Confidence-Guided Cross-Premise Contrastive Decoding for Enhanced LLMs Reasoning

Anonymous ACL submission

Abstract

Large language models (LLMs) are prone to distraction by contextual information during reasoning. Previous work primarily focuses on improving the generation of the next token while overlooking the potential bias introduced by existing premises. In this paper, we propose a novel decoding method to mitigate this issue. We establish a framework that uses predicted logits to assess the model's confidence. By decomposing the full context into multiple premises, we gain a clearer understanding of the relevance of each premise to the question. When predicting the next token, we adjust the original model output by contrasting the most confident logits with the least confident ones. Our method effectively reveals how the model dynamically activates and adjusts its consideration of each premise as reasoning progresses.

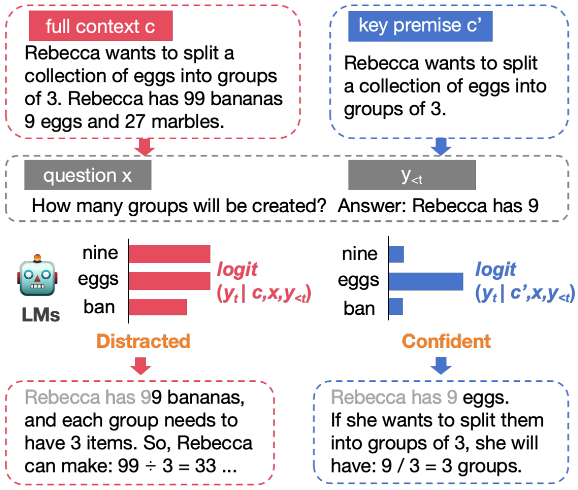

Figure 1: An illustration of a reasoning task. The language model becomes distracted by semantic coherence, thereby leading to error accumulation.

1 Introduction

Large language models (LLMs) have demonstrated significant effectiveness across various reasoning tasks (Ahn et al., 2024; Zhang et al., 2024b). With the continuous advancement of LLMs' capabilities, generating step-by-step intermediate rationale can effectively guide the model toward reliable answers (Wei et al., 2022).

A multitude of research endeavors has been dedicated to optimizing the intermediate reasoning process of LLMs during inference time (Snell et al., 2024). These efforts can be categorized into two paradigms: *1) Fusion-based approaches*, which leverage additional information from the model itself or external sources to bolster the robustness of reasoning (Li et al., 2023; O'Brien and Lewis, 2023; Shi et al., 2024b). *2) Reasoning space search-based approaches*, which search for the optimal solution across various possible reasoning paths to derive the answer (Wang and Zhou, 2024; Xie et al., 2023, 2024; Mo and Xin, 2024).

However, previous research primarily focuses on how to enable LLMs to generate better next tokens or rationales, while overlooking the influence of the premise and context in the question on the subsequent generation (Liu et al., 2024; Chen et al., 2024). Since LLMs are autoregressive architectures, the existing context is typically closely tied to the generation of new tokens, encompassing aspects such as grammatical correctness, instruction adherence, and semantic coherence. Yet, when tackling reasoning tasks, due to the intricate logical relationships involved, the models often struggle to capture the appropriate contextual cues, resulting in an unrealistic token probability distribution. This distribution can lead to biased reasoning sequences, and errors are amplified as they accumulate.

We argue that the challenge of LLMs being prone to distraction still poses a threat to reasoning tasks (Shi et al., 2023). Due to the implicit attention mechanisms employed by LLMs, it is difficult to discern the relationship between the generated tokens and the premises in the question (Malkin et al., 2022). For instance, models tend to prioritize maintaining both syntactic and semantic coherence while neglecting the correctness of reasoning, as illustrated in Figure 1. The way the model conditions various contexts does not align with expectations, and this issue is difficult to correct externally.

To address these challenges, we propose a **C**onfidence-guided **C**ross-premise **C**ontrastive **D**ecoding (**C^{3}D**) method to enhance the transparency of premises during LLMs' reasoning. Through empirical experiments, we observe that LLMs tend to perform better when faced with simple and explicit instructions (Prystawski et al., 2023; Lightman et al., 2023), as such instructions have lower uncertainty and are easier to execute. Therefore, we first decompose the reasoning problem into multiple premises. When generating the next token, we simultaneously decode the current position using both the multiple premises and the question. Since one premise will closely enlighten the token at the current position, LLMs will assign higher confidence to the token generation under that premise. We then use the premise with the highest confidence and the premise with the lowest confidence for contrastive enhancement to adjust the probability distribution of the next token. This effectively reduces the reasoning bias caused by ambiguous contextual evidence in the model.

We validate our method on multiple arithmetic and symbolic reasoning tasks. The experiments show that our approach significantly improves performance without training, external verifier, or extensive path search. Additionally, our method provides greater transparency and interpretability, helping us better understand the reasoning process of LLMs. In summary, our contributions are three-fold:

- We propose a reasoning enhancement method based on cross-premise awareness and contrastive decoding, in which we design token-level confidence evaluation to support the reliability of the model's reasoning chain.
- Our method effectively reveals how language models dynamically awaken their consideration of different premises as the reasoning process flows. We also visualize the influence of each premise on the generation of downstream tokens.
- Our method can achieve stable improvements in reasoning performance without the need for training, external verifiers, or path search. Extensive experiments validate the effectiveness of our approach.

