the higher performance between **Pre**- and **Post**-negated scenarios for the input example.

Correct Answers	HaluEval (Pre-negated)		HaluEval (Post-negated)		BamBoo (Pre-negated)		BamBoo (Post-negated)	
	Halu. = "YES"	Halu. = "NO"	Halu. = "YES"	Halu. = "NO"	Halu. = "YES"	Halu. = "NO"	Halu. = "YES"	Halu. = "NO"
Llama-2-7B	408	289	504 △	78 ▽	33	135	82 △	56 ▽
Llama-3-8B	398	443	422 △	186 ▽	12	174	40 △	77 ▽
Mistral-7B-v0.3	413	427	578 △	39 ▽	26	176	66 △	53 ▽
Qwen3-4B	410	191	467 △	37 ▽	42	173	158 △	20 ▽

Table 5: Label distribution across pre- and post-negated scenarios in the HaluEval and BamBoo subsets of the NegHalu. **Halu.** represents the number of examples classified as "Hallucinated = YES" or "Hallucinated = NO" among correctly predicted answers for each model. △ and ▽ indicate increases and decreases, respectively, in Post-negated compared to Pre-negated.

Models	SelfCheckGPT-WikiBio					
	Pre (Acc)	Post (Acc)	Pre (Avg)	Post (Avg)	Pre (Dis)	Post (Dis)
Llama-2-7B	**0.6169**	0.4686	0.5357	0.9205	0.3810	0.5319
Llama-3-8B	**0.6786**	0.4610	0.3598	0.8853	0.3209	0.5390
Mistral-7B-v0.3	**0.6450**	0.4881	0.2489	0.6245	0.3550	0.5119
Qwen3-4B	**0.5866**	0.5022	0.8939	0.9892	0.4134	0.4978

Table 6: Performance comparison of models on the SelfCheckGPT-WikiBio subset in the NegHalu. **Acc** represents accuracy, **Avg** denotes the average of cumulative scores for model responses (0: True, 1: Hallucinated), and **Dis** indicates the absolute distance between the assigned labels and responses. Bold and Underline highlight the higher scores based on each column.

NegHalu - HaluEval Table 4 presents the performance of models on the QA, Dialogue, and Summarization tasks in the HaluEval dataset under pre- and post-negated input scenarios. Across all tasks, model performance generally decreases when detecting hallucinations with post-negated inputs. Interestingly, Llama2 demonstrates robustness to negated inputs, maintaining performance levels comparable to non-negated scenarios, while Llama3 and Mistral experience significant performance drops. Qwen3-4B shows mixed behavior, performing well on Dialogue and Summarization in the pre-negated setting but dropping sharply in QA and post-negated cases. As shown in Table 5, models exhibit a strong tendency to classify negated texts as hallucinated, with an 21.0% increase in hallucination predictions and a 74.8% decrease in faithfulness judgments. This suggests that models may develop biases toward specific labels when processing negated inputs.

NegHalu - BamBoo Table 4 compares the performance of models in the BamBoo dataset for AbsHallu and SenHallu tasks under pre- and post-negated input scenarios. Across all models and metrics, the post-negated scenario consistently results in significant performance declines, with decreases as large as 51.8. Furthermore, Table 5 highlights greater label distribution shifts in BamBoo compared to HaluEval. For post-negated inputs, hallucination predictions increase by approximately 206.2%, while faithfulness judgments decrease by around 68.7%. These findings align with the trends observed in HaluEval, further confirming that LLMs face considerable challenges in detecting hallucinations when processing negated text.

NegHalu - SelfCheckGPT Table 6 shows the performance of models in classifying hallucinations on a sentence-level basis within SelfCheckGPT-WikiBio under pre- and post-negated input scenarios. Consistent with earlier datasets, model performance consistently decreases with negated inputs. Additionally, the proportion of hallucination judgments increases across all models, accompanied by growing gaps between predictions and ground-truth labels. These results suggest that negated text inputs introduce new and unintended hallucination patterns, reflecting similar trends across all evaluated datasets.

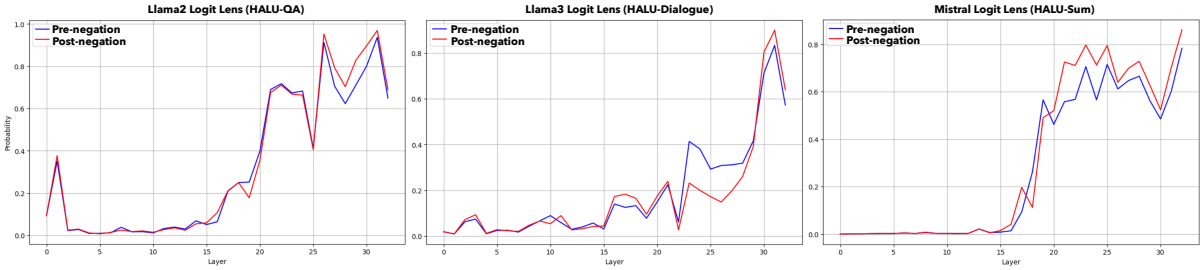

Figure 2: Logit lens results showing probability shifts for pre- and post-negated examples in the HaluEval subsets of NegHalu. Each curve tracks the probability of the first output token across the layers of the model, comparing pre-negated and post-negated inputs. The blue curves represent scenarios with pre-negated inputs, while the red curves indicate scenarios with post-negated inputs.

