Counterfactual Debating with Preset Stances for Hallucination Elimination of LLMs

Anonymous ACL submission

Abstract

Large Language Models (LLMs) excel in various natural language processing tasks but struggle with hallucination issues. Existing solutions have considered utilizing LLMs' inherent reasoning abilities to alleviate hallucination, such as self-correction and diverse sampling methods. However, these methods often overtrust LLMs' initial answers due to inherent biases. The key to alleviating this issue lies in overriding LLMs' inherent biases for answer inspection. To this end, we propose a **C**ounter**F**actual **M**ulti-**A**gent **D**ebate (CFMAD) framework. CFMAD presets the stances of LLMs to override their inherent biases by compelling LLMs to generate justifications for a predetermined answer's correctness. The LLMs with different predetermined stances are engaged with a skeptical critic for counterfactual debate on the rationality of generated justifications. Finally, the debate process is evaluated by a third-party judge to determine the final answer. Extensive experiments on four datasets of three tasks demonstrate the superiority of CFMAD over existing methods.

1 Introduction

Large Language Models, especially closed-source ones such as GPT-4 (Achiam et al., 2023) and Gemini (Team et al., 2023), have demonstrated state-of-the-art performance across various natural language processing tasks (Bubeck et al., 2023; Zhao et al., 2023). However, LLMs still struggle with the hallucination problem, *i.e.,* occasionally generating unfaithful content (Zhang et al., 2023; Bang et al., 2023; Zheng et al., 2023). Due to the black-box nature of closed-source LLMs, it is difficult for users to directly intervene in or optimize their internal mechanisms to address the hallucination problems. Currently, extensive research is investigating how to use LLMs' inherent reasoning abilities to alleviate hallucinations without model intervention (Shinn et al., 2024; Liang et al., 2023).

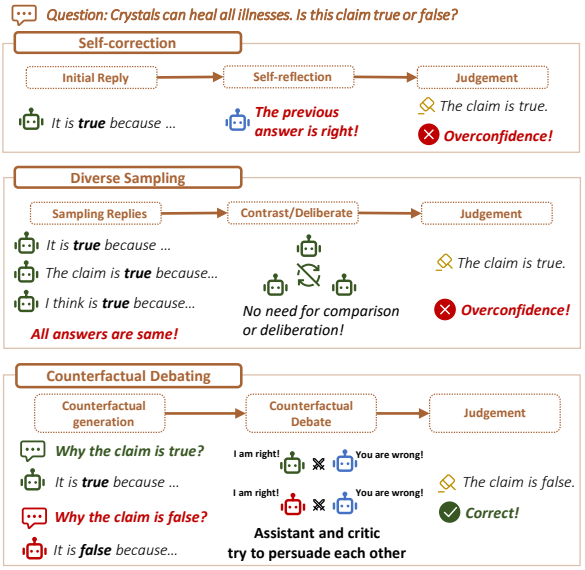

Figure 1: Comparison of CFMAD with self-correction and diverse sampling methods. CFMAD presets stances for LLMs to override their inherent biases.

Related work of using LLMs' own abilities for hallucination elimination can be categorized into self-correction and diverse sampling methods, which imitate human deep reasoning and broad reasoning to enhance LLMs' reasoning capabilities, respectively (Zhang et al., 2024b). Self-correction methods (Shinn et al., 2024; Madaan et al., 2024) guide LLMs to reflect on and refine their previous answers iteratively. Diverse sampling methods (Zhang et al., 2024b; Du et al., 2023; Wang et al., 2023; Mielke et al., 2022; Xiong et al., 2023) first sample multiple initial answers, and then compare or deliberate on the differences among these answers to reach a consistent answer.

While self-correction and diverse sampling methods show potential for improving the output reliability of LLMs, they still have the overconfidence issue (Mielke et al., 2022; Xiong et al., 2023) as illustrated in Figure 1. Self-correction methods may overtrust LLMs' initially generated an-

swers, making it difficult to effectively recognize errors (Huang et al., 2024b; Stechly et al., 2023; Valmeekam et al., 2023). By contrast, diverse sampling methods may repeatedly generate the same incorrect answers due to LLMs' inherent biases and beliefs (Wang et al., 2024b), limiting LLMs to contrast and deliberate on other possible answers. We believe that a key reason for the above overconfidence issue is that these methods do not intervene in the LLMs' answer-generation process, allowing LLMs to refine or sample diverse answers according to their own biases and beliefs.

The main challenge in addressing the overconfidence issue is to override LLMs' inherent biases and beliefs, compelling them to inspect answers they would not normally consider. To achieve this, we consider presetting different stances for LLMs, allowing LLMs to imagine each answer as correct in each round of reasoning, and then generate the reasons why the answer is valid. By overriding the LLM's original beliefs with this new mindset, we can regulate LLMs to assess the possibility of each answer being correct. Thereafter, we can eliminate the incorrect answers by reflecting the generated reasons for all answers.

To this end, we propose a **C**ounter**F**actual **M**ulti-**A**gent **D**ebate (CFMAD) framework comprising two key stages: abduction generation and counterfactual debate. In the abduction generation stage, LLMs are tasked with producing potential correct reasons for a predetermined answer. Subsequently, in the counterfactual debate stage, a structured debate method is employed to assess these abductions and ascertain the sole correct response. Specifically, we introduce a critic who questions the validity of each generated abduction, and prompt the LLM to defend its position in a debate with the critic. The deliberation is then presented to an impartial third-party judge for final adjudication. Extensive experiments spanning fact-checking, reading comprehension, and commonsense reasoning tasks validate the effectiveness of CFMAD over existing benchmarks across four datasets. We release our code and data at https://anonymous.4open.science/r/CFMAD-468D/.

