Reasoning in Flux: Enhancing Large Language Models Reasoning through Uncertainty-aware Adaptive Guidance

Zhangyue Yin[◇] **Qiushi Sun**[♡] **Qipeng Guo**[♣] **Zhiyuan Zeng**[◇] **Xiaonan Li**[◇]
Junqi Dai[◇] **Qinyuan Cheng**[◇] **Xuanjing Huang**[◇†] **Xipeng Qiu**[◇†]
[◇]School of Computer Science, Fudan University
[♡]The University of Hong Kong [♣]Shanghai AI Laboratory
{yinzy21,cengzy23,jqdai22,chengqy21}@m.fudan.edu.cn
qiushisun@u.nus.edu guoqipeng@pjlab.org.cn
{lixn20, xjhuang, xpqiu}@fudan.edu.cn

Abstract

Machine reasoning, which involves solving complex problems through step-by-step deduction and analysis, is a crucial indicator of the capabilities of Large Language Models (LLMs). However, as the complexity of tasks escalates, LLMs often encounter increasing errors in their multi-step reasoning process. This study delves into the underlying factors contributing to these reasoning errors and seeks to leverage uncertainty to refine them. Specifically, we introduce Uncertainty-aware Adaptive Guidance (UAG), a novel approach for guiding LLM reasoning onto an accurate and reliable trajectory. UAG first identifies and evaluates uncertainty signals within each step of the reasoning chain. Upon detecting a significant increase in uncertainty, UAG intervenes by retracting to a previously reliable state and then introduces certified reasoning clues for refinement. By dynamically adjusting the reasoning process, UAG offers a plug-and-play solution for improving LLMs' performance in complex reasoning. Extensive experiments across various reasoning tasks demonstrate that UAG not only enhances the reasoning abilities of LLMs but also consistently outperforms several strong baselines with minimal computational overhead. Further analysis reveals that UAG is notably effective in identifying and diminishing reasoning errors.

1 Introduction

The impressive advancements of Large Language Models (LLMs) have recently brought about a new era in machine reasoning (Brown et al., 2020; Chowdhery et al., 2022; OpenAI, 2023; Jiang et al., 2024; Sun et al., 2024c, *inter alia*). For challenging scenarios, decomposing a problem into a series of intermediate steps has been shown to significantly improve the performance of LLMs (Cobbe et al., 2021; Yu et al., 2023; Sun et al., 2024a).

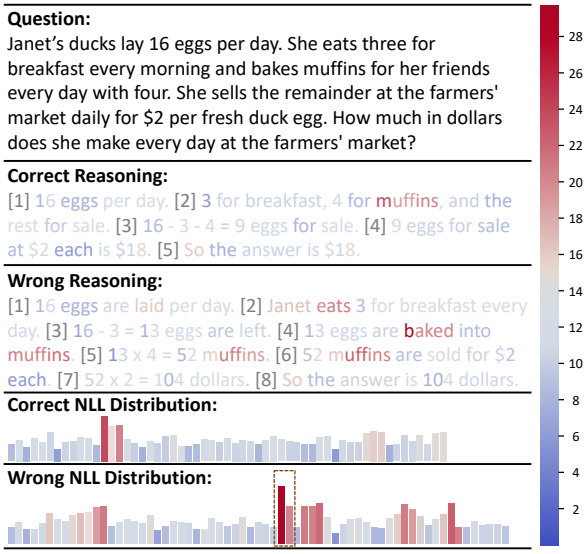


Figure 1: An example from the GSM8K dataset, where the model displays higher uncertainty in the tokens where the reasoning is incorrect (highlighted in red boxes). Each step is delineated by periods.

However, as the difficulty of tasks escalates, the reasoning chains inevitably become more complex and lengthy (Zhang et al., 2023b). This poses a challenge for LLMs in managing the accumulation of errors across multiple intermediate steps (Chen et al., 2022; Chu et al., 2023).

To mitigate the aforementioned issues, existing studies have focused on addressing the challenges from the perspective of alleviating uncertainty. For example, self-consistency decoding (Wang et al., 2023) samples multiple reasoning chains and employs a majority voting mechanism to mitigate the inherent randomness. Tree-of-thought (Yao et al., 2023) enables the exploration and evaluation of coherent thought units that serve as intermediate steps in problem-solving. Furthermore, Xie et al. (2023) conceptualize reasoning as a beam search, incorporating step-wise evaluation in the decoding process. While these methods have shown promise in enhancing reasoning performance, their manipulation of the reasoning process is conducted after the gen-

[†] Corresponding Authors

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2401–2416

eration of individual intermediate steps, *lacking the ability to make fine-grained, flexible adjustments with each step* to steer the model effectively. This oversight has long been neglected in the study of reasoning chains, yet it unveils a new pathway to improve the reasoning capabilities of LLMs.

Delving into individual reasoning steps, our observation reveals that the model can exhibit signs of uncertainty when faced with potential errors spontaneously. Figure 1 illustrates two distinct reasoning chains generated by LLaMA-2 (Touvron et al., 2023b), where a gradual shift towards red hues indicates an increase in the model's uncertainty. This increase in uncertainty is particularly notable when the model is in the midst of an incorrect reasoning step. For instance, the model erroneously interprets that 13 eggs were "baked" instead of "left" in one flaw, which leads to a cascade of subsequent errors.

Inspired by this, we propose a novel approach: UAG, which integrates clues throughout the reasoning process based on the model's fine-grained uncertainty. UAG first guides the model in an autoregressive manner to conduct reasoning. Upon detecting a significant rise in uncertainty within a step, the erroneous step will be removed, and targeted reasoning clues will be incorporated. Prior research (Cao, 2023; Diao et al., 2023) indicate that reasoning exemplars imbued with rich clues can substantially assist models in completing complex reasoning tasks. From a Bayesian perspective, we infer the necessity of selecting examples based on their relevance and originality. Furthermore, we cluster these example samples to minimize the search space and reduce computational costs. Our empirical studies on various datasets, including mathematical, commonsense, and symbolic reasoning, demonstrate that UAG significantly enhances model performance on complex reasoning tasks. Moreover, our method is plug-and-play, applicable to various open-source LLMs such as LLaMA (Touvron et al., 2023a) and Mistral (Jiang et al., 2023).

Our primary contributions are as follows:

- We conduct a pioneering study that explores the underlying causes of errors in LLM reasoning, focusing on the role of uncertainty.
- We introduce the Uncertainty-aware Adaptive Guidance (UAG), a novel technique that leverages model uncertainty to evaluate and enhance the reliability of each reasoning step.
- Our experimental evaluations, conducted across a diverse range of reasoning tasks, show that UAG significantly outperforms a series of strong baselines.