2 Related Works

2.1 Large Language Models Reasoning

When confronting reasoning tasks, LLMs typically require CoT (Chain-of-Thought) (Wei et al., 2022) capabilities to perform step-by-step intermediate reasoning. Many studies focus on constructing more data to strengthen the underlying CoT abilities of LLMs, including methods based on Supervised Fine-Tuning (SFT) (Hao et al., 2024; Luo et al., 2023; Ranaldi and Freitas, 2024), Reinforcement Learning (RL) (Lightman et al., 2023; Zhang et al., 2024a), and Prompting techniques (Kojima et al., 2022; Zhang et al., 2022). These approaches alter the model's output logic and often demand high-quality data or evaluation models, as well as significant human effort and training costs.

2.2 Inference Time Scaling

In addition to training with more data, another technical approach explores improving LLMs during inference time (Snell et al., 2024). These methods aim to enhance the overall reasoning quality by designing effective supervision strategies for each step of the model's output, and it does not alter the model's inherent capabilities. Some studies employ internal or external auxiliary mechanisms to improve the robustness of LLMs (Li et al., 2023; Chang et al., 2023), while others opt for more direct approaches to search for optimal solutions within diverse reasoning spaces (Wang and Zhou, 2024; Xie et al., 2023, 2024; Mo and Xin, 2024). Our method falls into the category of internal model enhancement, which is low-dependency and low-overhead.

2.3 Contrastive Decoding

By contrasting a credible state with a non-credible state, contrastive decoding injects logits into the token generation process, thereby enhancing the faithfulness of the model's output from within (Shi et al., 2024a). For example, Contrastive Decoding (CD) (O'Brien and Lewis, 2023) uses an expert LM and an amateur LM to contrast and improve the professionalism of the generated tokens. Context-Aware Decoding (CAD) (Shi et al., 2024b), on the other hand, contrasts problems with and without context within a single LM to reduce the irrelevance of tokens to the context. Decoding by Contrasting Layers (DoLa) (Chuang et al., 2023) stimulates the intrinsic knowledge of LMs by contrasting different layers. COIECD (Yuan et al., 2024) utilizes information entropy to address the issue of knowledge conflicts in models. Similarly, our method contrasts generations under different premises and further filters them based on confidence levels.

3 Method

We now introduce our proposed **C**onfidence-guided **C**ross-premise **C**ontrastive **D**ecoding (**C^{3}D**) method, which is a token-level, fine-grained premise-aware contrastive approach.

For a reasoning task, given an input question x and a context c that contains the necessary premises for reasoning, the generation process of a standard large language model $\mathcal{M}$ can be defined as:

$$\begin{aligned} y_t &\sim p_{\mathcal{M}}(y_t|c, x, y_{<t}) \\ &\propto \exp\big(\text{logit}_{\mathcal{M}}(y_t|c, x, y_{<t})\big) \end{aligned} \tag{1}$$

where y_t is the new token generated at time step t based on the context c, the question x, and the previously generated sequence $y_{<t}$. It is sampled proportionally to the logit scores processed by $\mathcal{M}$ (Shi et al., 2024b).

However, the default sampling method is influenced by various factors. For instance, when the information in the context is complex and unclear, the predictions of language models tend to exhibit uncertainty (Zheng et al., 2023; Chen et al., 2024; Qiu and Miikkulainen, 2024), manifested as a smooth distribution over the logits (Ulmer et al., 2023). This smooth distribution further leads to an averaging of sampling probabilities. Once the model selects an incorrect token, subsequent generations are affected as well. Even when the temperature is set to 0, it is difficult to guarantee that the top-ranked token is always correct. Moreover, to maintain linguistic coherence, the model will amplify these cumulative errors, ultimately compromising the correctness of the reasoning.

3.1 Confidence Estimation with Logits

To further explore the internal prediction mechanisms of the model, some methods utilize the logit lens (Belrose et al., 2023) for interpretability analysis. By observing the logits or probability distribution at the final layer, we can understand how the model assigns weights to each word in the vocabulary (Qiu et al., 2024; Yuan et al., 2024).

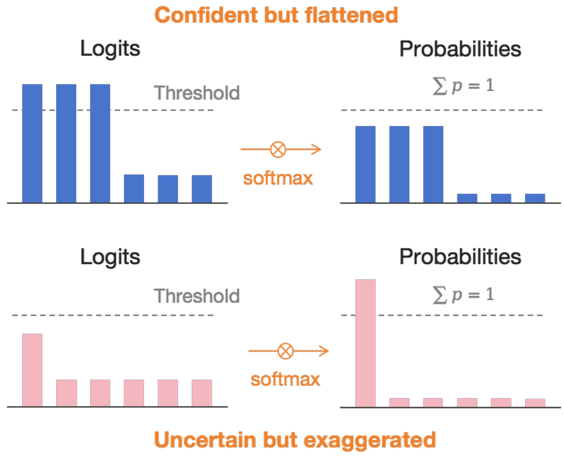

Figure 2: An example where entropy-based probability is insufficient to measure the model's confidence.

Generally, when a word is assigned a weight significantly higher than others, it indicates that the model has high confidence in this word, and it is highly likely to be reasonable and reliable (Zhang et al., 2023; Duan et al., 2024). This situation typically occurs in cases such as common collocations or when the intent is clear. Therefore, we can use the entropy of the predicted probabilities to measure the model's confidence α in the next token:

$$H = -\sum_{\tau \in \mathcal{V}} p_{\mathcal{M}}(\tau) \log\big(p_{\mathcal{M}}(\tau)\big) \tag{2}$$

$$\alpha(y_t) = \frac{1}{\exp(H_{y_t})} \tag{3}$$

where H is the entropy at the current position over the vocabulary $\mathcal{V}$. We further take the negative exponential of the entropy as an estimate of confidence. When the entropy is higher, the probability distribution over the vocabulary is more uniform, and the confidence is lower; when the entropy is lower, the distribution over the vocabulary becomes "sharper", and the confidence is higher (with a maximum value of 1).