The contributions of this work are threefold:

- We propose to preset various stances for LLMs, overriding their inherent biases and beliefs to address the overconfidence issue of LLMs.
- We propose a CFMAD framework, which instructs LLMs to generate abduction with preset stances and then conduct counterfactual debate to eliminate incorrect answers.
- We conduct extensive experiments on three generative tasks with four datasets, validating the effectiveness of CFMAD.

2 Preliminary Experiments

We formulate methods for self-correction and diverse sampling, and subsequently conduct a quantitative experiment to expose the overconfidence issue prevalent in both approaches.

2.1 Problem Definition

Self-correction. Self-correction methods involve two steps: reflection and refinement (Shinn et al., 2024). Given a question q and $R_0 = LLM(q)$ representing the initial response of an LLM, self-correction methods further instruct the LLM to reflect on the initial response R_0 and generate feedback by $F = LLM(q, R_0)$. Given R_0 and F, the LLM then generates a revised answer in the refinement stage, denoted as $R_1 = LLM(q, R_0, F)$.

Diverse Sampling. Diverse Sampling methods usually involve three steps: sampling, deliberation, and judging (Zhang et al., 2024b). First, N initial responses are sampled by: $R_0^i = LLM(q, \theta_i), i \in [1, N]$. Here θ_i represents settings such as improving temperature or using different prompts, which are widely used in diverse sampling to enhance the diversity of responses. In the following deliberation stage, each response is refined by contrasting with other responses, thereby improving the initial responses: $R_1^i = LLM(q, R_0^i, \{R_0^{j \neq i}\})$. Finally, a judging process is employed to determine the final answer $R_f = judge(R_1^1, R_1^2, ..., R_1^N)$.

However, the LLM with self-correction or diverse sampling face the issue of overconfidence. Formally, the LLM with self-correction tends to overtrust the initial response R_0, resulting in R_1 having the same error as R_0 (Zhang et al., 2024b). Meanwhile, for diverse sampling, the incorrect answer might repeat in $\{R_0^1, R_0^2, ..., R_0^N\}$, resulting in the deliberation and judging stages potentially accepting such an incorrect answer.

2.2 Investigation of Overconfidence

To investigate the overconfidence issue, we conduct some preliminary experiments on the representative self-correction and diverse sampling methods.

Testing Methods. We evaluate four representative methods and count the number of testing samples exhibiting the overconfidence issue as follows:

- **Self-reflection** (Shinn et al., 2024): This method instructs the LLMs to reflect on an initial answer and subsequently provide feedback, asking the LLM to refine and generate a revised response based on this feedback. If the revised answer for a testing sample remains the same as the initial incorrect response, we treat it as a sample with the overconfidence issue.
- **Self-consistency** (Wang et al., 2023): This approach samples multiple initial answers using the same prompts, followed by a voting process to determine the final answer. We implement it by sampling seven initial answers and consider a test sample as an overconfidence sample if six out of the seven responses are identically incorrect.
- **Self-contrast** (Zhang et al., 2024b): In this method, three initial answers are generated by the LLMs using self-generated, varying prompts. These answers are then contrasted to derive the final answer. If all three initial responses are the same incorrect answers for a given testing sample, it is regarded as an overconfidence sample.
- **MAD** (Du et al., 2023): This strategy involves sampling multiple initial answers from different agents and using a debate process to decide on the final answer. Similarly, an overconfidence sample is defined as that three initial responses are the same and incorrect.

Results. We assess the overconfidence issue by applying these methods to a representative LLM, GPT-3.5-turbo (Ouyang et al., 2022), on the CommonsenseQA (Talmor et al., 2019) and Hover (Jiang et al., 2020) datasets. We calculate the proportion of incorrect answers attributable to overconfidence among all incorrect cases. As shown in Figure 2, for self-reflection, MAD, and Self-Contrast, more than half of the errors are caused by overconfidence. For Self-consistency, although the overconfidence issue is alleviated due to the increase in sample number and temperature, approximately 40% of the errors are still caused by the overconfidence of LLMs. This validates the severity of the overconfidence issue in existing self-correction and diverse sampling methods.

A key reason for the overconfidence issue of LLMs might be that self-correction and diverse sampling methods do not intervene in the LLM's answer-generation process, permitting LLMs to refine and sample diverse answers based on their inherent biases and beliefs. Consequently, LLMs tend to trust the initial incorrect answer, hindering the consideration of alternative potential answers.

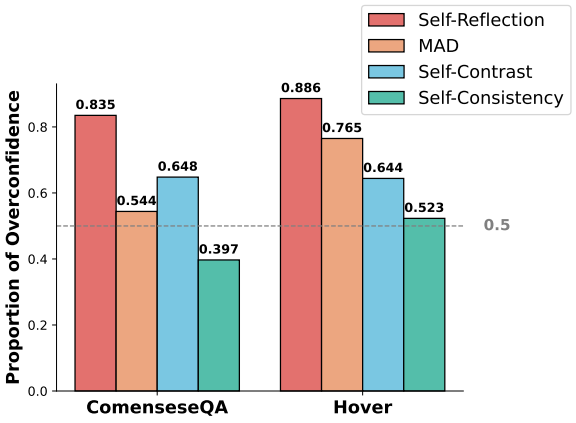

Figure 2: Proportion of the overconfident answers among all incorrect answers.

3 Counterfactual Multi-agent Debate

To address the overconfidence issue, the key lies in overriding the inherent biases and beliefs of LLMs for answer generation. To achieve this, we consider initially configuring the LLMs with various stances, allowing them to hypothesize the correctness of each possible answer and uncover the underlying rationale of each answer. This approach compels LLMs to inspect all potential answers, liberating them from inherent biases and beliefs. Subsequently, we can critically assess the potential rationales to identify the correct answer.