2 Related Work

2.1 Demonstration Guidance.

Recent advancements have emphasized the significance of effective exemplar selection to guide model reasoning (Zhang et al., 2022; Shum et al., 2023; Paranjape et al., 2023; Su et al., 2023). Pioneering this field, Zhang et al. (2023c) introduce AutoCoT, a method that automates the creation of exemplars by sampling a diverse array of problems and autonomously generating corresponding reasoning chains (Wei et al., 2022; Kojima et al., 2022). This approach eliminates the need for manually crafting task-specific examples. Additionally, some strategies leverage the intrinsic knowledge of LLMs to enhance the accuracy and factuality of reasoning processes through exemplar extraction (Wang et al., 2024).

In parallel, Diao et al. (2023) investigate the application of active learning for selecting informative exemplars, which utilizes the model's inherent uncertainties. Ye and Durrett (2023) focus on assessing the validity of reasoning chains within exemplars by evaluating their log-likelihood and performance accuracy on novel instances. Further extending the utility of LLMs in reasoning, Li and Qiu (2023) introduce memory-of-thought, a novel approach that retrieves pre-established, high-confidence thought processes to aid in current reasoning tasks. The potential for LLMs to utilize their self-generated reasoning chains for continuous self-improvement has been demonstrated in recent studies (Huang et al., 2023; Zheng et al., 2023; Lu et al., 2024; Madaan et al., 2023).

2.2 Decomposition and Validation.

Zhou et al. (2023) address the challenges that LLMs encounter in complex reasoning tasks by advocating for the decomposition of these tasks into simpler sub-questions. This concept aligns with the modular decomposition strategy introduced by Khot et al. (2023), which aims to optimize the handling of individual subtasks effectively. Building upon these foundations, Yao et al. (2023) innovate further by integrating a verification process within the decomposition framework. They conceptualize reasoning as a tree search, allowing LLMs to traverse various decision branches and perform both forward and backward exploration at each node.

Similarly, Besta et al. (2024) propose modeling the reasoning process as a graph structure, which offers a more dynamic framework for synthesizing LLM thoughts. Further advancing evaluation techniques, Yin et al. (2024) introduce a two-stage evaluation framework that incorporates local scoring and global evaluation to enhance LLM decision-making. Recent developments also include collaborative efforts, where problems are distributed among multiple LLMs for resolution (Yin et al., 2023a). Despite the innovative nature of these methods, they inherently increase computational demands, a consequence of the extensive decomposition (Hao et al., 2023; Zhang et al., 2023a; Han et al., 2023; Zhang et al., 2024), exploration (Liu et al., 2023), and verification (Sel et al., 2023).

2.3 Decoding Enhancement.

In the realm of decoding strategies, Wang et al. (2023) introduce self-consistency decoding, a significant shift from traditional greedy decoding. This method enhances reasoning capabilities by generating multiple reasoning paths and selecting the most consistent answer, effectively reducing the randomness associated with single-sample decoding. Complementing this, Fu et al. (2023b) advocate for guiding LLMs through more complex reasoning processes by using exemplars that feature increased reasoning complexity.

Furthering this line, Xie et al. (2023) propose a model akin to beam search that incorporates a self-evaluation mechanism to refine the decoding process. Building on this, Li et al. (2023a) introduce contrastive decoding, which helps LLMs avoid basic errors common in smaller models by utilizing model comparisons. This significantly enhances the reasoning capabilities of LLMs, as echoed in the work of O'Brien and Lewis (2023). Additionally, Chuang et al. (2023) explore intra-model dynamics by contrasting outputs from later versus earlier layers, aiming to cultivate more factually accurate reasoning within LLMs. Despite these advancements, these methodologies primarily rely on the internal representations within models (Sun et al., 2023; Stechly et al., 2023), often neglecting the integration of external reasoning cues.

3 Preliminary

This section outlines the foundational concepts that underpin our UAG method. Given a problem $\mathcal{Q}$, a LLM generates an answer $\mathcal{A}$, constructing it token by token through a probabilistic approach as described below:

$$P(\mathcal{A}|\mathcal{Q}) = \prod_{i=1}^{|\mathcal{A}|} P_{\mathcal{M}}(a_i|\mathcal{Q}, a_{<i}), \tag{1}$$

where $P_{\mathcal{M}}(a_i|\mathcal{Q}, a_{<i})$ represents the probability of generating the i-th token of the answer, given the problem $\mathcal{Q}$ and the sequence of previously generated tokens $a_{<i}$.

CoT (Wei et al., 2022) involves supplementing the problem $\mathcal{Q}$ with several demonstrations $\mathcal{D}$ that include detailed reasoning processes. This methodology guides the model to first generate the rationale $\mathcal{R}$, followed by the answer $\mathcal{A}$.

$$P(\mathcal{R}, \mathcal{A}|\mathcal{D}, \mathcal{Q}) = P(\mathcal{A}|\mathcal{D}, \mathcal{Q}, \mathcal{R})P(\mathcal{R}|\mathcal{D}, \mathcal{Q}), \tag{2}$$

Applying Bayesian theorem (Bayes and Price, 1763) allows us to further refine our understanding:

$$\begin{aligned} P(\mathcal{R}, \mathcal{A}|\mathcal{D}, \mathcal{Q}) &= \frac{P(\mathcal{D}|\mathcal{Q}, \mathcal{R}, \mathcal{A})P(\mathcal{R}, \mathcal{A}|\mathcal{Q})P(\mathcal{Q})}{P(\mathcal{D}, \mathcal{Q})} \\ &= \frac{P(\mathcal{D}|\mathcal{Q}, \mathcal{R}, \mathcal{A})P(\mathcal{R}, \mathcal{A}|\mathcal{Q})}{P(\mathcal{D}|\mathcal{Q})}, \end{aligned} \tag{3}$$

where a low $P(\mathcal{R}, \mathcal{A}|\mathcal{Q})$ indicates the model's difficulty in generating the desired rationale $\mathcal{R}$ and answer $\mathcal{A}$ without additional context. Our goal is to enhance the probability of accurately generating both the rationale and the answer by improving $P(\mathcal{R}, \mathcal{A}|\mathcal{D}, \mathcal{Q})$. According to Bayes' Theorem, this requires increasing $P(\mathcal{D}|\mathcal{Q}, \mathcal{R}, \mathcal{A})$ while decreasing $P(\mathcal{D}|\mathcal{Q})$.