Models	HaluEval (Pre/Post) - Accuracy			BamBoo (Pre/Post) - F1			SelfCheckGPT (Pre/Post) - Accuracy		
	0-shot	2-shot	4-shot	0-shot	2-shot	4-shot	0-shot	2-shot	4-shot
Llama-2-7B	0.4466 / 0.4500	0.5241 / 0.4850	0.4975 / 0.4325	73.2 / 47.3	69.4 / 42.5	63.6 / 43.4	0.5519 / 0.4838	0.6169 / 0.4686	0.5758 / 0.4805
Llama-3-8B	0.0233 / 0.0025	0.7358 / 0.5066	0.7333 / 0.5083	77.4 / 43.9	77.1 / 46.5	78.1 / 48.1	0.6851 / 0.4729	0.6786 / 0.4610	0.6742 / 0.4729
Mistral-7B-v0.3	0.5600 / 0.5100	0.6300 / 0.5142	0.5167 / 0.5058	80.9 / 36.8	75.9 / 38.5	81.1 / 36.2	0.6515 / 0.4686	0.6450 / 0.4881	0.5703 / 0.4946
Qwen3-4B	0.5317 / 0.3867	0.5825 / 0.4200	0.5817 / 0.4750	82.8 / 28.2	83.1 / 26.1	83.6 / 39.8	0.5779 / 0.4232	0.5866 / 0.5022	0.5325 / 0.4989

Table 7: Performance comparison of models across HaluEval, BamBoo, and SelfCheckGPT subsets in the NegHalu dataset. **Accuracy** and **F1** scores are reported for pre- and post-negated scenarios across 0-shot, 2-shot, and 4-shot.

A2. Lack of Distinction Between Pre- and Post-Negated Text

Logit Lens Observation When negated text is provided as input, hallucination detection performance decreases, and models show a bias toward classifying the input as containing hallucinations. To understand the underlying cause, we trace the internal states of LLMs during the hallucination detection. We observe the final token of inputs containing pre- and post-negated text. Figure 2 illustrates the probability shifts for the next token by analyzing the hidden states at each layer of Llama2, Llama3, and Mistral on NegHalu. While the specific layers and magnitude of these shifts vary across models, strong probability fluctuations generally occur in the middle layers, followed by another significant fluctuation near the final layers.

Subtle Differences LLMs in Figure 2 show only marginal differences between pre- and post-negated examples. This implies that despite the transformation of context or knowledge induced by negation, the models fail to clearly recognize these changes when determining hallucination. This phenomenon appears to be associated with the treatment of negation within the model's latent representation, where negation functions more as a single token rather than as a logical operator (Bhar and Asher, 2024). Moreover, post-negated examples generally exhibit greater confidence in their decisions or show significant fluctuations near the final layers. This behavior closely resembles the token probability shifts observed in hallucination-inducing cases, as reported in (Chen et al., 2024; Jiang et al., 2024), and is considered a contributing factor to the bias toward hallucination judgments for negated examples.

Negation Amplifies Hallucination Bias To further analyze the above results, we examine the probability shifts across each layer for cases where hallucination judgments were correct versus incorrect for pre- and post-negated examples. As shown in Figure 4, Llama3 and Mistral present substantial differences in probability shifts between successful and unsuccessful hallucination detection for pre-negated examples. However, for post-negated examples, these differences are relatively minor. The models exhibit stronger confidence in incorrect predictions compared to correct ones. These results demonstrate that when negated text is provided as input, models experience confusion in hallucination detection, highlighting the risk of falling into new hallucination patterns induced by negation.

A3. Limited Improvements but Reveal Underlying Challenges

In-Context Learning Shows Inconsistent Gains Across Models and Tasks Table 7 compares model performance across 0-, 2-, and 4-shot settings, where the number of examples provided

Models	HaluEval (Pre/Post) - Acc		BamBoo (Pre/Post) - F1		SelfCheckGPT (Pre/Post) - Acc	
	2-shot + CoT	4-shot + CoT	2-shot + CoT	4-shot + CoT	2-shot + CoT	4-shot + CoT
Llama-2-7B	0.5308 / 0.4850	0.5808 / 0.4617	48.3 / 40.0	40.0 / 33.5	0.5108 / 0.5000	0.5043 / 0.5000
Llama-3-8B	0.7388 / 0.5067	0.6992 / 0.5108	69.0 / 44.3	72.6 / 42.2	0.5130 / 0.5000	0.5020 / 0.5000
Mistral-7B-v0.3	0.5400 / 0.5147	0.7000 / 0.5192	80.2 / 38.6	80.3 / 37.8	0.6082 / 0.4805	0.5610 / 0.5065
Qwen3-4B	0.5767 / 0.5333	0.5667 / 0.5350	82.1 / 26.7	81.8 / 39.2	0.6494 / 0.5000	0.5974 / 0.5011

Table 8: Performance comparison of models across HaluEval, BamBoo, and SelfCheckGPT subsets in the NegHalu dataset. **Accuracy (Acc)** and **F1** scores are reported for pre- and post-negated scenarios across CoT conditions.

in the input prompt varies. On average, the 2-shot setting yields the highest performance across comparable cases. However, the influence of in-context examples on performance varies depending on the dataset and model, indicating that an increase in the number of examples does not necessarily guarantee improved performance. Interestingly, Llama3 demonstrates significant difficulty in following instructions to generate responses for hallucination detection in the zero-shot setting, resulting in notably low scores. This outcome shows that, as the number of shots increases, instruction-following abilities—beyond hallucination detection—also play a role in determining performance. Mistral exhibits relatively stable improvements across shots, though post-negated cases still reduce performance. Qwen3-4B shows stable instruction-following across shot settings, with strong pre-negated performance on BamBoo and SelfCheckGPT. However, it exhibits sharp declines under post-negated conditions, indicating particular sensitivity to negation despite otherwise consistent few-shot gains. These results imply that In-Context Learning can immediately enhance hallucination detection performance for certain models, datasets, and tasks, particularly when balanced examples with pre-/post-negated text and hallucination/no-hallucination cases are included. However, even in the 4-shot with balanced examples, performance declines were observed for hallucination detection with negated text. This raises questions about whether models fundamentally understand negated text and whether they can mitigate newly induced hallucinations.