To this end, we propose a CFMAD framework comprising two sequential stages: abduction generation and counterfactual debate, depicted in Figure 3. In the abduction generation phase, we initialize multiple LLM agents and configure each one to adopt a predetermined stance, assuming a specified answer is correct. Subsequently, these agents are instructed to generate abductions, *i.e.,* potential correct reasons for the given answer. In the counterfactual debate phase, we create an adversarial debate scenario. Each abducting agent, adopting a predetermined answer as correct, faces a critical evaluator tasked with challenging the validity of the abductions generated by the agent. Meanwhile, the abducting agent is directed to defend its position on the abduction correctness. Eventually,

Figure 3: Illustration of CFMAD framework with two stages. In the abduction generation stage, we initialize multiple LLM agents, each configured to assume a specific answer is correct and to generate supporting abductions. In the subsequent counterfactual debate stage, each agent is challenged by a critical evaluator for debating. The debating processes are assessed by a third-party judge for final adjudication.

the deliberations between each agent-critic pair are presented to a third-party judge to deliver the final adjudication.

3.1 Abduction Generation

Prior studies have illustrated that LLMs exhibit proficiency in counterfactual reasoning (Nguyen et al., 2024; Bhattacharjee et al., 2024), thereby allowing them to engage in reasoning with predetermined stances. Specifically, given a possible answer a_i, we preset the LLMs' stance with a_i by the following prompt:

> Why is a_i the correct answer? Your answer should look like this: The answer is a_i because ...

Even if a_i is incorrect, the LLM can still follow our instructions to perform counterfactual reasoning to generate plausible justifications.

Drawing from this insight, CFMAD assigns the LLM agents the task of generating abductions for each potential answer. Concretely, as depicted in Figure 3, when presented with a set of possible answers $\{a_1, a_2, ..., a_M\}$, we activate multiple abducting agents. Each agent is tasked with assuming that a specific answer a_i is correct and then generating the corresponding abduction r_i.

3.2 Counterfactual Debate

Among the generated abductions $\{r_1, r_2, ..., r_M\}$, only one is factual, while the remainder are incorrect justifications. Hence, we introduce a counterfactual debate mechanism to discern the correct answer from the pool of abductions. Specifically, for each abducting agent g_i, who is preset with the position that a_i is correct, we introduce a critic evaluator to challenge the correctness of a_i. By showing the agent's abduction r_i to critic c_i and instructing the critic with a prompt like:

> The agent's answer may be wrong. Please persuade the agent that the answer is incorrect.

Simultaneously, we preset the stance of g_i, ensuring it firmly believes in the correctness of its answer and addresses challenges from the critic. For instance, we provide g_i with a prompt such as:

> Please refute the critic's answer and persuade the critic that your answer is correct.

With the aforementioned configuration, we orchestrate an adversarial debate scenario for each agent-critic pair.

The abduction r_i for an incorrect answer a_i inevitably incorporates numerous fabricated reasoning processes and factually incorrect elements. The adversarial debate process will help to unveil the errors or unreasonable justifications in r_i.

After multi-round debating, we present the debate process of all agent-critic pairs to a third-party judge, enabling them to meticulously analyze and juxtapose the varied debate trajectories, thereby discerning the final answer.

4 Experiments

In this section, we conduct extensive experiments on the widely studied fact-checking, reading comprehension, and commonsense reasoning tasks.

4.1 Experimental Setup

Datasets. We conduct experiments on four datasets: Hover (Jiang et al., 2020), BoolQ (Clark et al., 2019), CosmosQA (Huang et al., 2019), and CommonsenseQA (Talmor et al., 2019). Hover and BoolQ are binary prediction tasks with only true or false answers. CosmosQA and CommenseQA are multi-choice tasks with 4 and 5 options, respectively. Note that we split Hover into two subsets named Hover 3-hop and Hover 4-hop with questions requiring 3 and 4 steps of reasoning, respectively. Comparison between Hover 3-hop and Hover 4-hop might reveal the influence of problem difficulty on method effectiveness since more complex questions typically require more reasoning hops. More details about these datasets can be found in the Appendix A.1.

Baselines. As introduced in Sections 2.2, we compare CFMAD with the four baselines: Chain-of-thought (CoT) prompting (Wei et al., 2022), Self-Reflection (Shinn et al., 2024), Self-Consistency (Wang et al., 2023), Self-Contrast (Zhang et al., 2024b), MAD (Du et al., 2023).

Implementation Details. The implementation detail involves three key factors: backbone, prompt, and inference temperature. For all compared methods, we use GPT-3.5-turbo-0613[1] as our backbone LLM and present their prompts in Appendix B and C. As to the inference temperature, we set it to 0.2 in most methods for the sake of fair comparison. The only exception is Self-Consistency, where we follow the original paper and set the temperature to 1 since the method requires high diversity of samples (Wang et al., 2023).

Evaluation Metrics. For binary prediction datasets, *i.e.,* Hover and BoolQ, we follow the previous work (Wang and Shu, 2023) and adopt the macro-F1 score as the evaluation metric. As to multi-choice datasets, *i.e.,* CosmosQA and CommenseQA, we report accuracy following previous work (Wang and Zhao, 2023).

4.2 Performance Comparison

Table 1 shows the performance of the compared methods on all datasets. From the table, we have the following observations:

- In all cases, CFMAD outperforms all baselines, showing stronger reasoning capabilities. Such performance gain indicates the effectiveness of the abduction generation and counterfactual debate mechanism.

- Among all self-correction and diverse sampling methods, Self-Reflection performs the worst in all cases, sometimes even worse than CoT. Given that Self-Reflection encounters the most severe overconfidence issue (as shown in Figure 2), we postulate that such inferior performance is due to overconfidence.