Given that the answer $\mathcal{A}$ typically comprises fewer tokens than the rationale $\mathcal{R}$, our primary focus is on optimizing the impact of the reasoning process $\mathcal{R}$. To this end, we define the following criteria:

- **Relevance**: $P(\mathcal{D}|\mathcal{Q}, \mathcal{R}, \mathcal{A})$ indicates that the reasoning within $\mathcal{D}$ aligns closely with our expected reasoning process, emphasizing the relevance of the exemplified reasoning.
- **Originality**: $P(\mathcal{D}|\mathcal{Q})$ suggests that the reasoning within $\mathcal{D}$ introduces novel concepts or steps unknown to the model (Yin et al., 2023b), highlighting the originality of the exemplar's reasoning process.

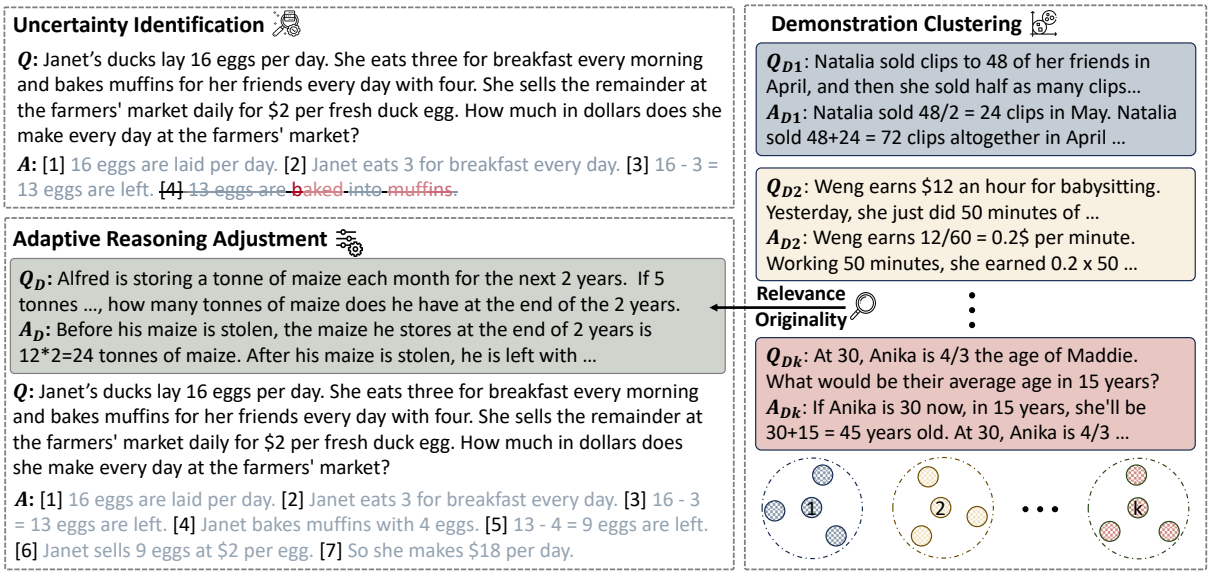


Figure 2: An overview of the Uncertainty-aware Adaptive Guidance (UAG) method. The process begins with the model incrementally generating reasoning based solely on the question while monitoring the uncertainty of each token. A significant increase in uncertainty signals potential errors, prompting a reversion to the last complete sentence. Exemplars are then selected from a curated set of examples based on their relevance and originality to assist in refining and completing the reasoning process.

4 Uncertainty-aware Adaptive Guidance

Uncertainty-aware Adaptive Guidance (UAG) aims to enhance the reasoning capabilities of LLMs by incorporating uncertainty awareness and adaptability. As depicted in Figure 2, our method progresses through three interconnected phases: Uncertainty Identification, Adaptive Reasoning Adjustment, and Demonstration Clustering. Each phase plays a crucial role in refining the LLM's reasoning process, which we detail in the subsequent sections.

4.1 Uncertainty Identification

In complex reasoning tasks, LLMs generate reasoning chains $\mathcal{R}$ sequentially, one token at a time. This generative process can be formally described as follows:

$$P(\mathcal{R}|\mathcal{Q}) = \prod_t P_{\mathcal{M}}(r_t|\mathcal{Q}, r_{<t}), \tag{4}$$

where $P(\mathcal{R}|\mathcal{Q})$ represents the probability of generating the reasoning chain $\mathcal{R}$ given a question $\mathcal{Q}$, and $P_{\mathcal{M}}(r_t|\mathcal{Q}, r_{<t})$ denotes the probability assigned by the model $\mathcal{M}$ to the t-th token, given the question and the preceding tokens $r_{<t}$.

A significant challenge in LLM reasoning is the unreliability and inaccuracy of generated reasoning chains, often resulting from cumulative errors in specific reasoning steps (Xie et al., 2023; Zhang et al., 2023b; Wang et al., 2024; Hao et al., 2023). The uncertainty during the decoding process typically reflects the model's confidence level or lack thereof (Manakul et al., 2023; Iter et al., 2023).

We define the uncertainty of generating the t-th token as:

$$\mathcal{H}(r_t) = -\log P(r_t|r_{<t}), \tag{5}$$

where $p(r_t|r_{<t})$ is the probability that the LLM assigns to the t-th token, given the previous tokens. To assess changes in uncertainty, we utilize the following difference function:

$$\Delta\mathcal{H}(r_t) = \mathcal{H}(r_t) - \mathcal{H}(r_{t-1}), \tag{6}$$

where $\Delta\mathcal{H}(r_t)$ quantifies the uncertainty gap for the t-th token relative to the $t-1$-th token. This metric is crucial for highlighting fluctuations in model confidence between decoding steps. Specifically, a positive $\Delta\mathcal{H}(r_t)$ indicates rising uncertainty, signaling challenging reasoning steps or potential mistakes. Conversely, a negative value indicates increasing reliability in the sequence.

If the increase in $\Delta\mathcal{H}(r_t)$ exceeds a predefined threshold θ, formally expressed as:

$$\text{if } \Delta\mathcal{H}(r_t) > \theta, \tag{7}$$

this condition suggests a significant rise in uncertainty, indicating a potential need for intervention, such as introducing additional context or implementing a corrective mechanism to steer the reasoning chain toward a more reliable trajectory.

4.2 Adaptive Reasoning Adjustment

When faced with increased uncertainty, UAG aims to rectify errors by eliminating erroneous reasoning steps and introducing supplementary reasoning clues, as aligned with the relevance and originality criteria defined in Section 3.