CoT Reasoning Fails to Provide Consistent Improvements To address LLMs' low hallucination detection capabilities when processing negated text, we applied CoT reasoning with in-context learning examples to enhance performance. Table 8 presents the experimental results of applying CoT reasoning steps to 2-shot and 4-shot in-context learning settings. As described in §D, the

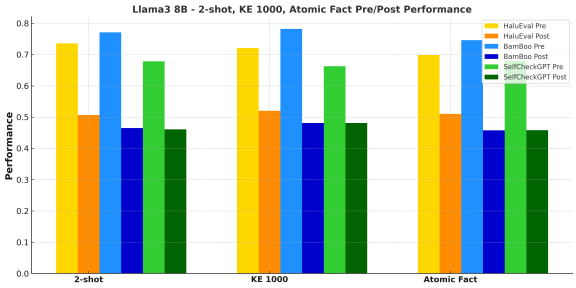

Figure 3: Performance comparison of Llama3 across NegHalu subsets under negated knowledge updates using AlphaEdit and two different target corpus.

models are instructed to articulate the reasoning behind their judgment during hallucination detection. The results demonstrate that the effectiveness of CoT reasoning varies significantly depending on the dataset, task, number of context examples, and the model used. Mistral and Llama2 show substantial performance improvement on pre-negated examples in HaluEval when using CoT prompts. However, consistent performance gains are not observed, with post-negated examples often showing minimal improvement or even performance degradation. These findings align with Li et al. (2023), who argue that CoT reasoning steps alone are insufficient as a fundamental solution for improving hallucination detection.

Knowledge Editing Modifies Model Behavior but Does Not Resolve the Negation Problem Based on the earlier empirical results, where negated text is misclassified as containing hallucinations, we apply knowledge editing to mitigate newly induced hallucinations or biases. To address this, we use negated knowledge—transformed from positively framed knowledge—to ensure consistency with real-world facts or commonsense. We employ AlphaEdit, a null-space-constrained knowledge editing, to update negated knowledge while minimizing damage to the model's existing knowledge. Our experiments use two different editing target corpora: 1,000 factual statements from ROME's

dataset (Meng et al., 2022) (KE 1000) and atomic facts parsed from each dataset's given knowledge, transformed into negated knowledge (Atomic Fact). Figure 3 compares the performance of models in a 2-shot setting without knowledge editing versus with knowledge updated using AlphaEdit applied to KE 1000 or Atomic Fact corpus. The results show that knowledge editing with KE 1000 effectively minimizes damage to pre-negated knowledge while slightly improving performance on post-negated examples. However, this approach does not fundamentally resolve issues caused by negated text, as performance degradation on pre-negated examples appears in SelfCheckGPT, showing limitations in handling negated knowledge effectively.

NegHalu+	HaluEval (Pre/Post) - Accuracy		
	QA	Dialogue	Summarization
Llama-2-7B	0.4825 / 0.4125	0.6150 / 0.5650	0.4750 / 0.5250
Llama-3-8B	0.7650 / 0.4525	0.7825 / 0.5150	0.6600 / 0.5675
Mistral-7B-v0.3	0.5900 / 0.5075	0.7050 / 0.5575	0.5950 / 0.5025
Qwen3-4B	0.4425 / 0.2775	0.7425 / 0.5150	0.5625 / 0.4675

Table 9: Performance of models on HaluEval where originally explicit negation examples were partially replaced (5% each) with implicit and morphological negation.

6 Effect of Adding Implicit and Morphological Negation

Table 9 presents the results after extending the originally explicit-only HaluEval with a small proportion of implicit and morphological negation. Implicit forms include expressions such as *doubt*, *hardly*, *fails to*, *unlikely that*, and *questionable whether*. Morphological forms use affixes such as *un*, *in*, *im*, and *dis*, as well as the suffix *less*, to produce words like *incorrect*, *impossible*, and *useless*.

The overall pattern remains clear. Post-negated inputs reduce accuracy across models and tasks, but the scale and distribution of the decline shift once diverse negation types are introduced. Llama2 shows a smaller gap between pre- and post-negated inputs, with summarization even improving from 47.5% to 52.5%, which suggests partial robustness when negation cues vary. Llama3 continues to display the strongest sensitivity, while Mistral shows slightly milder degradation on dialogue. Qwen3-4B still suffers sharp declines in QA and dialogue, revealing a consistent vulnerability to negation regardless of its form.

NegHalu+ maintains the overall effect of negation but reveals distinct failure profiles compared with the explicit-only setting. Some models such as Llama2 and Mistral show slightly reduced performance gaps, while others such as Llama3 and Qwen3-4B exhibit sharper declines, especially on QA and dialogue. These results indicate that robustness measured only with explicit negation underestimates true vulnerability, and that a mixture of negation types exposes model-specific weaknesses across tasks.

7 Conclusion

In this study, we explore the impact of negated text on hallucination detection in LLMs by constructing NegHalu, a dataset designed to evaluate model performance under pre- and post-negation scenarios. Through systematic experiments, we examine three key research questions and uncover fundamental limitations in how LLMs process negation. Our answers highlight that LLMs exhibit performance degradation, systematic biases, internal behavioral constraints, and limited performance improvements in handling hallucinations. Furthermore, we identify the risk of new types of hallucinations emerging due to negation, posing additional challenges for model reliability. For future work, we emphasize the need for deeper architectural refinements and advanced strategies to improve LLMs' ability to process negation effectively, ensuring robust and reliable performance across diverse contexts.