[1] https://chatgpt.com/.

Method	Hover 3-hop	Hover 4-hop	BoolQ	CosmosQA	CommenseQA
CoT	0.6108	0.5886	0.7767	0.7833	0.7467
Self-Reflection	0.5986	0.5813	0.7728	0.7867	0.7567
Self-Consistency	0.6342	0.6044	0.8033	0.8067	<u>0.7733</u>
MAD	<u>0.6476</u>	0.6069	0.8020	0.7933	0.7700
Self-Contrast	0.6359	<u>0.6178</u>	<u>0.8267</u>	<u>0.8133</u>	0.7633
CFMAD (Ours)	**0.6757**	**0.6361**	**0.8366**	**0.8267**	**0.7933**

Table 1: Overall performance comparison on all experiment datasets. Bold font and underline indicate the best and second-best performance, respectively.

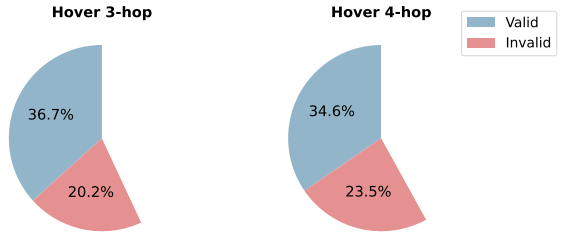

Figure 4: **Proportion of changes in initial stances.** "Valid" means the stances changed from incorrect to correct. "Invalid" represents the stances changed from correct to incorrect.

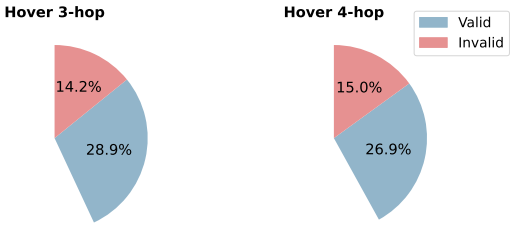

Figure 5: **The final judgment on inconsistent stances.** "Valid" means that the judge makes a correct judgment while "Invalid" denotes making an incorrect judgment.

- While Self-Consistency exhibits lower levels of overconfidence than Self-Contrast by providing answers within a wider scope, it does not consistently outperform Self-Contrast across all tasks. This suggests that incorporating diverse perspectives alone does not guarantee superior reasoning outcomes; the effective utilization of these varied viewpoints is crucial for optimal performance.

4.3 In-depth Analysis

We proceed to analyze the performance enhancement of CFMAD. We posit that the efficacy of CFMAD stems from two key factors: 1) Agents instructed to generate abductions for incorrect answers are more likely to waver and change their stance during the debate process due to the contradictions with factual information. 2) Engaging in counterfactual debates aids judges in distinctly discerning between accurate and inaccurate answers. Subsequently, we undertake experimental investigations to delve into these aspects.

4.3.1 Counterfactual Answers are More Prone to Change

As to stance change, we first analyze whether the agents would change their stance even when instructed to maintain their original position. For simplicity, we conduct our analysis using the Hover dataset with binary answers. Specifically, we first ask two agents to generate abductions for both "True" and "False" answers, respectively. Given that there are only two possible answers, one of these abductions is necessarily factual while the other is counterfactual. Given these abductions, we conduct a single round of counterfactual debate. For both agents with factual and counterfactual abductions, we instruct the critic to persuade the agent that their claim is actually incorrect. After that, we present the critic's argument to the corresponding agents and instruct them to maintain their original stance by pointing out the errors in the critic's answer and reiterating your point. Finally, we observe whether these factual and counterfactual agents would change their stance.

The results on Hover 3-hop and Hover 4-hop are shown in Figure 4. From Figure 4, we find that over 50% of factual and counterfactual agents reached a consensus after one round of counterfactual debate. It means that a significant number of agents were persuaded by the critic, while we instruct these agents to maintain their original stance. Specifically, more than 34% of the stance changes came from counterfactual agents, which is 10% higher than the changes from factual agents. We believe this is because counterfactual answers inherently contradict the facts, making it easier for the critic to point out issues and for the agents to realize the problems and subsequently change their stance.

Method	Hover 3-hop	CosmosQA	CommenseQA
CFMAD	**0.6815**	**0.8267**	**0.7933**
Direct Judge	0.6027	0.7633	0.7500
Repl. w/ SR	0.6063	0.7800	0.7600
Repl. w/ MAD	0.6224	0.6767	0.7200

Table 2: Ablation studies on the effectiveness of our counterfactual debate component.

4.3.2 Counterfactual Debates Contain Additional Clues

We first analyze the contribution of the counterfactual debate by continuing the previous experiment. For those agents that do not reach a consensus, we present the entire debate process between the critic and the factual and counterfactual agents to a third-party judge. The judge then makes a final decision on which stance is more factual. As shown in Figure 5, the number of correct judgments was twice that of incorrect judgments, indicating that even if a consensus is not ultimately reached, leveraging the judge to evaluate the counterfactual debate process can still significantly improve the accuracy of the final decision.

To further investigate the effectiveness of the counterfactual debate, we evaluate several variants of CFMAD, including:

- **Direct Judge:** Removing the counterfactual debate and directly presenting the generated abductions to the judge for final decision.
- **Replace with Self-Reflection:** Replacing the counterfactual debate with self-reflection, where the LLM reflects on each generated abduction. Both the original answer and the reflection process were shown to the judge for final decision.
- **Replace with MAD:** Replacing the counterfactual debate with three rounds of MAD (Du et al., 2023), then presenting the MAD debate process to the judge for the final decision.