Considering that entropy does not always represent the model's uncertainty, as some information is lost during the softmax process (Gupta et al., 2024; Ma et al., 2025). For example, the model might consider multiple words to be reasonable, each assigned a high logit value, but after softmax, their probabilities become averaged. Alternatively, the model might be uncertain about the response, but when all logits are low, softmax can still increase the probability of a particular word, as illustrated in Figure 2. Given this, we incorporate consideration

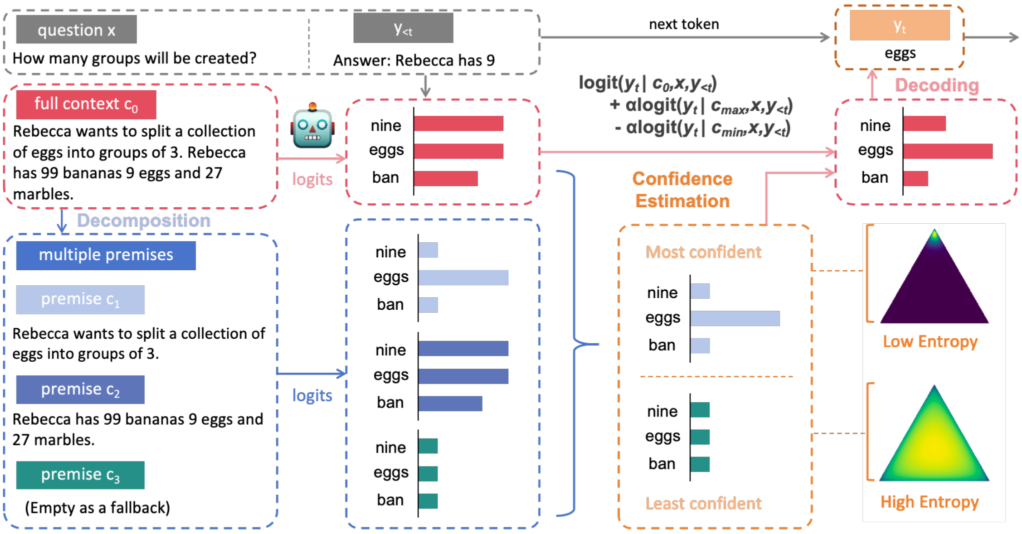

Figure 3: An illustration of our proposed C^{3}D method. The full context is decomposed into multiple premises, which then simultaneously obtain logits for the current position of the original question. By contrasting the most confident and least confident logits, the standard decoding process can be enhanced. This approach effectively mitigates the model's distraction issue. The illustration of entropy is copied from (Ulmer et al., 2023).

of the extreme values of logits:

$$\mathcal{L}(y_t) = \frac{1}{K}\sum_{k=1}^{K} \text{topk}\big(\text{logit}_{\mathcal{M}}(y_t)\big) \tag{4}$$

where topk$(\cdot)$ extracts the largest k values from the logits. The idea behind this is that the magnitude of $\mathcal{L}$ also serves as an indicator of confidence (Ulmer et al., 2023).

3.2 Multi-Premises Decomposition

Inspired by empirical experiments, we observe that models often perform better when given simple and focused instructions (Prystawski et al., 2023; Lightman et al., 2023). This is because simple instructions typically have lower uncertainty, making it easier for the model to capture the key information. Therefore, we decompose the original context c into multiple simpler premises:

$$c = \{c_1, c_2, ..., c_n\} \tag{5}$$

where each premise c_n is a sentence from the context. This can be easily achieved through sentence segmentation.

Then, we can obtain the confidence level of each premise for the current position:

$$\alpha_n = \alpha(y_t|c_n, x, y_{<t}) \tag{6}$$

$$\mathcal{L}_n = \mathcal{L}(y_t|c_n, x, y_{<t}) \tag{7}$$

The hypothesis here is that when a premise is informative for the current decoding position, it will be assigned higher confidence. We aim to identify such premises and enable the model to distinguish the key information in the context from redundant details.

3.3 Dynamic Contrastive Decoding

To overcome reasoning errors caused by contextual distractions, we recompute the predicted logits during the decoding phase. Specifically, we select the premise logit with the highest confidence as the positive example and the premise logit with the lowest confidence as the negative example. We use their contrastive difference to adjust the original logits. Note that when the $\mathcal{L}$ values of all premises fall below a certain threshold, they are all considered untrustworthy, and in such cases, we rely solely on α as the confidence measure. Otherwise, we simply use $\mathcal{L}$ as our basis.

$$c_{max} = \begin{cases} \arg\max_{c_n}\{\mathcal{L}_0, \mathcal{L}_1, ..., \mathcal{L}_n\} \text{ if } \exists \mathcal{L} \geq T \\ \arg\max_{c_n}\{\alpha_0, \alpha_1, ..., \alpha_n\} \text{ if } \forall \mathcal{L} < T \end{cases} \tag{8}$$

$$\begin{aligned} \text{logit}'_{\mathcal{M}}(y_t|c, x, y_{<t}) &= \text{logit}_{\mathcal{M}}(y_t|c, x, y_{<t}) \\ &+ \alpha_{max}\text{logit}_{\mathcal{M}}(y_t|c_{max}, x, y_{<t}) \\ &- \alpha_{max}\text{logit}_{\mathcal{M}}(y_t|c_{min}, x, y_{<t}) \end{aligned} \tag{9}$$

where T is an empirically determined threshold, and $\mathcal{L}_0$ and α_0 denote the confidence of the full context. This decoding process is performed sequentially, and it dynamically selects a pair of contrastive examples for each generated token. Meanwhile, the confidence level α scales the magnitude of this adjustment. As a result, this method can mitigate the model's distraction by contextual information. Figure 3 presents the overall framework.