Limitations

This study has several limitations that warrant consideration. First, we use only a subset of the HaluEval dataset, which may lead to results for pre-negated text differing slightly from those obtained using the full dataset. Additionally, the BamBoo dataset contains an insufficient number of samples to achieve balanced labels, resulting in experiments being conducted with slight label imbalance. Second, due to computational resource constraints, we were unable to compare larger models. Even if such experiments were conducted, significant differences in hyperparameter settings would be required, which could lead to outcomes different from those reported here. Third, during the creation and verification of NegHalu, there may be a small number of errors in labeling hallucinations that differ from human interpretations. However, these are not considered substantial enough to overturn the overall experimental findings. Fourth, while our verification step uses multiple GPT-4 models, we

acknowledge that employing the same model as both generator and judge could introduce bias in favor of its own generations. To mitigate this risk, we ensembled models with different temperature settings and required full agreement among them. In addition, the authors manually reviewed and adjusted the outputs where necessary. Fifth, while we explored various methods to address the hallucinations and biases introduced by the negated text, we were unable to propose a complete solution. We consider solutions targeting only the impact of negated text on hallucination problems to be impractical. Instead of relying on external modules or extensive tuning, we apply intrinsic knowledge and minimal knowledge editing. We hope that the analyses presented in this paper will provide a solid foundation for future research. Lastly, we excluded ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023) from our experiments, as their application resulted in significant performance degradation on pre-negated knowledge, unlike AlphaEdit.

Acknowledgments

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (RS-2024-00398115, Research on the reliability and coherence of outcomes produced by Generative AI). This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2021R1A6A1A03045425). This work was supported by the Commercialization Promotion Agency for R&D Outcomes(COMPA) grant funded by the Korea government(Ministry of Science and ICT)(2710086166).

References

AI@Meta. 2024. Llama 3 model card.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. 2023. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4895–4901.

Hiba Arnaout, Simon Razniewski, Gerhard Weikum, and Jeff Z Pan. 2022. Uncommonsense: Informative negative knowledge about everyday concepts. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 37–46.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Swarnadeep Bhar and Nicholas Asher. 2024. Strong hallucinations from negation and how to fix them. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 12670–12687.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: LLMs' internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations*.

Jiangjie Chen, Wei Shi, Ziquan Fu, Sijie Cheng, Lei Li, and Yanghua Xiao. 2023. Say what you mean! large language models speak too positively about negative commonsense knowledge. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9890–9908.

Prafulla Kumar Choubey, Alex Fabbri, Jesse Vig, Chien-Sheng Wu, Wenhao Liu, and Nazneen Rajani. 2023. Cape: Contrastive parameter ensembling for reducing hallucination in abstractive summarization. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10755–10773.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R Glass, and Pengcheng He. 2023. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*.

David Dale, Elena Voita, Loïc Barrault, and Marta R Costa-jussà. 2023. Detecting and mitigating hallucinations in machine translation: Model internal workings alone do well, sentence similarity even better. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 36–50.

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. 2023. Analyzing transformers in embedding space. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16124–16170.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, et al. 2024a. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128.

Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2024b. Bamboo: A comprehensive benchmark for evaluating long text modeling capacities of large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2086–2099.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Xiang Wang, Xiangnan He, and Tat-seng Chua. 2024. Alphaedit: Null-space constrained knowledge editing for language models. *arXiv preprint arXiv:2410.02355*.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Reto Gubelmann and Siegfried Handschuh. 2022. Context matters: A pragmatic study of plms' negation understanding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4602–4621.

Nuno M. Guerreiro, Duarte M. Alves, Jonas Waldendorf, Barry Haddow, Alexandra Birch, Pierre Colombo, and André F. T. Martins. 2023. Hallucinations in large multilingual translation models. *Transactions of the Association for Computational Linguistics*, 11:1500–1517.

Md Mosharaf Hossain, Dhivya Chinnappa, and Eduardo Blanco. 2022. An analysis of negation in natural language understanding corpora. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 716–723.

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

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. 2023b. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023a. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Ziwei Ji, Zihan Liu, Nayeon Lee, Tiezheng Yu, Bryan Wilie, Min Zeng, and Pascale Fung. 2023b. Rho: Reducing hallucination in open-domain dialogues with knowledge grounding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4504–4522.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Che Jiang, Biqing Qi, Xiangyu Hong, Dayuan Fu, Yang Cheng, Fandong Meng, Mo Yu, Bowen Zhou, and Jie Zhou. 2024. On large language models' hallucination with regard to known facts. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1041–1053.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818.

Aditya Khandelwal and Suraj Sawant. 2020. Negbert: A transfer learning approach for negation detection and scope resolution. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5739–5748.

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. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464.

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

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

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Marvin Minsky. 1997. Negative expertise. In *Expertise in context: human and machine*, pages 515–521.

Roser Morante, Sarah Schrauwen, and Walter Daelemans. 2011. Annotation of negation cues and their scope: Guidelines v1. *Computational linguistics and psycholinguistics technical report series, CTRS-003*, pages 1–42.

nostalgebraist. 2020. Interpreting gpt: the logit lens.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Jaehyung Seo, Jaewook Lee, Chanjun Park, SeongTae Hong, Seungjun Lee, and Heui-Seok Lim. 2024. Kocommongen v2: A benchmark for navigating korean commonsense reasoning challenges in large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 2390–2415.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Thinh Hung Truong, Timothy Baldwin, Karin Verspoor, and Trevor Cohn. 2023. Language models are not naysayers: an analysis of language models on negation benchmarks. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (* SEM 2023)*, pages 101–114.

Chantal van Son, Emiel Van Miltenburg, and Roser Morante. 2016. Building a dictionary of affixal negations. In *Proceedings of the Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics (ExProM)*, pages 49–56.

Norbert Vanek, Ana Matić Škorić, Sara Košutar, Štěpán Matějka, and Kate Stone. 2024. Mental simulation of the factual and the illusory in negation processing: evidence from anticipatory eye movements on a blank screen. *Scientific reports*, 14(1):2844.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Shiping Yang, Renliang Sun, and Xiaojun Wan. 2023. A new benchmark and reverse validation method for passage-level hallucination detection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3898–3908.