We conduct the ablation experiments on three datasets. For CosmosQA and CommenseQA, we used the same 300 data as in Table 1. For Hover 3-hop, we randomly sampled 300 data points due to cost limitations. The results are shown in Table 2. We can see that our proposed counterfactual debate component outperforms the other control group across all tasks. This demonstrates that the counterfactual debate component helps the judge more effectively determine the correct final answer.

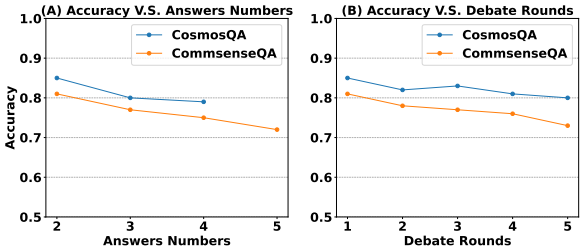

Figure 6: Comparison of different numbers of (A) initial counterfactual answers and (B) debate rounds.

4.4 Impact of Hyperparameters

We then investigate the influence of hyperparameters on the effectiveness of CFMAD, including the number of initial counterfactual answers and debate rounds.

Number of Initial Counterfactual Answers. Considering that datasets like CosmosQA and CommenseQA have multiple potential answers, we explore the influence of initial counterfactual answers by increasing the number of sampled stances. Note that directly sampling a few stances from many options may fail to include the correct answer when the initial number of stances is small (*e.g.,* 2 out of 5 choices). We thus need to conduct the comparison under the condition that the correct answer is included. To this end, we use a CoT prompt to generate three answers and select the most frequently occurring answer as the most potential stance. We then randomly sample the remaining stances to complete the initial settings. Considering the expensive time and monetary costs, we randomly sampled 100 data from each dataset. The final result is shown in Figure 6(A), where the accuracy of the final judgment decreases as the number of initial counterfactual responses increases. We believe this is because the presence of too many incorrect stances can confuse the LLMs. Notably, only two initial counterfactual answers are needed to achieve good results, which also saves time and cost.

Number of Debate Rounds. We also test the impact of conducting multiple rounds of counterfactual debate. As shown in Figure 6(B), the accuracy decreases with the increase of debate rounds. We speculate that through multiple rounds of debate, LLM-based agents and critics may veer away from our predetermined stances to adhere to the biases in the LLM itself, thereby influencing the efficacy of the debate. As such, we conduct only one-round debate between the agent and critic by default.

5 Related Work

Prompting LLM for Better Reasoning. Researchers have made significant progress in improving the reasoning abilities of LLMs through designing better prompting methods. These methods often enhance the LLM's reasoning capabilities in either reasoning depth or breadth. CoT prompting (Wei et al., 2023) guides the model to generate intermediate reasoning steps before arriving at a final answer, thus improving the reasoning depth. Self-correction methods (Madaan et al., 2024; Shinn et al., 2024; Paul et al., 2023; Xi et al., 2024) are also typical examples of enhancing LLM reasoning depth. They leverage the LLM's self-correction ability, generating feedback by LLM itself to iteratively refine its answers, thereby enhancing its accuracy and reliability. Breadth reasoning approaches, on the other hand, involve sampling diverse responses with temperature larger than 0 (Wang et al., 2023; Yoran et al., 2023) or guiding the LLM to generate responses from different perspectives (Huang et al., 2024a; Zhang et al., 2024b), gathering more diverse insights for the answer. This helps to derive the correct answer from the collection of a wider range of potential responses to improve the overall reliability.

Multi-agent Debate. Recent research has explored how to engage multiple agents of the same model or different models in debates to jointly improve decision-making and reasoning processes (Du et al., 2023; Liang et al., 2023; Wang et al., 2024b), which can be divided into two modes: collaborative and adversarial. In the collaborative mode, each agent provides its own answer to the same question and then refines its answer with reference to the responses of other agents (Du et al., 2023). This mode may encounter overconfidence issues that the initial responses of most agents arrive at the same incorrect answer. In the adversarial mode, for a given answer, two agents are initialized: one believing the answer is correct, and the other believing the answer is incorrect, and they are instructed to debate and challenge each other's response to reach a more precise conclusion (Liang et al., 2023; Wang et al., 2024b). The difference between our counterfactual debate and the adversarial debate lies in that they first have the LLM generate a single answer and then conduct a debate about that answer, while we first have the LLM explore multiple answers as thoroughly as possible, and then conduct debates for each of these answers. Additionally, existing work also leverages multiple side rationales in LLM reasoning (Jung et al., 2022; Liu et al., 2023; Balepur et al., 2023) which is similar to our abduction, yet not all of them has shown promising results. We incorporate them in the counterfactual debate process and achieve enhanced reasoning.

Confidence Calibration. Recently, confidence calibration for LLMs has gained significant attention (Lin et al., 2022; Kuhn et al., 2023; Huang et al., 2023; Tian et al., 2023). The goal of confidence calibration is to obtain LLM's confidence score on its own answer which aligns with the actual answer accuracy. However, some studies found that LLMs sometimes generate confidence scores that are poorly calibrated and often assign high confidence scores to incorrect answers (Shrivastava et al., 2023; Yang et al., 2024; Xiong et al., 2024). Some methods attempt to calibrate the confidence for LLMs through estimating response consistency across multiple perspectives (Zhang et al., 2024a; Wang et al., 2024a), and various prompting strategies for LLM to self-estimate the confidence (Tian et al., 2023; Kadavath et al., 2022; Li et al., 2024), where some work also leverages explanation and rationales (Li et al., 2024; Feng et al., 2024). However, these works mainly aim at improving calibration errors or identifying incorrect answers instead of directly improving the answer accuracy.