For refinement, we first backtrack within the reasoning chain to the last coherent step, denoted as r_m, where $r_{\leq m}$ represents the most recently completed reasoning step (as illustrated in Figure 2). To mitigate the uncertainty in the reasoning process, guided by Eq 3, our objective is to carefully select a demonstration to serve as external insights. We aim to increase $P(\mathcal{D}|\mathcal{Q}, \mathcal{R}, \mathcal{A})$ and decrease $P(\mathcal{D}|\mathcal{Q})$, where $\mathcal{D}$ comprises $\{\mathcal{Q}_d, \mathcal{R}_d, \mathcal{A}_d\}$.

Given that the model has not yet completed its reasoning, we utilize the existing reasoning process $r_{\leq m}$ to calculate the relevance score $\mathcal{S}_R$ as follows:

$$\begin{aligned}\mathcal{S}_R &= \log P(\mathcal{D}|\mathcal{Q}, r_{\leq m}) \\ &= \log P(\mathcal{Q}_d, \mathcal{R}_d, \mathcal{A}_d|\mathcal{Q}, r_{\leq m})\end{aligned} \tag{8}$$

A lower probability suggests higher originality; hence, we calculate the originality score through the negative log-likelihood of the probability:

$$\begin{aligned}\mathcal{S}_O &= -\log P(\mathcal{D}|\mathcal{Q}) \\ &= -\log P(\mathcal{Q}_d, \mathcal{R}_d, \mathcal{A}_d|\mathcal{Q})\end{aligned} \tag{9}$$

The selection score $\mathcal{S}$ is then computed as the weighted average of the two scores:

$$\mathcal{S} = \lambda_1 \mathcal{S}_R + \lambda_2 \mathcal{S}_O, \tag{10}$$

where λ_1 and λ_2 are the weights assigned to the relevance and originality scores, respectively.

Following this, we rank the demonstrations according to the selection score $\mathcal{S}$, ordered from highest to lowest as $\{\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3, \dots\}$, corresponding to $\{\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3, \dots\}$. We retain the initial reasoning process $r_{\leq m}$ and sequentially append each $\mathcal{D}_i$ in front of the problem $\mathcal{Q}$. Our objective is to identify a $\mathcal{D}_i$ such that, upon generating up to the next step, the uncertainty remains consistently below a predefined threshold θ:

$$\exists \mathcal{D}_i : \forall k, \Delta\mathcal{H}(r_{m+k}) \leq \theta, \tag{11}$$

where k is the index of the tokens generated after appending $\mathcal{D}_i$. This ensures that the enhanced reasoning process effectively mitigates uncertainty and increases the reliability of the reasoning chain.

4.3 Optimizing Demonstration Selection Through Clustering

Given the reasoning process for each question $\mathcal{Q}$, it becomes necessary to retrieve demonstrations $\mathcal{D}$ when the uncertainty exceeds the predefined threshold θ. This could lead to a high computational overhead due to the extensive retrieval of demonstrations. To address this, we propose a clustering approach for organizing the demonstration set $\{\mathcal{D}\}$, aimed at selecting representative exemplars efficiently.

Initially, we compute a vector representation for each demonstration $\mathcal{D}_i$ using a sophisticated text embedding model, `text-embedding-3-large`. Subsequently, these vectorized demonstrations are grouped into k distinct clusters using the k-means clustering algorithm. This method efficiently groups similar reasoning processes, enhancing the efficiency of demonstration selection.

We structure each cluster $\mathcal{C}_j$ by sorting its demonstrations according to their proximity to the cluster's centroid, prioritizing those closest as they are most representative of the cluster's reasoning pattern:

$$\mathcal{C}_j = [\mathcal{D}_1^j, \mathcal{D}_2^j, \dots], \quad \mathcal{C}_j \in K\text{-}Means(\{\mathcal{D}\}), \tag{12}$$

where each cluster $\mathcal{C}_j$ comprises demonstrations $[\mathcal{D}_1^j, \mathcal{D}_2^j, \dots]$ that exhibit similar reasoning traits, determined by the K-means clustering of the demonstration set $\{\mathcal{D}\}$.

This clustering-based approach not only streamlines the demonstration selection process but also ensures that the model has access to a diverse yet concise set of reasoning examples. It effectively balances the need for a broad variety of reasoning examples with computational efficiency, ensuring that the reasoning process is both accurate and practical.

5 Experiments

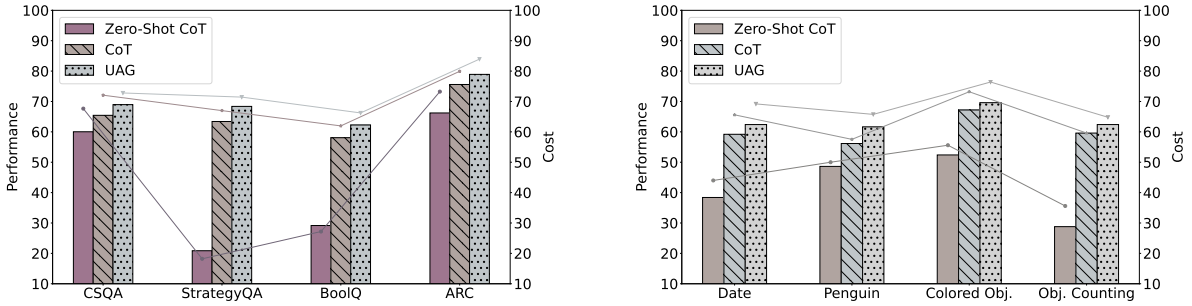
5.1 Experimental Setup

In this section, we delineate and scrutinize the performance of our proposed UAG, utilizing a variety of LLM backbones across a series of reasoning benchmarks. To evaluate the efficacy of UAG, we primarily employ accuracy as the performance metric. Furthermore, we undertake an analysis of

https://openai.com/index/new-embedding-models-and-api-updates

	GSM8K	MultiArith	SingleEQ	AddSub	SVAMP	AQuA	Average	Avg. #Tokens
Single Reasoning Chain								
ZS-CoT	41.93	64.50	71.65	75.19	59.40	31.89	57.43	178.25
CoT	38.89	75.50	77.36	75.19	59.20	30.71	59.48	87.88
ComplexCoT	43.67	76.50	77.56	76.20	56.90	31.50	60.38	124.96
UAG	**46.70**	**77.66**	**79.92**	**77.97**	**60.70**	**33.85**	**62.80**	151.76
Multiple Reasoning Chains								
ZS-CoT-SC	52.16	75.33	77.17	80.25	67.40	37.01	64.89	884.79
CoT-SC	47.08	85.00	85.43	79.24	67.40	36.22	66.73	442.23
ComplexCoT-SC	56.63	85.83	83.86	79.49	**67.90**	35.83	68.26	629.18
UAG-SC	**58.07**	**87.66**	**86.41**	**81.26**	67.60	**39.76**	**70.12**	772.53

Table 1: Results on Arithmetic Reasoning Tasks (Accuracy in %). The best result is highlighted in **bold**, while the method with the lowest computational cost is denoted in green. We utilize the Mistral-7B (Jiang et al., 2023) backbone for all methods to ensure a fair comparison. For brevity, Zero-Shot-CoT is abbreviated as ZS-CoT. In multiple reasoning chains scenario, the self-consistency is applied to determine the final outcome. Additionally, we report the average number of generated tokens (#Tokens) to compare the computational efficiency of each method.