Algorithm 1 Confidence-guided Cross-premise Contrastive Decoding

Require: A reasoning task x with context c, and a language model $\mathcal{M}$
Ensure: Response sequence $y = \{y_1, y_2, ..., y_t\}$

Decompose c into premises $\{c_1, c_2, ..., c_n\}$
Add the full context and an empty set to the premise set $\mathcal{C} = \{c, c_1, c_2, ..., c_n, \varnothing\}$
while t < max_length **do**
  Logit list $\leftarrow \varnothing$
  for $c_i \in \mathcal{C}$ **do**
    Add $\text{Logit}_{\mathcal{M}}(y_t|c_i, x, y_{<t})$ to the Logit list
  end for
  if $\exists \mathcal{L} \geq T$ for $\mathcal{L}$ in Logit list **then**
    Select c_{max} with the highest $\mathcal{L}$ and c_{min} with the lowest $\mathcal{L}$
  else
    Select c_{max} with the highest α and c_{min} with the lowest α
  end if
  Contrast with c_{max} and c_{min}
  Sample y_t from the adjusted logits
  if y_t is eos_token **then**
    Break
  end if
end while

4 Experiments

We evaluate our method on multiple tasks that require models to reason based on context. We primarily focus on the following research questions:

- **RQ1**: Can our method consistently improve reasoning performance?
- **RQ2**: How do multiple contextual premises influence the reasoning process?
- **RQ3**: What is the relationship between the model's confidence and the downstream responses?

4.1 Experimental Setup

4.1.1 Datasets

We validate our approach on commonly used benchmark datasets for reasoning, including three arithmetic reasoning tasks: GSM8K (Cobbe et al., 2021), AQUA (Ling et al., 2017), and SVAMP (Patel et al., 2021), as well as three symbolic reasoning tasks: Coin Flip (Wei et al., 2022), BIG-bench Date Understanding, and BIG-bench Object Tracking (Srivastava et al., 2023). These datasets encompass a wide range of reasoning tasks, from simple to complex, and require leveraging contextual information rather than relying on the model's memorized knowledge. Notably, the information provided in the questions is not always helpful, and some problems even contain completely irrelevant distractors. The model must carefully discern the given premises while avoiding reasoning pitfalls.

To validate the anti-distraction effect of our method, we also conduct tests on GSM-IC (Shi et al., 2023). This dataset is based on GSM8K but introduces irrelevant premises to the original questions, thereby distracting the language model. For experimental efficiency, we randomly sample 100 questions from GSM-IC as the test subset.

Since our primary focus is on how to make better use of the problem premises, we do not choose tasks like commonsense reasoning or mathematical computation. These tasks mainly rely on the model activating its stored knowledge for reasoning, where context information is usually minimal or absent.

4.1.2 Baselines

We consider single-pass decoding methods as our baselines. Specifically, we compare with regular decoding, self-consistency (SC) (Wang et al., 2023), context-aware decoding (CAD) (Shi et al., 2024b), and Decoding by Contrasting Layers (DoLA) (Chuang et al., 2023). Among these, CAD and DoLA are both contrastive decoding-based methods. The former primarily contrasts scenarios with and without context, while the latter focuses on contrasting different layers of the model.

4.1.3 Language Models

To obtain the internal logits of the model, we apply our method to open-source large language models. We select Llama-2-7B-chat and Llama-2-13B-chat as the base models. Recently, strong reasoning models, particularly those from the DeepSeek series (Guo et al., 2025), have demonstrated exceptional performance. Therefore, we also aim to validate our method on such strong reasoning models. To maintain consistency with the aforementioned models, we choose DeepSeek-R1-Distill-Llama-

Models	Decoding	*Arithmetic* GSM8K	*Arithmetic* AQuA	*Arithmetic* SVAMP	*Symbolic* Coin	*Symbolic* Date	*Symbolic* Object	Avg.
Llama-2-7B-chat	Regular	21.68	24.01	41.90	47.00	39.29	30.80	34.11
	SC	**26.14**	21.65	47.19	52.80	40.37	32.53	36.78
	CAD	21.75	23.62	**49.90**	48.40	34.96	31.80	35.07
	DoLA	22.14	22.44	43.80	51.20	40.08	30.53	35.02
	Ours	25.47	**29.92**	47.59	**54.80**	**44.99**	**32.66**	**39.24**
Llama-2-13B-chat	Regular	34.49	15.74	49.40	47.40	46.07	27.33	36.84
	CAD	31.69	12.60	52.10	50.80	37.69	33.33	36.37
	Ours	**37.98**	**26.37**	**55.10**	**63.00**	**51.49**	**35.80**	**44.96**
DeepSeek-R1-Distill-Llama-8B	Regular	62.77	63.39	80.80	70.60	66.40	53.87	66.31
	CAD	65.80	50.79	77.80	66.20	68.29	76.67	67.59
	Ours	**77.01**	**65.35**	**85.30**	**82.00**	**74.53**	**90.25**	**79.07**

Table 1: Performance (%) comparison across different decoding methods. Our proposed C^3D consistently improves performance across various arithmetic and symbolic reasoning tasks. Moreover, the enhancement effect of our method is more pronounced on stronger base models, such as DeepSeek-R1-Distill-Llama-8B.