6 Conclusion

In this paper, we addressed the overconfidence issue presented in existing self-correction and diverse sampling methods for hallucination elimination in LLM reasoning. We revealed the overconfidence issues of these two methods through experiments, and pointed out that the overconfidence issue mainly stems from the LLM's inherent biases towards overly favoring a particular answer while lacking sufficient exploration of other potential answers. To address this, we proposed the CFMAD framework, which first presets the stance for the LLM, encouraging it to explore as many answers as possible, and then uses counterfactual debate to expose and correct the errors in the incorrect answers. Empirical results validate the superiority of CFMAD over baselines in mitigating hallucinations. In this work, we mainly test CFMAD on binary and multiple-choice questions. In the future, we intend to extend CFMAD to more scenarios with open-ended questions.

Limitations

Our work has the following limitations: First, we require the LLM to generate reasons for each possible answer and conduct debates for each answer, which results in additional computational overhead. Secondly, since it is necessary to preset the stance for the LLM, we must identify potential answers. We address this by initially using CoT prompts sampling to generate three possible answers. However, it is worth exploring superior methods to improve the recall rate of correct answers.

Ethics Statement

Our ethical concerns include the following points. First, although we can mitigate LLM hallucinations using CFMAD, the LLM may still produce some inaccurate answers, which could potentially cause harm. Secondly, our experiments are conducted exclusively on English datasets, meaning the applicability of our findings to other languages has not been comprehensively evaluated.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Nishant Balepur, Shramay Palta, and Rachel Rudinger. 2023. It's not easy being wrong: Evaluating process of elimination reasoning in large language models. *arXiv preprint arXiv:2311.07532*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Amrita Bhattacharjee, Raha Moraffah, Joshua Garland, and Huan Liu. 2024. Zero-shot llm-guided counterfactual generation for text. *arXiv preprint arXiv:2405.04793*.

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

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't hallucinate, abstain: Identifying llm knowledge gaps via multi-llm collaboration. *arXiv preprint arXiv:2402.00367*.

Baizhou Huang, Shuai Lu, Weizhu Chen, Xiaojun Wan, and Nan Duan. 2024a. Enhancing large language models in coding through multi-perspective self-consistency. *Preprint*, arXiv:2309.17272.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024b. Large language models cannot self-correct reasoning yet. *Preprint*, arXiv:2310.01798.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. *arXiv preprint arXiv:1909.00277*.

Yuheng Huang, Jiayang Song, Zhijie Wang, Huaming Chen, and Lei Ma. 2023. Look before you leap: An exploratory study of uncertainty measurement for large language models.

Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. 2020. Hover: A dataset for many-hop fact extraction and claim verification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. 2022. Maieutic prompting: Logically consistent reasoning with recursive explanations. *arXiv preprint arXiv:2205.11822*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation.

Moxin Li, Wenjie Wang, Fuli Feng, Fengbin Zhu, Qifan Wang, and Tat-Seng Chua. 2024. Think twice before assure: Confidence estimation for large language models through reflection on multiple answers. *arXiv preprint arXiv:2403.09972*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *arXiv preprint arXiv:2305.19118*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*.

Ziyi Liu, Isabelle Lee, Yongkang Du, Soumya Sanyal, and Jieyu Zhao. 2023. Score: A framework for self-contradictory reasoning evaluation. *arXiv preprint arXiv:2311.09603*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36.

Sabrina J Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. 2022. Reducing conversational agents' overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872.

Van Bach Nguyen, Paul Youssef, Jörg Schlötterer, and Christin Seifert. 2024. Llms for generating and evaluating counterfactuals: A comprehensive study. *arXiv preprint arXiv:2405.00722*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2023. Refiner: Reasoning feedback on intermediate representations.

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

Vaishnavi Shrivastava, Percy Liang, and Ananya Kumar. 2023. Llamas know what gpts don't show: Surrogate models for confidence estimation. *Preprint*, arXiv:2311.08877.

Kaya Stechly, Matthew Marquez, and Subbarao Kambhampati. 2023. Gpt-4 doesn't know it's wrong: An analysis of iterative prompting for reasoning problems. *arXiv preprint arXiv:2310.12397*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *Preprint*, arXiv:1811.00937.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *Preprint*, arXiv:2305.14975.

Karthik Valmeekam, Matthew Marquez, and Subbarao Kambhampati. 2023. Can large language models really improve by self-critiquing their own plans? *arXiv preprint arXiv:2310.08118*.

Haoran Wang and Kai Shu. 2023. Explainable claim verification via knowledge-grounded reasoning with large language models. *arXiv preprint arXiv:2310.05253*.

Pei Wang, Yejie Wang, Muxi Diao, Keqing He, Guanting Dong, and Weiran Xu. 2024a. Multi-perspective consistency enhances confidence estimation in large language models. *arXiv preprint arXiv:2402.11279*.

Qineng Wang, Zihao Wang, Ying Su, Hanghang Tong, and Yangqiu Song. 2024b. Rethinking the bounds of llm reasoning: Are multi-agent discussions the key? *arXiv preprint arXiv:2402.18272*.

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

Yuqing Wang and Yun Zhao. 2023. Gemini in reasoning: Unveiling commonsense in multimodal large language models. *Preprint*, arXiv:2312.17661.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

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.

Zhiheng Xi, Senjie Jin, Yuhao Zhou, Rui Zheng, Songyang Gao, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Self-polish: Enhance reasoning in large language models via problem refinement. *Preprint*, arXiv:2305.14497.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *Preprint*, arXiv:2306.13063.

Ruixin Yang, Dheeraj Rajagopa, Shirley Anugrah Hayati, Bin Hu, and Dongyeop Kang. 2024. Confidence calibration and rationalization for llms via multi-agent deliberation. *arXiv preprint arXiv:2404.09127*.