Mengyu Ye, Tatsuki Kuribayashi, Jun Suzuki, Goro Kobayashi, and Hiroaki Funayama. 2023. Assessing step-by-step reasoning against lexical negation: A case study on syllogism. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14753–14773.

Biao Zhang and Rico Sennrich. 2019. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Zheng Zhao, Shay B Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249.

A Summary of Research Questions and Answers

We summarize the key findings of our study by revisiting the research questions and their corresponding answers.

RQ1. *Can LLMs distinguish between hallucinations and faithful statements in negated text as effectively as in affirmative text?*

- **A1. LLMs exhibit performance degradation and bias in hallucination detection for negated text.** Our results show that LLMs struggle to maintain consistent hallucination detection performance in negated text, displaying a bias toward hallucination predictions in post-negated cases. Across all evaluated tasks—including QA, dialogue, and summarization—models incorrectly classify negated factual statements as hallucinations at significantly higher rates. This effect is observed regardless of model type, suggesting that negation disrupts LLMs' hallucination judgments in a systematic and task-agnostic manner.

RQ2. *Can the model internally recognize differences caused by negation when detecting hallucinations?*

- **A2. LLMs show a lack of distinction between pre- and post-negated text.** Through logit lens analysis, we observe that LLMs exhibit strong probability shifts in the middle and final layers when

processing negated inputs, yet these shifts do not result in meaningful internal differentiation between pre- and post-negated statements. Instead, negation appears to function as a lexical modifier rather than a logical transformation, leading to overgeneralized hallucination judgments. Additionally, models display increased confidence in incorrect predictions, reinforcing the idea that negation is not properly incorporated into the model's reasoning process. 13

RQ3. *Can targeted intervention strategies improve hallucination detection in the negated text?*

- **A3. Limited improvements but revealing underlying challenges.** We explore in-context learning, CoT reasoning, and knowledge editing as potential mitigation strategies. However, none of these methods provided a fundamental solution to negation-induced hallucination biases. In-context learning showed inconsistent performance gains, with improvements depending more on instruction-following ability rather than genuine negation comprehension. CoT reasoning improved pre-negated hallucination detection in certain cases but failed to generalize across datasets and did not consistently improve performance in post-negated cases. Knowledge editing slightly reduced hallucination errors in some conditions but failed to eliminate systematic negation biases, suggesting that negation errors are deeply embedded within the model's internal representations rather than merely arising from incorrect factual knowledge.

B Experimental Details

To evaluate the LLMs, we used a single NVIDIA A6000 GPU with 48GB memory capacity and AMD EPYC 7513 32-core Processor CPUs.

Model Details For the NegHalu experiments, we followed the hyperparameter settings defined by the existing benchmark datasets. Across all datasets, we used greedy decoding without sampling methods. The maximum output length was set to 4096 for HaluEval, 32 for BamBoo, and 5 for SelfCheckGPT-WikiBio.

Llama2 (Touvron et al., 2023) (*meta-llama/Llama-2-7b-chat-hf*) is a Transformer-based language model with 7 billion parameters. This model employs the SwiGLU activation function, Rotary Position Embedding (RoPE) (Su et al., 2024), and RMSNorm (Zhang and Sennrich, 2019) to enhance stability. Its configuration includes a maximum token length of 4096, 32 attention heads, 32 hidden layers, a vocabulary size of 32,000, and float16.

Llama3 (AI@Meta, 2024) (*meta-llama/Meta-Llama-3-8B-Instruct*) is a Transformer-based language model with 8 billion parameters, sharing the same fundamental structure as Llama2. It supports a maximum token length of 4096, 32 attention heads, 32 hidden layers, a vocabulary size of 128,256, and bfloat16.

Mistral (Jiang et al., 2023) (*mistralai/Mistral-7B-Instruct-v0.3*) is a Transformer-based language model with 7.3 billion parameters. It leverages Grouped-Query Attention (GQA) (Ainslie et al., 2023) and Sliding Window Attention (SWA) (Beltagy et al., 2020) mechanisms for computational efficiency. Its configuration includes a maximum token length of 4096, 32 attention heads, 32 hidden layers, a vocabulary size of 32,768, and bfloat16.

Qwen3 (Bai et al., 2023) (*Qwen/Qwen3-4B*) is a Transformer-based language model with 4 billion parameters. It adopts RoPE (Su et al., 2024), GQA (Ainslie et al., 2023), and RMSNorm (Zhang and Sennrich, 2019) to improve computational efficiency and stability. The model is configured with a maximum token length of 32,768 (extendable up to 131,072 with YaRN), 36 hidden layers, 32 attention heads with 8 key-value heads, and a vocabulary size of 151,936. We used the dense variant of Qwen3-4B with bfloat16 precision. In our NegHalu experiments, the model was run without enabling the *think* mode, using only direct response generation under greedy decoding.

Knowledge Editing Details To update negated knowledge in Llama3 using AlphaEdit (Fang et al., 2024), we target layers 4 through 8 based on causal tracing results derived from 100,000 triplets obtained from Wikipedia (Meng et al., 2023). The editing corpus includes two sources: **KE 1000**, which consists of 1,000 factual statements from ROME (Meng et al., 2022), and **Atomic Fact**, which represents up to 1,000 atomic facts parsed from the given knowledge in each dataset. To parse these facts and transform them into negated knowledge, we use the GPT-4 omni (`gpt-4o-2024-08-06`) (OpenAI, 2023) API. The prompts used for Atomic Fact parsing and negated knowledge construction are described in Tables 10 and 11

The key hyperparameters for AlphaEdit include a null-space threshold of 2e-2, which controls the preservation of existing knowledge during edits, and L2 regularization set to 10, which stabilizes

JSON Creation Prompt | Parsing Atomic Facts
##Instruction:
"Use the following JSON data as a guide to convert it to Atomic Facts."
"Do not omit or modify any existing key-values in the given JSON data."
"The output should be in the following format:"
"Generate up to 10 atomic facts."
{"Atomic_Fact_1": "First fact.", "Atomic_Fact_2": "Second fact.", ...}

Table 10: JSON creation prompt for parsing Atomic Facts. The prompt outlines the format and content requirements for generating up to 10 atomic facts based on the provided JSON data.