8B[1] as the representative model for our experiments. It is distilled from Llama-3.1-8B.

4.1.4 Implementation Details

Our method introduces two hyperparameters: k to control the top k logit values for confidence $\mathcal{L}$, and threshold T to adjust the reference between $\mathcal{L}$ and α. Specifically, we search for k within the range [1, 5, 10, 15, 20, 25] and T within the range [14, 16, 18, 20]. Since our method requires simultaneous decoding of multiple segments, we employ KV cache to enhance efficiency. For more details, please refer to Appendix A. We perform all experiments on a single 80GB A800 GPU.

4.2 Overall Performance (RQ1)

Table 1 presents the performance of different models across various reasoning tasks. We further categorize the observations into Llama-2 Model Observations and DeepSeek-Distill Model Observations based on the reasoning capabilities of the models.

4.2.1 Llama-2 Model Observations

On the Llama-2 series, our method consistently and significantly improves regular decoding performance. Particularly on the AQuA and Coin Flip datasets, the 7B and 13B models show the most substantial improvements. AQuA contains non-intuitive and complex mathematical problems, while Coin Flip requires multi-step state tracking. Both tasks demand the model to thoroughly understand the problem's meaning. Given that the comprehensive understanding capability of the Llama-2 series is not particularly strong, the original decoding is easily influenced by the context. Our strategy, however, better assists the model in grasping finer-grained information.

For similar contrastive decoding methods, such as CAD and DoLA, their performance across different datasets is inconsistent. This suggests that relying solely on full-context contrast or layer-wise contrast is insufficient to obtain evidence for token generation.

4.2.2 DeepSeek-Distill Model Observations

We further explore the performance of our method on stronger models. We observe that although DeepSeek-R1-Distill-Llama-8b already performs excellently on multiple tasks, our method can further enhance its reasoning performance. Specifically, we note improvements of 11.2% on GSM8K, 11.4% on Coin Flip, and 13.58% on Object Tracking. Such significant improvements indicate that strong reasoning models can better benefit from premises. We speculate that the reason is that weaker models can sometimes be overly confident even when incorrect (Fu et al., 2025), whereas strong reasoning models exhibit this behavior less frequently. Therefore, the latter can benefit more from the most confident premises.

[1]https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Llama-8B

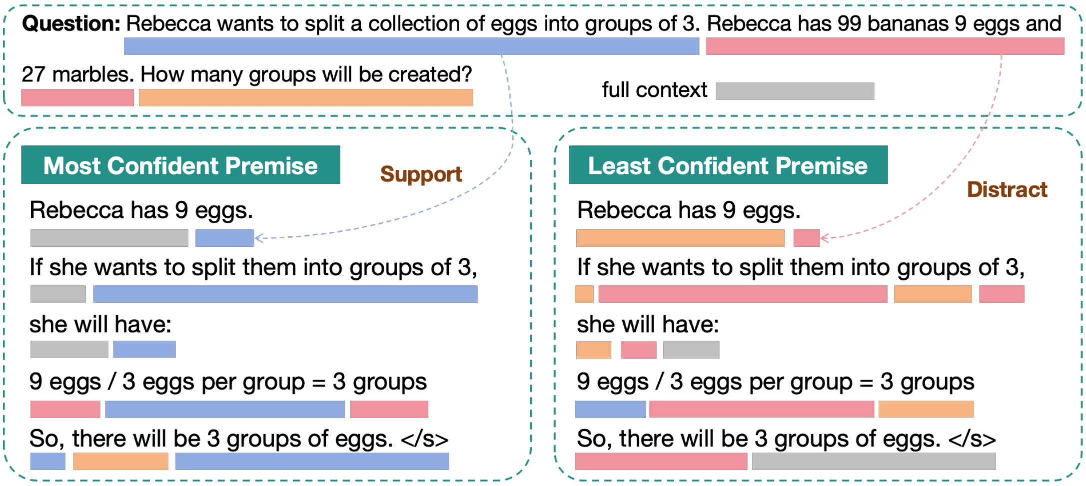

Figure 4: A visualized case study. Best viewed in color. The problem above is divided into three premises: we mark premise 1 in blue, premise 2 in pink, and premise 3 in orange, while the full context is marked in gray. The bottom left shows which premise supports each generated token (most confident), and the bottom right shows which premise distracts each generated token (least confident). The corresponding colors can help us better understand the reasoning process.

Decoding	7B	13B	DS
Regular w/o Irrelevant Context	49.0	68.0	94.0
Regular w/ IC	34.0	55.0	80.0
CAD w/ IC	36.0	54.0	75.0
Ours w/ IC	41.0	62.0	85.0

Table 2: The performance (%) on the GSM-IC subset. With the insertion of irrelevant context into the questions, the baseline methods show significant performance degradation. Our method, however, remains robust against such corruption.