Ori Yoran, Tomer Wolfson, Ben Bogin, Uri Katz, Daniel Deutch, and Jonathan Berant. 2023. Answering questions by meta-reasoning over multiple chains of thought. *arXiv preprint arXiv:2304.13007*.

Mozhi Zhang, Mianqiu Huang, Rundong Shi, Linsen Guo, Chong Peng, Peng Yan, Yaqian Zhou, and Xipeng Qiu. 2024a. Calibrating the confidence of large language models by eliciting fidelity. *arXiv preprint arXiv:2404.02655*.

Wenqi Zhang, Yongliang Shen, Linjuan Wu, Qiuying Peng, Jun Wang, Yueting Zhuang, and Weiming Lu. 2024b. Self-contrast: Better reflection through inconsistent solving perspectives. *Preprint*, arXiv:2401.02009.

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*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Shen Zheng, Jie Huang, and Kevin Chen-Chuan Chang. 2023. Why does chatgpt fall short in answering questions faithfully? *arXiv preprint arXiv:2304.10513*.

A Experiments Details

A.1 Dataset Details

We performed experiments using four datasets: Hover, BoolQ, CosmosQA, and CommonsenseQA. The details of these datasets are as follows:

- **Hover:** Hover is a fact-checking task dataset. Each instance in the Hover dataset consists of a claim and supporting evidence. The task requires multi-hop reasoning based on the supporting evidence to determine whether the evidence supports the claim or not.

- **BoolQ:** BoolQ is a reading comprehension task dataset that consist of questions that can be answered with a simple "yes" or "no". And each question is paired with a paragraph from Wikipedia that contains the answer.

- **CosmosQA:** CosmosQA is a dataset focused on reading comprehension and commonsense reasoning. Each instance consists of a context and a question with four answer options that require inference beyond the text, using commonsense knowledge to determine the correct answer.

- **CommonsenseQA**: CommonsenseQA is a challenging dataset that tests a model's ability to use commonsense knowledge to answer multiple-choice questions. Each question has one correct answer and four distractors.

In this work, we first tested all 3-hop and 4-hop instances in the validation set of Hover, with 1,835 instances for 3-hop and 1,039 instances for 4-hop to demonstrate our method's effectiveness. Next, due to budget constraints, we randomly selected 300 instances from the validation set of each of the remaining three datasets to conduct our experiments.

A.2 Method Implementation Details

For MAD, we initialized 3 agents and conducted 3 rounds of debate. For Self-Contrast, we had the LLM initially generate answers from 3 perspectives for subsequent contrast. For Self-Consistency, we initially generated 7 answers, voting for the final answer. For our CFMAD framework, we initially preset two predetermined answers to instruct the LLMs to generate abduction. For datasets like CosmosQA and CommonsenseQA, which have multiple potential answers, we first use 3 rounds of CoT prompting to obtain one potentially correct answer as a predetermined answer. Then, we randomly select another predetermined answer from the remaining options.

B Basline Prompts

B.1 CoT Prompt

- **Fact Check Task**
  Evidence: {evidence}
  Claim: {claim}
  You are a fact checker. Please fully understand the evidence and claim, and answer is the claim true or false? Let us verify step by step.

- **Commonsense Resoning**
  {Question and option here}

  Play the role of a common sense reasoning expert. Choose the most appropriate answer for the question. You are expected to explain your

reasoning process step-by-step before providing the final answer.

Output format:
Reasoning steps: [Your precise reasoning steps here]
Judgement: The correct answer is Option [X].

B.2 Relfection Prompt

- **Reflection Prompt**
  As a critic, review the assistant's response. Identify any incorrect or missing information, and provide feedback.
  {Question Content Here}
  Assistant's reply: {CoT_reply}
  Output format:
  Judgement: [Critically evaluate the assistant's response.]
  Potential Improvements: [Suggest ways to enhance the accuracy or clarity of the assistant's response.]

- **Revision Prompt**
  {Question Content Here}
  Assistant's reply: {CoT_reply}
  Feeback: {reflection_reply}
  Based on the feedback provided, revise your response to the question.
  Output format:
  The correct answer is Option [X].

B.3 MAD

Here we show the prompt for CommonsenseQA. The prompt structure is similar for other tasks, and the specific prompts for other tasks can be found in our code.

- **Initial Prompt 1**
  {Question Content Here}
  Play the role of a common sense reasoning expert. Choose the most appropriate answer for the question. You are expected to explain your reasoning process step-by-step before providing the final answer.

- **Initial Prompt 2**
  {Question Content Here}
  Which option is the most appropriate answer based on the common sense?

- **Initial Prompt 3**
  {Question Content Here}
  Let us think step by step and find the most appropriate answer based on the common sense.

- **Debate Prompt**
  {Question Content Here}
  Let us think step by step and find the most appropriate answer based on the common sense.
  Assistant: {Your previous response}
  Other agent1: {Other agents' previous responses1}
  Other agent2: {Other agents' previous responses2}
  Using the judgements from other agents as additional information, can you give an updated response.

- **Judge Prompt**
  {Question Content Here}
  Let us think step by step and find the most appropriate answer based on the common sense.
  Agent1: {last response of agent 1}
  Agent2: {last response of agent 2}
  Agent3: {last response of agent 3}
  Three agents have given their answers.
  According to the majority of the answers, what is the most appropriate answer? Your answer should look like this: "The correct answer is Option [X]"

B.4 Self-contrast

Here we show the prompt for CommonsenseQA. The prompt structure is similar for other tasks, and the specific prompts for other tasks can be found in our code.