JSON Creation Prompt | Negated Atomic Facts
##Instruction:
"Refer to the following JSON data and transform the Atomic_Fact into a Negated Atomic Fact while keeping its meaning identical."
"Do not omit or modify any existing keys or values in the provided JSON data."
"The output must be in the following format:"
{"Atomic_Fact_1": "Input Data", "Negated_Atomic_Fact_1": "Negated FACT"}
"Match the original Atomic Fact and the Negated Atomic Fact and store them together."
"The Negated Atomic Fact must strictly retain the same meaning as the original Atomic Fact and should only contain negated text that does not conflict with factual knowledge of the real world."
##Examples:
{"Atomic_Fact_1": "The Secret Life of Bees belongs to Teen drama", "Negated_Atomic_Fact_1": "The Secret Life of Bees does not belong to any genre other than Teen drama"}
{"Atomic_Fact_2": "Teen drama includes A Walk to Remember as an example", "Negated_Atomic_Fact_2": "Teen drama does not exclude A Walk to Remember as an example"}

Table 11: JSON creation prompt for processing Atomic and Negated Atomic Facts. The prompt outlines the format and content requirements, providing examples of JSON objects that pair original Atomic Facts with their corresponding Negated Atomic Facts while preserving meaning and factual consistency.

the updates and prevents overfitting. The learning rate for vector updates is set at 1e-1 to ensure efficient optimization, while the clamp norm factor of 0.75 limits excessive parameter changes to maintain model stability. Finally, the updated batch size is set to 100, balancing computational efficiency and precision during the editing process.

C Dataset Details

- **HaluEval** (Li et al., 2023) comprises benchmarks for tasks such as question answering, knowledge-grounded dialogue, and summarization. From this dataset, we employ 1,500 examples for each task: **Halu-QA** (question answering), **Halu-Dialogue** (knowledge-grounded dialogue), and **Halu-Sum** (summarization). These examples are chosen to enable the model to detect hallucinations by assessing generated outputs in comparison to the provided context or factual knowledge.

- **BamBoo** (Dong et al., 2024b), we focus on two tasks: (1) **AbsHallu**, which involves determining whether summarizations contain hallucinated contents, with 200 examples selected for evaluation, and (2) **SenHallu**, a sentence-level task that evaluates the factual correctness of individual sentences, with 200 examples.

- **SelfCheckGPT-WikiBio** (Manakul et al., 2023) contains GPT-3-generated biographies for 238 individuals, with each sentence labeled to indicate whether it is hallucinated. This dataset offers fine-grained annotations, enabling the evaluation of hallucination detection in structured and narrative texts.

D Chain-of-Thought Prompt Template

Table 12 presents the generalized Chain-of-Thought (CoT) reasoning prompt used for hallucination detection in the NegHalu dataset. This template structures the input for LLMs, incorporating pre- and post-negated text alongside step-by-step reasoning to enhance model interpretability.

The prompt follows a systematic format, where the model is assigned the role of a hallucination detector, given task instructions, and provided with contextual knowledge to evaluate hallucination likelihood. Each example consists of: (i) Knowledge & Context: The factual grounding for evaluating the given statement. (ii) Negated Text: Either a pre-negated or post-negated version of the statement. (iii) Reasoning Step: A step-by-step explanation of why the text is or isn't a hallucination (included only for CoT). (iv) Hallucination Label: The ground-truth classification as hallucinated or factual.

The 4-shot setting includes balanced examples across pre-/post-negation and hallucination/no-hallucination cases, whereas the 2-shot setting only includes pre-/post-negated text, omitting the reasoning step when CoT is not applied. This structured prompt helps analyze whether CoT reasoning improves LLMs' ability to handle negation and detect hallucinations more accurately.