(a) Commonsense Reasoning

(b) Symbolic Reasoning

Figure 3: Performance comparison and cost curves on (a) commonsense reasoning and (b) symbolic reasoning. The cost and performance of different methods correspond to each other. Histograms show the accuracy, while line charts illustrate the cost.

the computational costs among different methods, quantified by the number of tokens generated during the reasoning process. Implementation details and configurations of the various approaches can be found in Appendix B.

Benchmarks. Our evaluation encompasses three distinct categories of reasoning tasks, ensuring a comprehensive analysis of UAG's versatility and effectiveness:

- *Arithmetic Reasoning:* This category includes GSM8K (Cobbe et al., 2021), MultiArith (Roy and Roth, 2015), SingleEq (Koncel-Kedziorski et al., 2016), AddSub (Hosseini et al., 2014), SVAMP (Patel et al., 2021), and AQUA (Ling et al., 2017), which encompass a range of arithmetic problem-solving tasks.
- *Commonsense Reasoning:* We employ StrategyQA (Geva et al., 2021), CommonsenseQA (CSQA; Talmor et al., 2019), BoolQ (Clark et al., 2019), and AI2 Reasoning Challenge (ARC-c; Clark et al., 2018) to gauge the model's ability to understand and apply commonsense knowledge.
- *Symbolic Reasoning:* This involves datasets derived from BigBench (bench authors, 2023; Suzgun et al., 2023), specifically Date Understanding, Penguins in a Table, Colored Objects, and Object Counting, which test the model's skill in abstract and symbolic thought processes.

Following Zhang et al. (2023c), our experiments are conducted in a "test question only" scenario, where we lack access to correct answers and must independently construct exemplars. Detailed descriptions and statistics of these benchmarks are provided in the appendix A.

Baselines. For an intuitive and comprehensive performance comparison, we incorporate three primary categories of baselines: (1) Zero-shot CoT (Kojima et al., 2022) for reasoning without exemplars; (2) CoT (Wei et al., 2022) for exemplar-guided chain-of-thought prompting; and (3) Com-

plexCoT (Fu et al., 2023b) for complexity-based prompting. Furthermore, we also employ self-consistency decoding (Wang et al., 2023) as a strong baseline with multiple reasoning chains for comparison. In our analysis, UAG's performance is contrasted not only with these baselines but also with a variety of exemplar selection (Zhang et al., 2023c) and decoding enhancement techniques (O'Brien and Lewis, 2023; Chuang et al., 2023) in Appendix C.4. For generation, we adhere to the few-shot exemplars of baselines and use the number of generated tokens as a metric to assess the computational cost of each method.

Backbones. We derive embeddings for clustering through `text-embedding-3-large`. In our evaluation, we utilize open-source models LLaMA-2 (Touvron et al., 2023b) and Mistral (Jiang et al., 2023), applying various prompting techniques such as Zero-Shot CoT (ZS-CoT) (Kojima et al., 2022), CoT (Wei et al., 2022), and ComplexCoT (Fu et al., 2023b). Section 5.3 details our examination of the scalability of our approach across various model sizes, specifically using the 7B, 13B, and 70B parameter configurations of LLaMA-2. Additionally, we integrate a Mixture of Experts model, Mistral-8x7B (Jiang et al., 2024) to further diversify our evaluations.

5.2 Main Results

Arithmetic Reasoning. In Table 1, we present the results of arithmetic reasoning tasks. Our method manifests substantial performance enhancement across most benchmarks, in both single and multiple chain scenarios. Notably, we observe absolute increments in accuracy of 7.81%, 2.78%, and 3.14% on GSM8K, AddSub, and AQuA benchmarks, respectively, when compared to CoT approach (Wei et al., 2022). This disparity in performance gains can be attributed to UAG's strategy of constraining the reasoning search space by strategically eliminating uncertainty, as reflected in its superior performance across tasks. This underscores the efficacy of introducing controllable randomness in the UAG decoding process to expand the search space. A more extensive comparison of UAG against a broader spectrum of methods is detailed in Appendix C, where we delve deeper into the underlying factors contributing to UAG's enhanced performance.

Commonsense and Symbolic Reasoning. As illustrated in Figure 3a and Figure 3b, UAG consistently outperforms existing approaches across a variety of tasks. For instance, on the BoolQ dataset, our approach achieves an accuracy rate of 62.26%, significantly surpassing the 58.07% accuracy of the baseline method. Particularly noteworthy is our performance on the StrategyQA and BoolQ datasets, where we observe that the Zero-Shot CoT (Kojima et al., 2022) struggled in guiding the model to generate precise and accurate answers, leading to significant performance declines.

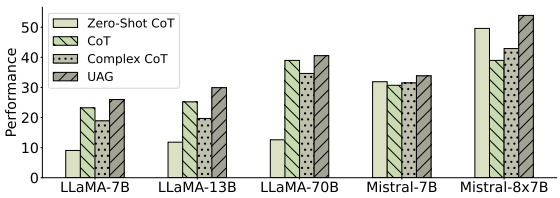


Figure 4: Performance comparison across models on the AQuA dataset. UAG achieves consistent performance improvement across various models.

Computational Cost. Despite the significant improvements our approach yields across various benchmarks, a potential concern arises regarding the computational overhead. Upon examining Figures 3a and 3b, we note that UAG, when applied to commonsense and symbolic reasoning tasks, does not incur substantial performance costs in comparison to CoT. For instance, in the Date Understanding dataset, UAG demonstrates only a 20% increase in overhead relative to CoT, yet it achieves an enhancement of over 3% in performance. Furthermore, as illustrated in Table 1, in arithmetic reasoning, UAG's computational demand is comparable to ComplexCoT and is even more efficient than the exemplar-free Zero-Shot CoT approach. Notably, UAG's mechanism of refining reasoning based on finished reasoning steps ensures that the additional computational overhead is minimal.