- **Self-Curate Prompt**
  You are a commonsense reasoning specialist. You need to complete multiple choice questions related to commonsense reasoning. Given a question, you need to carefully analyze the question and dynamically generate several useful prompt instructions. These prompt instructions should be diverse and also useful for commonsense reasoning. These prompt instructions are used to guide the language model to think in different ways, attention to different emphases, and reason from different perspectives for more accurate commonsense reasoning.
  For instance, you can adopt multi-faceted thinking (logical thinking, lateral thinking, analogical thinking, etc .), different reasoning perspectives(e.g., top-down, bottom-up , step-by-step), and different emphases of concern, (entity words, numbers, time, etc) for input question in prompt instruction.

Here are some guidance rules for Prompt Generation:
1. Tone Requirement: Please generate prompt instructions in the third person.
2. Content Requirement: Each prompt instruction should adopt a different way of thinking, or focus on a different perspective, or different emphases to solve the question.
3. Number Requirement: Dynamically generate the most valuable 3 prompt instructions based on the input math question.
4. Format Requirement: Each prompt instruction should start with ### and end with @@@
5. Others: Prompt instructions should focus on commonsense reasoning. So don't ask any other irrelevant questions in the prompt.
Here is an example : The question is: Who is the first president of the United States?
Output:
bottom - up perspective : ### As a specialist in commonsense reasoning, you have to judge the given question from a bottom-up perspective. Breaking the question down into smaller components or details. What specific pieces of information are provided in the question, and how do they contribute to understanding the problem? @@@
The input question is: {question}. Please generate the most suitable three prompts:

- **Contrast Prompt**
  You are a specialist in commonsense reasoning. Given some candidate judgements for a question, you should carefully compare the difference for each two judgements in their reasoning steps.
  When you compare, you need to consider the following questions:
  1: Are the two judgements have different final judge and judge reasons?
  2: Where are the differences in their reason steps and judge reasons?
  3. Why are the answers of the two judgements different?
  After contrasting , you should generate a checklist based on these differences between candidate judgements . You should carefully consider each discrepancy and the reasons behind it, summarizing them into a few checking instructions in the checklist. This checklist can guide others to reexamine the input question and these candidate judgements to eliminate these discrepancies .
  {Question Content Here}
  Judgements:
  Judgement1: {reply1},
  Judgement2: {reply2},
  Judgement3: {reply3}
  Output Format:
  For Judgement1 and Judgement2 : [Give the difference between Judgement1 and Judgement2 here]
  For Judgement1 and Judgement3 : [Give the difference between Judgement1 and Judgement3 here]
  For Judgement2 and Judgement3 : [Give the difference between Judgement2 and Judgement3 here]
  Checklist : [Give the directives for checking here]

- **Reflection Prompt**
  Given a question, multiple inconsistent judgements, their differences in their reasoning processes and a checklist. You should revise the inconsistent reasoning step for each judgements, eliminate the differences, and output a new judgement.
  Guidance Rules for Reflection:
  1. Please check carefully according to the requirements on the checklist. It helps you to resolve conflicts between different judgements.
  2. When you finish revising inconsistent judgements, please ensure all revised judgements should have the same answer . If not , please revise again until all inconsistencies are removed , and all candidates are consistent.
  {Question Content Here}
  The candidate judgements and their discrepancy are as follows:
  {
  "Candidate": {
  "Judgement": "{reply1}",
  "Judgement": "{reply2}",
  "Judgement3": "{reply3}"
  },
  "Discrepancy": {
  "difference_1_2": {
  "source": "Judgement1",
  "target": "Judgement2",
  "relation": {difference_1_2}
  },
  "difference_1_3": {
  "source": "Judgement1",
  "target": "Judgement3",
  "relation": {difference_1_3}

},
"difference_2_3": {
"source": "Judgement2",
"target": "Judgement3",
"relation": {difference_2_3}
} } }
Checklist: {checklist}
Please revise each inconsistent judgement and give your final judgement.
Output Format:
The answer is Option [X].

C Our Prompts

C.1 Fact Check Task

- **Abduction Generation**
  Evidence: {evidence}
  Claim: {claim}
  Please fully understand the evidence and claim, and answer why the claim is {true/false}?

- **Counterfactual Debate for Critic**
  Evidence: {evidence}
  Claim: {claim}
  Assistant: {reply of assistant}
  The Assistant's answer maybe wrong. Please persuade the assistant that the claim is actually incorrect based on the evidence.

- **Counterfactual Deabte for Assistant**
  Evidence: {evidence}
  Claim: {claim}
  Please fully understand the evidence and claim, and answer why the claim is true?
  Fact checker: {reply of assistant}
  Critic: {reply of crtic}
  Play the role of fact checker. Please point out the errors in critic's answer and reiterate your point.

- **Judge**
  Evidence: {evidence}
  Claim: {claim}
  {Debate Process for each stance}
  After hearing the positive and negative sides, do you think the claim is true or false? [True/False]

C.2 Commonsense Reasoning

- **Abduction Generation**
  {Question Content Here}
  Try to explain why the question's answer might be option {predetermined answer}.
  Output Format:
  Judgement: The answer is option {predetermined answer}.
  Reasoning: [Your reasoning here]

- **Counterfactual Debate for Critic**
  {Question Content Here}
  Assistant: {reply of assistant}
  The Assistant's answer maybe wrong. Please persuade the assistant that his answer maybe wrong.

- **Counterfactual Deabte for Assistant**
  {Question Content Here}
  Assistant: {reply of assistant}
  Critic: {reply of critic}
  As assistant, please refute the critic's answer and persuade the critic that your answer is correct.

- **Judge**
  {Question Content Here}
  Which option is the answer of the question? The results of the analysis for each of the possible options are as follows:
  {Debate Process for each stance}
  After seeing the debate process above, do you think which option is the most appropriate answer for the question? Please only give a correct answer and no other replies.
  Output format:
  Judgement: The correct answer is Option [X].
  Reasoning steps: [Your precise reasoning steps here]