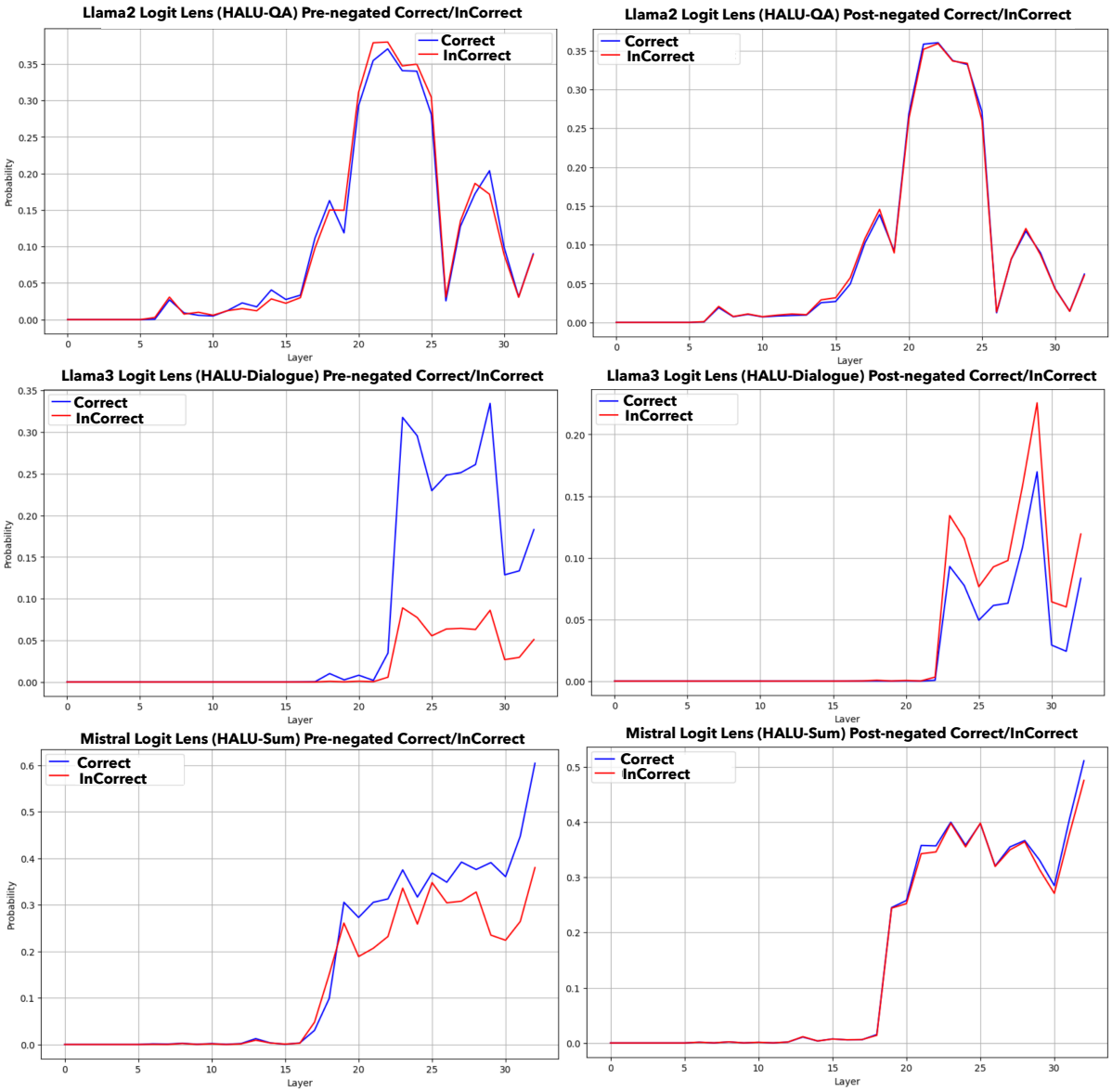

Figure 4: Logit lens results showing probability shifts for pre- and post-negated examples in the HaluEval subsets of NegHalu. The blue curves represent cases where the model generates the correct answer, while the red curves indicate cases where the model generates an incorrect answer.

E Lens Observation Results

Figure 4 illustrates the probability shifts for cases where the model generates correct and incorrect answers under pre- and post-negated inputs. We conducted lens observations using the HaluEval subsets of NegHalu to analyze these shifts. To ensure the diversity and generalizability of the experiments, we present results for different tasks across various models.

F Dataset Verification Details

Based on our Logical Negation and Label Validation criteria (as described in Table 16), our evaluation revealed several illustrative cases. For instance, in one Logical Negation failure, a pre-negated answer stating “American” for the question on James Henry Miller’s wife was modified to “American, not British.” Although a “not” was added, it incorrectly reversed the intended meaning. Additionally, a dialogue about recommending movies shifted from “Panic Room is a similar movie” to “Panic Room is not a similar movie,” which constitutes a logically inappropriate response. In contrast, a successful Logical Negation example is seen in the QA case for "The Messenger," where the pre-negated answer “The Messenger starred Michael Sheen” was correctly negated to “The Messenger did not

CoT Reasoning Prompt | NegHalu

##System:
System Role: {HALLUCINATION DETECTOR}
Instruction: {TASK INSTRUCTION}

##Example 1:
Knowledge: {GIVEN KNOWLEDGE}
Context: {GIVEN CONTEXT}
Pre-Negated Text: {DETECTION TARGET}
Reasoning Step: {REASON WHY}
Hallucination: NO

##Example 2:
Knowledge: {GIVEN KNOWLEDGE}
Context: {GIVEN CONTEXT}
Post-Negated Text: {DETECTION TARGET}
Reasoning Step: {REASON WHY}
Hallucination: YES

##Example 3:
Knowledge: {GIVEN KNOWLEDGE}
Context: {GIVEN CONTEXT}
Pre-Negated Text: {DETECTION TARGET}
Reasoning Step: {REASON WHY}
Hallucination: YES

##Example 4:
Knowledge: {GIVEN KNOWLEDGE}
Context: {GIVEN CONTEXT}
Post-Negated Text: {DETECTION TARGET}
Reasoning Step: {REASON WHY}
Hallucination: NO

Table 12: Generalized CoT reasoning prompt used for each dataset and task in the NegHalu dataset. The 4-shot examples used for In-Context Learning and CoT include balanced examples with pre/post-negated and hallucination/no-hallucination cases. The 2-shot examples include only pre/post-negated cases, and the Reasoning Step is omitted when CoT is not applied.

star Michael Sheen," effectively inverting the claim to match the factual context.

Regarding New Label Validation, we encountered cases where the negation process led to misclassifications; for example, when addressing the common profession of Am Rong and Alexandre Rockwell, the initial negative phrasing resulted in a hallucinated statement that was only partially corrected in the post-negated version, and a dialogue about "Pulp Fiction" failed to update the label despite the negation of an erroneous claim about Fred Savage. Conversely, another dialogue about FC Bayern Munich successfully shifted from a factually accurate pre-negated response (indicating the team plays football) to a negated version that properly contradicts known facts, thereby updating the label from not hallucinated to hallucinated.