5.3 Further Analysis

Performance on various models. In Figure 4, we compare the performance of UAG against other baselines across different LLMs. UAG consistently outperforms these baselines. Intriguingly, CoT does not always outperform Zero-Shot-CoT (Kojima et al., 2022), e.g., Zero-Shot-CoT outperforms CoT on Mistral-7B, and this advantage is more pronounced on Mistral-8x7B. Furthermore, an increased number of exemplars does not unconditionally enhance performance; in certain cases, such as with Mistral, better results are achieved even with-

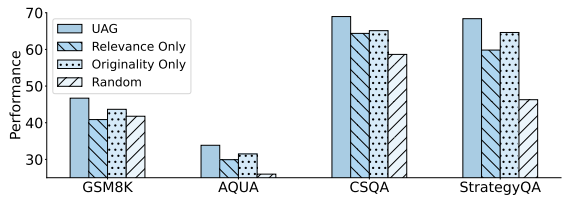


Figure 5: Ablation Study on Relevance and Originality. Performance impact across datasets when omitting either relevance or originality.

out exemplars (Li et al., 2023b). In this scenario, Mistral is even better without exemplar. This phenomenon underscores UAG's strategic advantage in introducing the most appropriate exemplars on-demand, which substantially contributes to the observed performance improvements of our method.

Importance of Relevance and Originality. In Section 3, we introduce the concepts of relevance and originality, with exemplar selection criteria detailed in Eq 10. An ablation study, as depicted in Figure 5, evaluates the impact of these factors. We modify the experiment by omitting the condition in Eq 11 and employing random selection for comparison. The results reveal that excluding originality leads to an average performance drop of 5.73%, as the model then relies solely on question relevance, often being misled by errors in analogous questions (Zhang et al., 2023c). Conversely, eliminating relevance also results in significant performance decline, rendering the exemplar selection process ineffective. A further analysis of the weights λ_1 and λ_2 is elaborated in Appendix C.1.

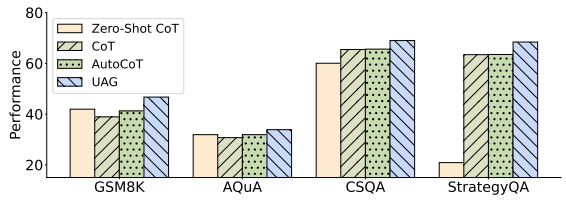


Figure 6: Performance comparison of UAG and Auto-CoT methods on various reasoning datasets.

Comparison to Existing Demonstration Selection Method. In Figure 6, we conduct an analysis of our UAG method against another representative demonstration selection strategy utilizing clustering, Auto-CoT (Zhang et al., 2023c), across various reasoning datasets. This comparison, under the same experimental setting, reveals a notable performance enhancement by UAG over Auto-CoT in each dataset. These findings indicate the superiority of UAG, which dynamically selects appropriate demonstrations based on uncertainty during the reasoning process, as opposed to Auto-CoT's static pre-selection approach.

6 Conclusion

In this paper, we start by observing the causes behind LLM reasoning errors from the perspective of uncertainty, paving the way for nuanced and adaptive modifications in the reasoning process. Driven by this insight, we introduce the Uncertainty-aware Adaptive Guidance (UAG). UAG strategically mitigates reasoning errors by identifying and eliminating uncertainties with reasoning steps, concurrently leveraging exemplars chosen for their relevance and originality to dispel uncertainty and steer the subsequent reasoning trajectory. Comprehensive experimental evaluations across a spectrum of reasoning tasks demonstrate that UAG not only surpasses several strong baselines but also exhibits adaptability to a diverse array of models, underscoring its efficacy and wide applicability in enhancing LLM reasoning capabilities. Further analysis showcases the remarkable versatility of UAG, which can function as a plug-and-play module and efficiently identify and eliminate reasoning errors.

Ethics Statement

The development and deployment of Large Language Models (LLMs), such as those described in this paper, necessitate careful consideration of various ethical concerns. We outline several key areas of focus:

Data Privacy and Security. Our approach, involving the enhancement of LLMs for reasoning tasks, does not require the collection or processing of personal data. The prompts and methods used are devoid of personal information, aligning with privacy preservation principles.

Impact on Workforce. The automation capabilities of LLMs might affect employment in certain sectors. It is important to consider the broader societal implications and support the workforce in adapting to these technological changes.

Environmental Considerations. The computational resources required for training and running LLMs have environmental impacts. We advocate for the use of sustainable practices and the exploration of energy-efficient models.

In conducting this research, we have adhered to ethical guidelines and ensured compliance with

the licensing requirements of the datasets used, as detailed in Table 2. Our commitment to ethical research extends beyond legal compliance, encompassing a broader responsibility to the societal implications of our work.

Limitations

While our Uncertainty-aware Adaptive Guidance (UAG) method demonstrates significant improvements in reasoning tasks, it is important to acknowledge certain limitations that point towards areas for future development:

Applicability to Closed-Source Models. A notable constraint of our method is its limited applicability to closed-source models. Models such as ChatGPT and Claude, which do not provide access to the probability distribution of tokens, pose a challenge to the implementation of UAG. This limitation restricts the versatility of our approach, as it cannot be directly applied to these commercially closed-source models.

Generalization to Broader Generative Tasks. While our research has focused predominantly on reasoning tasks, the potential of leveraging uncertainty in LLMs extends to a wider spectrum of generative applications (Manakul et al., 2023), such as code generation (Sun et al., 2024b). This includes using uncertainty as a metric for evaluating model hallucination (Ji et al., 2023), generation quality, and other aspects of generative performance. However, our current scope has not encompassed these areas, and we recognize this as an opportunity for future research to expand the applicability and utility of UAG in these domains.

Acknowledgement

We extend our gratitude to the members of the FudanNLP group for their insightful suggestions and thought-provoking discussions that greatly enhanced this work. We also sincerely appreciate the anonymous reviewers and area chairs for their constructive feedback, which was instrumental in advancing the quality of our study. This work was supported by the National Natural Science Foundation of China (No. 62236004). The computations in this research were performed using the CFFF platform of Fudan University.

References

Mr. Bayes and Mr. Price. 1763. An essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, f. r. s. communicated by mr. price, in a letter to john canton, a. m. f. r. s. *Philosophical Transactions (1683-1775)*, 53:370–418.

BIG bench authors. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2024. Graph of thoughts: Solving elaborate problems with large language models.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *NeurIPS*, volume 33, pages 1877–1901.

Lang Cao. 2023. Enhancing reasoning capabilities of large language models: A graph-based verification approach.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2023. A survey of chain of thought reasoning: Advances, frontiers and future.

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.

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. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.

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.