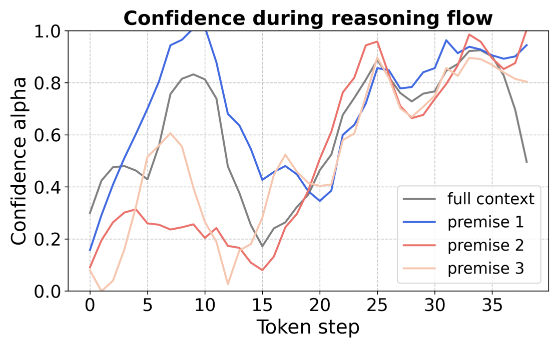

Figure 5: Visualization of how reasoning flows.

4.3 Performance on Data with Irrelevant Context (RQ1)

Table 2 presents the performance comparison on the GSM-IC subset. Since GSM-IC inserts an irrelevant premise into each question, this distracts the language model. We observe that the performance of baseline models significantly drops compared to scenarios without irrelevant context. In contrast, our method maintains comparable reasoning accuracy. This phenomenon demonstrates that our approach can effectively mitigate the negative impact of irrelevant context on the decoding process.

4.4 Case Study (RQ2)

To gain a deeper understanding of how LLMs utilize known premises during the reasoning process, we further perform a case study for illustration. Figure 4 shows the relationship between each premise in the problem and the downstream responses. We mark each premise with a distinct color and annotate the most confident and least confident premises for each generated token.

We observe that the beginning of each response tends to use the full context, while for specific information, the model favors those premises that most strongly support the reasoning, such as premise 1. Premise 3, which contains the least information, initially has the highest uncertainty. Similarly, the information in premise 2 distracts the model, resulting in a lower confidence.

4.5 How Reasoning Flows (RQ2)

Figure 5 further visualizes how the confidence α values of each premise change during token generation. This provides us with a clearer perspective on how the model drives the flow of reasoning. Specifically, premise 1 dominates the early stages

Decoding	GSM8K	AQuA
C^{3}D	25.47	29.92
- w/o $\mathcal{L}$	19.11	28.35
- w/o α	23.09	28.74
Regular	21.68	24.04

Table 3: Ablation studies on $\mathcal{L}$ and α.

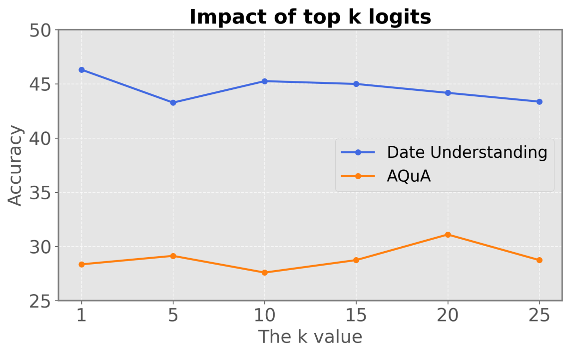

Figure 6: The trend of accuracy impact under different top-k values.

of generation. As reasoning information accumulates, premises with initially less information, such as premises 2 and 3, also gain insight and become part of the reasoning process. Eventually, by the end, each premise have gathered enough information and become confident.

4.6 Impact of Confidence $\mathcal{L}$ and α (RQ3)

We validate the contributions of the defined confidence measures $\mathcal{L}$ and α to reasoning. Table 3 presents the ablation studies on the GSM8K and AQuA datasets. The results show that both $\mathcal{L}$ and α have a positive effect on reasoning accuracy. Specifically, $\mathcal{L}$ has a slightly stronger impact on the model compared to α. As discussed in Section 3.1, the entropy-based α alone is insufficient to fully represent the model's confidence. When the accumulated logits fail to meet a certain threshold, the reliability of α also decreases. The introduction of $\mathcal{L}$ effectively compensates for this limitation.

4.7 Impact of Hyperparameter k and T (RQ3)

To further explore the impact of hyperparameter settings, we conduct additional experiments on the top-k logits and threshold T.

Figure 6 illustrates the trend of performance changes on AQUA and Date Understanding under different top-k logit values. As k increases from 1 to 25, AQUA shows an initial fluctuation

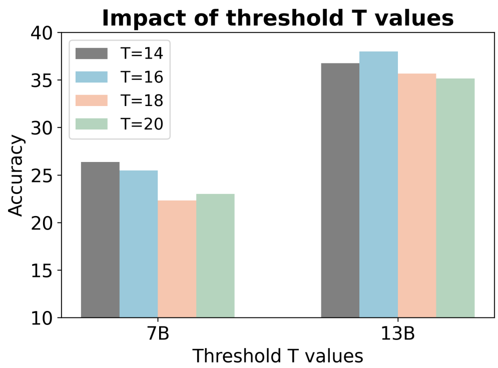

Figure 7: The performance of different T values on GSM8K across various models.

followed by an upward trend, while Date Understanding exhibits a gradual decline. This indicates that different datasets have varying preferences for top-k, which we speculate is related to the inherent properties of the datasets. Date Understanding primarily focuses on tokens related to dates, whereas AQUA requires a broader vocabulary space. However, overall, we can choose k=15 as a balanced compromise.

Since few studies discuss the impact of logit extremal values on responses, it is challenging to define a reasonable threshold. Ranging from 10 to 30, logit extremal values exhibit no clear pattern and are difficult to normalize. Therefore, we empirically select [14,16,18,20] as the experimental range. Figure 7 illustrates the effects of different thresholds T on GSM8K across two models. We observe that, despite changes in model size, the range of logits remains consistent. Additionally, their impact is relatively similar across models of different sizes. This phenomenon suggests that we can manually select a suitable threshold as a reference for the overall dataset.