HaluEval – Dialogue (Negation: Not Hallucinated → Hallucinated)
Knowledge: FC Bayern Munich: Sport – Football **Dialogue:** [Human] What do you think about FC Bayern Munich? **Pre-Negated Response (Not Hallucinated):** *"Unfortunately, I don't know much about this other team than they play football and are based out of Germany."* **Post-Negated Response (Hallucinated):** *"Unfortunately, I don't know much about this other team than they **do not** play football and are based out of Germany."*
HaluEval – QA (Negation: Hallucinated → Not Hallucinated)
Knowledge: The Messenger is a 2015 British supernatural mystery horror film directed by David Blair, written by Andrew Kirk and starring Robert Sheehan and Lily Cole. Robert Michael Sheehan (Irish: "Roibeárd Mícheál Ó Siodhacháin"; born 7 January 1988) is an Irish actor. **Question:** What male actor starred in The Messenger? **Pre-Negated Answer (Hallucinated):** *"The Messenger starred Michael Sheen."* **Post-Negated Answer (Not Hallucinated):** *"The Messenger did not star Michael Sheen."*

Table 13: HaluEval qualitative examples illustrating how negation can flip the hallucination label. In the first case, negation introduces a contradiction (creating a hallucination). In the second case, negation corrects an inaccurate statement (resolving a hallucination).

BamBoo – AbsHallu (Negation: Not Hallucinated → Hallucinated)
Title: Not Enough Data to Pre-train Your Language Model? MT to the Rescue! **Content:** Since the emergence of the attention-based Transformer architecture (Vaswani et al., 2017)... Data and models are publicly available. **Pre-Negated Hypothesis (Not Hallucinated):** *"The evaluation carried out on 9 NLU tasks indicates that models trained exclusively on translated data offer competitive results."* **Post-Negated Hypothesis (Hallucinated):** *"The evaluation carried out on 9 NLU tasks indicates that models trained exclusively on translated data do not offer competitive results."*
BamBoo – SenHallu (Negation: Hallucinated → Not Hallucinated)
Title: Towards Argument-Aware Abstractive Summarization of Long Legal Opinions with Summary Reranking **Content:** Legal opinions contain implicit argument structure spreading ... remaining unannotated portion of the CanLII dataset. **Pre-Negated Hypothesis (Hallucinated):** *"Our approach involves using document structure information to generate multiple candidate summaries, then reranking these candidates based on alignment with the document's argument role."* **Post-Negated Hypothesis (Not Hallucinated):** *"Our approach does not involve using document structure information to generate multiple candidate summaries, then reranking these candidates based on alignment with the document's argument role."*

Table 14: BamBoo qualitative examples showing how negation can flip the hallucination label for both abstract-level and sentence-level factuality judgments.

SelfCheckGPT-WikiBio (Negation: Not Hallucinated → Hallucinated)
WikiBio Text: Lee Hsien Loong is the third and current Prime Minister of Singapore, and has been in office since 2004. He is the elder son of Singapore's first Prime Minister, Lee Kuan Yew ... quickly rising to the rank of Brigadier-General. **Pre-Negated Generated Text (Not Hallucinated):** *"He is the eldest son of Singapore's first Prime Minister, Lee Kuan Yew."* **Post-Negated Generated Text (Hallucinated):** *"He is not the eldest son of Singapore's first Prime Minister, Lee Kuan Yew."*
SelfCheckGPT-WikiBio (Negation: Hallucinated → Not Hallucinated)
WikiBio Text: Admiral of the Fleet Matthew Aylmer, 1st Baron Aylmer (ca. 1650 – 18 August 1720) was a Royal Navy officer. He was ... the appointment of the Townshend ministry, Aylmer was reappointed Commander-in-Chief on 5 November 1714. ... seamen. **Pre-Negated Generated Text (Hallucinated):** *"He was made a baron in 1782 and was appointed Commander-in-Chief of the British forces in North America in 1783."* **Post-Negated Generated Text (Not Hallucinated):** *"He was not made a baron in 1782 and was not appointed Commander-in-Chief of the British forces in North America in 1783."*

Table 15: SelfCheckGPT-WikiBio qualitative examples where negation either introduces or resolves hallucination in biographical text generation.

Logical Negation – Fail
Question: What nationality was James Henry Miller's wife? **Pre-Negated Answer:** American **Post-Negated Answer:** American, not British **Dialogue History:** [Human]: Could you recommend some movies similar to Fight Club? **Pre-Negated Response:** Sure, Panic Room is a similar movie. **Post-Negated Response:** Sure, Panic Room is not a similar movie.
Logical Negation – Pass
Question: What male actor starred in The Messenger? **Pre-Negated Answer:** The Messenger starred Michael Sheen. **Post-Negated Answer:** The Messenger did not star Michael Sheen.
New Label Validation – Fail
Question: What profession does Am Rong and Alexandre Rockwell have in common? **Pre-Negated Answer (Hallucinated):** Alexandre Rockwell and Am Rong are not in the same profession. **Post-Negated Answer (Not Hallucinated):** Alexandre Rockwell and Am Rong are not both filmmakers. **Dialogue History:** [Human]: I like Pulp Fiction. What do you think about it? [Assistant]: I love it. It was written by Roger Avary [Human]: I heard he also wrote The Rules of Attraction. Do you know who is in that movie? **Pre-Negated Response (Hallucinated):** Fred Savage is in it **Post-Negated Response (Hallucinated):** Fred Savage is not in it
New Label Validation – Pass
Dialogue History: [Human]: What do you think about FC Bayern Munich? **Pre-Negated Response (Not Hallucinated):** "Unfortunately, I don't know much about this other team than they play football and are based out of Germany." **Post-Negated Response (Hallucinated):** "Unfortunately, I don't know much about this other team than they do not play football and are based out of Germany."

Table 16: Examples of dataset verification for logical negation and new label validation in hallucination detection. 'Fail' shows incorrect application, while 'Pass' shows successful handling.