Shizhe Diao, Pengcheng Wang, Yong Lin, and Tong Zhang. 2023. Active prompting with chain-of-thought for large language models.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023a. Gptscore: Evaluate as you desire.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2023b. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations*.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Chengcheng Han, Xiaowei Du, Che Zhang, Yixin Lian, Xiang Li, Ming Gao, and Baoyuan Wang. 2023. DialCoT meets PPO: Decomposing and exploring reasoning paths in smaller language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8055–8068, Singapore. Association for Computational Linguistics.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen Wang, Daisy Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8154–8173, Singapore. Association for Computational Linguistics.

Mohammad Javad Hosseini, Hannaneh Hajishirzi, Oren Etzioni, and Nate Kushman. 2014. Learning to solve arithmetic word problems with verb categorization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 523–533. Association for Computational Linguistics.

Jiaxin Huang, Shixiang Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2023. Large language models can self-improve. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1051–1068, Singapore. Association for Computational Linguistics.

Dan Iter, Reid Pryzant, Ruochen Xu, Shuohang Wang, Yang Liu, Yichong Xu, and Chenguang Zhu. 2023. In-context demonstration selection with cross entropy difference. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1150–1162, Singapore. Association for Computational Linguistics.

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

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2023. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations*.

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*.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. 2016. MAWPS: A math word problem repository. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1157, San Diego, California. Association for Computational Linguistics.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023a. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.

Xiaonan Li and Xipeng Qiu. 2023. MoT: Memory-of-thought enables ChatGPT to self-improve. In *Proceedings of the 2023 Conference on Empirical Meth-*

ods in Natural Language Processing, pages 6354–6374, Singapore. Association for Computational Linguistics.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023b. 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.

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 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 158–167. Association for Computational Linguistics.

Tengxiao Liu, Qipeng Guo, Yuqing Yang, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023. Plan, verify and switch: Integrated reasoning with diverse X-of-thoughts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2807–2822, Singapore. Association for Computational Linguistics.

Jianqiao Lu, Wanjun Zhong, Wenyong Huang, Yufei Wang, Qi Zhu, Fei Mi, Baojun Wang, Weichao Wang, Xingshan Zeng, Lifeng Shang, Xin Jiang, and Qun Liu. 2024. Self: Self-evolution with language feedback.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46594. Curran Associates, Inc.

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

Sean O'Brien and Mike Lewis. 2023. Contrastive decoding improves reasoning in large language models.

OpenAI. 2023. GPT-4 technical report.

Bhargavi Paranjape, Scott Lundberg, Sameer Singh, Hannaneh Hajishirzi, Luke Zettlemoyer, and Marco Tulio Ribeiro. 2023. Art: Automatic multi-step reasoning and tool-use for large language models.

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.

Subhro Roy and Dan Roth. 2015. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1743–1752. The Association for Computational Linguistics.

Bilgehan Sel, Ahmad Al-Tawaha, Vanshaj Khattar, Ruoxi Jia, and Ming Jin. 2023. Algorithm of thoughts: Enhancing exploration of ideas in large language models.

KaShun Shum, Shizhe Diao, and Tong Zhang. 2023. Automatic prompt augmentation and selection with chain-of-thought from labeled data.

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

Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2023. Selective annotation makes language models better few-shot learners. In *The Eleventh International Conference on Learning Representations*.

Jiankai Sun, Chuanyang Zheng, Enze Xie, Zhengying Liu, Ruihang Chu, Jianing Qiu, Jiaqi Xu, Mingyu Ding, Hongyang Li, Mengzhe Geng, Yue Wu, Wenhai Wang, Junsong Chen, Zhangyue Yin, Xiaozhe Ren, Jie Fu, Junxian He, Wu Yuan, Qi Liu, Xihui Liu, Yu Li, Hao Dong, Yu Cheng, Ming Zhang, Pheng Ann Heng, Jifeng Dai, Ping Luo, Jingdong Wang, Ji-Rong Wen, Xipeng Qiu, Yike Guo, Hui Xiong, Qun Liu, and Zhenguo Li. 2024a. A survey of reasoning with foundation models.

Qiushi Sun, Zhirui Chen, Fangzhi Xu, Kanzhi Cheng, Chang Ma, Zhangyue Yin, Jianing Wang, Chengcheng Han, Renyu Zhu, Shuai Yuan, Qipeng Guo, Xipeng Qiu, Pengcheng Yin, Xiaoli Li, Fei Yuan, Lingpeng Kong, Xiang Li, and Zhiyong Wu. 2024b. A survey of neural code intelligence: Paradigms, advances and beyond.

Qiushi Sun, Zhangyue Yin, Xiang Li, Zhiyong Wu, Xipeng Qiu, and Lingpeng Kong. 2023. Corex: Pushing the boundaries of complex reasoning through multi-model collaboration.

Tianxiang Sun, Xiaotian Zhang, Zhengfu He, Peng Li, Qinyuan Cheng, Xiangyang Liu, Hang Yan, Yunfan Shao, Qiong Tang, Shiduo Zhang, et al. 2024c. Moss: An open conversational large language model. *Machine Intelligence Research*, pages 1–18.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung,

Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

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

Jianing Wang, Qiushi Sun, Xiang Li, and Ming Gao. 2024. Boosting language models reasoning with chain-of-knowledge prompting.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, Xu Zhao, Min-Yen Kan, Junxian He, and Qizhe Xie. 2023. Self-evaluation guided beam search for reasoning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of Thoughts: Deliberate problem solving with large language models.

Xi Ye and Greg Durrett. 2023. Explanation selection using unlabeled data for chain-of-thought prompting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 619–637, Singapore. Association for Computational Linguistics.

Zhangyue Yin, Qiushi Sun, Cheng Chang, Qipeng Guo, Junqi Dai, Xuanjing Huang, and Xipeng Qiu. 2023a. Exchange-of-thought: Enhancing large language model capabilities through cross-model communication. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15135–15153, Singapore. Association for Computational Linguistics.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023b. Do large language models know what they don't know? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8653–8665, Toronto, Canada. Association for Computational Linguistics.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Zhiyuan Zeng, Xiaonan Li, Tianxiang Sun, Cheng Chang, Qinyuan Cheng, Ding Wang, Xiaofeng Mou, Xipeng Qiu, and Xuanjing Huang. 2024. Aggregation of reasoning: A hierarchical framework for enhancing answer selection in large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 609–625, Torino, Italia. ELRA and ICCL.

Fei Yu, Hongbo Zhang, Prayag Tiwari, and Benyou Wang. 2023. Natural language reasoning, a survey.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. BARTScore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems*.