5 Conclusion

We propose a confidence-guided cross-premise contrastive decoding method, which effectively mitigates reasoning errors in LLMs caused by contextual distractions. We validate the effectiveness of our method on both weak reasoning models and strong reasoning models (DeepSeek-R1-Distill-Llama-8B). Experiments show that our method achieves more significant improvements on strong reasoning models. Additionally, we visualize the role of each premise during the reasoning process, which can provide better guidance for future reasoning research.

Limitations

Considering both computational efficiency during decoding and potential biases introduced by additional information, we adopt only the most basic approach for premise decomposition in our current work. However, future research could explore more sophisticated methods for leveraging semantic information to obtain better contextual segments. Additionally, since LLMs inherently exhibit certain biases during inference due to their training, our work focuses on mitigating these biases through contrastive decoding. Future studies may further investigate how to obtain less biased output distributions from LLMs.

References

Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. 2024. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Chung-Ching Chang, David Reitter, Renat Aksitov, and Yun-Hsuan Sung. 2023. Kl-divergence guided temperature sampling. *arXiv preprint arXiv:2306.01286*.

Xinyun Chen, Ryan A Chi, Xuezhi Wang, and Denny Zhou. 2024. Premise order matters in reasoning with large language models. *arXiv preprint arXiv:2402.08939*.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. 2023. Dola: Decoding by contrasting layers improves factuality in large language models. *arXiv preprint arXiv:2309.03883*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2024. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, Bangkok, Thailand. Association for Computational Linguistics.

Tairan Fu, Javier Conde, Gonzalo Martínez, María Grandury, and Pedro Reviriego. 2025. Multiple choice questions: Reasoning makes large language models (llms) more self-confident even when they are wrong. *arXiv preprint arXiv:2501.09775*.

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

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. 2024. Language model cascades: Token-level uncertainty and beyond. *arXiv preprint arXiv:2404.10136*.

Shibo Hao, Sainbayar Sukhbaatar, D Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. 2024. Training large language models to reason in a continuous latent space, 2024. *URL https://arxiv.org/abs/2412.06769*.

Ke Ji, Jiahao Xu, Tian Liang, Qiuzhi Liu, Zhiwei He, Xingyu Chen, Xiaoyuan Liu, Zhijie Wang, Junying Chen, Benyou Wang, et al. 2025. The first few tokens are all you need: An efficient and effective unsupervised prefix fine-tuning method for reasoning models. *arXiv preprint arXiv:2503.02875*.

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023. Making language models better reasoners with step-aware verifier. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5315–5333, Toronto, Canada. Association for Computational Linguistics.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*.

Huan Ma, Jingdong Chen, Guangyu Wang, and Changqing Zhang. 2025. Estimating llm uncertainty with logits. *arXiv preprint arXiv:2502.00290*.

Nikolay Malkin, Zhen Wang, and Nebojsa Jojic. 2022. Coherence boosting: When your pretrained language model is not paying enough attention. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8214–8236, Dublin, Ireland. Association for Computational Linguistics.

Shentong Mo and Miao Xin. 2024. Tree of uncertain thoughts reasoning for large language models. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12742–12746. IEEE.

Sean O'Brien and Mike Lewis. 2023. Contrastive decoding improves reasoning in large language models. *arXiv preprint arXiv:2309.09117*.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, Online. Association for Computational Linguistics.

Ben Prystawski, Michael Li, and Noah Goodman. 2023. Why think step by step? reasoning emerges from the locality of experience. In *Advances in Neural Information Processing Systems*, volume 36, pages 70926–70947. Curran Associates, Inc.

Xin Qiu and Risto Miikkulainen. 2024. Semantic density: Uncertainty quantification for large language models through confidence measurement in semantic space. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zexuan Qiu, Zijing Ou, Bin Wu, Jingjing Li, Aiwei Liu, and Irwin King. 2024. Entropy-based decoding for retrieval-augmented large language models. *arXiv preprint arXiv:2406.17519*.

Leonardo Ranaldi and Andre Freitas. 2024. Aligning large and small language models via chain-of-thought reasoning. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1812–1827, St. Julian's, Malta. Association for Computational Linguistics.

Chufan Shi, Haoran Yang, Deng Cai, Zhisong Zhang, Yifan Wang, Yujiu Yang, and Wai Lam. 2024a. A thorough examination of decoding methods in the era of llms. *arXiv preprint arXiv:2402.06925*.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024b. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 783–791, Mexico City, Mexico. Association for Computational Linguistics.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Dennis Ulmer, Christian Hardmeier, and Jes Frellsen. 2023. Prior and posterior networks: A survey on evidential deep learning methods for uncertainty estimation. *arXiv preprint arXiv:2110.03051*.

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. *arXiv preprint arXiv:2203.11171*.

Xuezhi Wang and Denny Zhou. 2024. Chain-of-thought reasoning without prompting. In *Advances in Neural Information Processing Systems*, volume 37, pages 66383–66409. Curran Associates, Inc.

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.

Yuxi Xie, Anirudh Goyal, Wenyue Zheng, Min-Yen Kan, Timothy P Lillicrap, Kenji Kawaguchi, and Michael Shieh. 2024. Monte carlo tree search boosts reasoning via iterative preference learning. *arXiv preprint arXiv:2405.00451*.

Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, James Xu Zhao, Min-Yen Kan, Junxian He, and Michael Xie. 2023. Self-evaluation guided beam search for reasoning. In *Advances in Neural Information Processing Systems*, volume 36, pages 41618–41650. Curran Associates, Inc.

Xiaowei Yuan, Zhao Yang, Yequan Wang, Shengping Liu, Jun Zhao, and Kang Liu. 2024. Discerning and resolving knowledge conflicts through adaptive decoding with contextual information-entropy constraint. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3903–3922, Bangkok, Thailand. Association for Computational Linguistics.

Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. 2024a. Rest-mcts : Llm self-training via process reward guided tree search. In *Advances in Neural Information Processing Systems*, volume 37, pages 64735–64772. Curran Associates, Inc.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023. Enhancing uncertainty-based hallucination detection with stronger focus. *arXiv preprint arXiv:2311.13230*.

Yadong Zhang, Shaoguang Mao, Tao Ge, Xun Wang, Adrian de Wynter, Yan Xia, Wenshan Wu, Ting Song, Man Lan, and Furu Wei. 2024b. Llm as a mastermind: A survey of strategic reasoning with large language models. *arXiv preprint arXiv:2404.01230*.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

Huaixiu Steven Zheng, Swaroop Mishra, Xinyun Chen, Heng-Tze Cheng, Ed H Chi, Quoc V Le, and Denny Zhou. 2023. Take a step back: Evoking reasoning via abstraction in large language models. *arXiv preprint arXiv:2310.06117*.

A Decoding Efficiency

Since our method requires decoding multiple segments simultaneously, computational overhead is a potential concern. However, as our primary focus is enhancing the model's reasoning capability rather than computational efficiency, and given the abundance of existing research on accelerating LLMs, we adopt simple inference acceleration strategies to improve decoding efficiency. Specifically, we design a batch-based multi-segment decoding approach leveraging KV cache to store precomputed attention values. This design significantly reduces computational costs during decoding while allowing future integration of more advanced KV cache algorithms.

Additionally, inspired by Ji et al. (2025), we observe that the first few tokens during inference have greater impact on results compared to subsequent tokens — a phenomenon consistent with our findings in Section 4.5. Accordingly, we implement a simple truncation strategy: contrastive decoding is applied to the first few generated tokens, while original decoding is used for the remaining tokens.

Method / Tokens/s	7B	13B	DS
Regular Decoding	38.35	33.70	29.43
C^3D + KV Cache	35.76	31.70	25.99
C^3D + Truncation	31.70	25.43	25.34
C^3D + No Strategy	21.12	20.13	18.61

Table 4: The token throughput per second of different decoding strategies.

Table 4 demonstrates the decoding efficiency of different strategies. Using KV cache significantly improves decoding speed compared to not using it, achieving performance close to the original decoding efficiency while maintaining strong resistance to contextual distractions.

B Experiments on AIME

To evaluate our method's performance on more complex tasks, we conduct additional experiments using the AIME dataset. AIME (American Invitational Mathematics Examination) consists of a series of challenging mathematical problems widely used to assess the reasoning capabilities of LLMs. Table 5 presents our method's performance on AIME 2024[2] and AIME2025[3].

Decoding	AIME2024	AIME2025
Regular	46.67	26.67
C^3D	53.33	33.33

Table 5: DeepSeek-R1-Distill-Llama-8B performance comparison on AIME 2024 and AIME 2025 datasets.

C Case Study on GSM-IC

To further analyze how distraction affects the model's reasoning process, we compare the performance of different decoding methods on the GSM-IC dataset, as shown in Table 6. We observe that regular decoding is more susceptible to irrelevant information in the questions, leading to incorrect responses. In contrast, our method empowered by cross-premise contrastive decoding can selectively ignore such distracting information.

[2]https://huggingface.co/datasets/Maxwell-Jia/AIME_2024

[3]https://huggingface.co/datasets/opencompass/AIME2025

Regular	**C^{3}D**
Question: Gretchen draws caricatures in the park on the weekends. She charges \$20.00 per drawing. Ideally, Emma would love to sell 10000 drawings per weekend. If Gretchen sold 24 on Saturday and 16 on Sunday, how much money did she make?	**Question:** Gretchen draws caricatures in the park on the weekends. She charges \$20.00 per drawing. Ideally, Emma would love to sell 10000 drawings per weekend. If Gretchen sold 24 on Saturday and 16 on Sunday, how much money did she make?
Answer: Gretchen sells 24 drawings on Saturday and 16 drawings on Sunday. So, in total, she sells 24 + 16 = 40 drawings. Since each drawing costs \$20.00, Gretchen makes \$20.00 x 40 = \$800.00 per weekend. To find out how much money Emma would love to sell, we need to multiply the number of drawings Gretchen sold by the price per drawing: \$800.00 x 10000 = \$8,00,000.00. (Wrong)	**Answer:** Gretchen charges \$20.00 per drawing, so if she sells 24 drawings on Saturday, she will make \$20 x 24 = \$480.00. If she sells 16 drawings on Sunday, she will make \$20 x 16 = \$320.00. So, in total, Gretchen made \$480.00 + \$320.00 = \$800.00. (Correct)

Table 6: A comparative case study of different decoding methods on the GSM-IC dataset. Red text highlights distracting irrelevant information that negatively influences the model. The left panel demonstrates how standard decoding fails when exposed to such distractions, while the right panel shows our method's robust performance under the same conditions.