Jiajie Zhang, Shulin Cao, Tingjian Zhang, Xin Lv, Juanzi Li, Lei Hou, Jiaxin Shi, and Qi Tian. 2023a. Reasoning over hierarchical question decomposition tree for explainable question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14556–14570, Toronto, Canada. Association for Computational Linguistics.

Kun Zhang, Jiali Zeng, Fandong Meng, Yuanzhuo Wang, Shiqi Sun, Long Bai, Huawei Shen, and Jie Zhou. 2024. Tree-of-reasoning question decomposition for complex question answering with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19560–19568.

Yifan Zhang, Jingqin Yang, Yang Yuan, and Andrew Chi-Chih Yao. 2023b. Cumulative reasoning with large language models.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2023c. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations*.

Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, and Yu Li. 2023. Progressive-hint prompting improves reasoning in large language models. *ArXiv preprint*, abs/2304.09797.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.

A Statistics and Details of Datasets

For our experimental analysis, we carefully selected a diverse set of 14 datasets, encompassing the domains of arithmetic reasoning, commonsense reasoning, and symbolic reasoning. In Table 2, we comprehensively detail each dataset, including its source, type of answers it contains, the count of Chain-of-Thought (CoT) (Wei et al., 2022) prompt exemplars used, the size of the test sample, and the applicable licenses.

B Implementation Details

Baseline Implementation. In our main experiments, we employ three baselines: Zero-Shot CoT (Kojima et al., 2022), CoT (Wei et al., 2022), and ComplexCoT (Fu et al., 2023b). For Zero-Shot CoT, we append "Let's think step by step" after each question. For both CoT and Complex-CoT, we adhere to their original prompting exemplars. To maintain consistency, we standardize the prompt format in ComplexCoT to match CoT, replacing "Question:" and "Answer:" with "Q:" and "A:", respectively. The details of prompts used are listed in Table 2. While CoT and ComplexCoT have the same number of prompts, each example in ComplexCoT encompasses more intermediate steps. The experimental outcomes for CD and DoLA were sourced from Chuang et al. (2023) and O'Brien and Lewis (2023). For comparison, we use the LLaMA-1 model (Touvron et al., 2023a) and sample 20 reasoning chains for self-consistency (Wang et al., 2023), aligning with their settings.

Generation Settings In our experimental setup, we adapt the generation temperature τ for different tasks and baseline models. For the LLaMA model, Zero-Shot-CoT shows optimal performance at lower temperatures, specifically within $\tau \in [0.1, 0.5]$. This contrasts with CoT and Complex-CoT, which perform better at higher temperatures

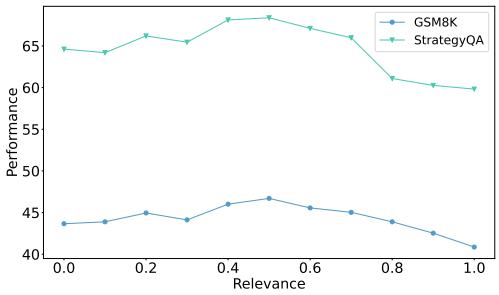


Figure 7: Weight analysis of Relevance and Originality on GSM8K and StrategyQA dataset.

($\tau > 0.5$), consistent with Xie et al. (2023). For the Mistral model, a temperature control within $\tau \in [0.3, 0.7]$ yields more favorable results. This variance in optimal temperature settings can be attributed to differences in model architecture and the distribution of the training data.

In scenarios involving multiple reasoning paths, we set the sampling temperature to 0.7 and set the number of reasoning chains to 5 to explore the reasoning space thoroughly. A majority voting mechanism is employed post-answer generation for the final decision (Wang et al., 2023). We note that some special tokens, such as the "muffins" in Figure 1, exhibits high uncertainty, likely due to infrequent usage, but did not consider these as reasoning errors if their uncertainty exceeded the threshold θ.

For different datasets, we follow the Auto-CoT setup (Zhang et al., 2023c) with varying cluster numbers; for mathematical reasoning, $k = 8$, while for symbolic reasoning, $k = 4$ was uniform. In commonsense reasoning, the cluster count was set to 7 for CSQA and StrategyQA, and 5 for BoolQ and ARC-c. To ensure precise answer extraction and mitigate evaluation errors, as discussed in Section 5.3, an exemplar was incorporated in the initial stages for Mistral-7B, LLaMA models.

Hardware utilization included a single RTX4090 for running LLaMA-7B and Mistral-7B models, two RTX4090s for LLaMA-13B, and two A100s for LLaMA-70B and Mistral-8x7B. In the multiple reasoning chains scenario, employing multiple-sampling effectively reduces the randomness associated with a single run. The UAG method is implemented using PyTorch and Transformers, with Copilot and ChatGPT assisting in code writing and debugging.

Hyper parameter. We set the respective weights of relevance and originality λ_1 and λ_2 to 0.5 and 0.5, respectively, and the ablation experiments for

Dataset	Reasoning Task	Answer Format	# EX.	# EVAL.	License
GSM8K (Cobbe et al., 2021)	Arithmetic	Number	8	1,319	MIT License
MultiArith (Roy and Roth, 2015)	Arithmetic	Number	8	600	Unspecified
SingleEq (Koncel-Kedziorski et al., 2016)	Arithmetic	Number	8	508	Unspecified
AddSub (Hosseini et al., 2014)	Arithmetic	Number	8	395	Unspecified
SVAMP (Patel et al., 2021)	Arithmetic	Number	8	1,000	MIT License
AQUA (Ling et al., 2017)	Arithmetic	Multi-choice	4	254	Apache-2.0
StrategyQA (Geva et al., 2021)	Commonsense	T/F	6	2,290	MIT license
CommonsenseQA (Talmor et al., 2019)	Commonsense	Multi-choice	7	1,221	Unspecified
BoolQ (Clark et al., 2019)	Commonsense	T/F	4	3,270	CC BY-SA 3.0
ARC-c (Clark et al., 2018)	Commonsense	Multi-choice	4	299	CC BY-SA 4.0
Date Understanding (Suzgun et al., 2023)	Symbolic	Multi-choice	3	250	MIT license
Penguins in a Table (Suzgun et al., 2023)	Symbolic	Multi-choice	3	146	MIT license
Colored Objects (Suzgun et al., 2023)	Symbolic	Multi-choice	3	250	MIT license
Object Counting (Suzgun et al., 2023)	Symbolic	Multi-choice	3	250	MIT license

Table 2: Comprehensive statistics of datasets utilized in our experiments. # EX. indicates the number of Chain-of-Thought (CoT) (Wei et al., 2022) prompting exemplars used for few-shot prompting. # EVAL. denotes the total count of evaluation samples in each dataset.

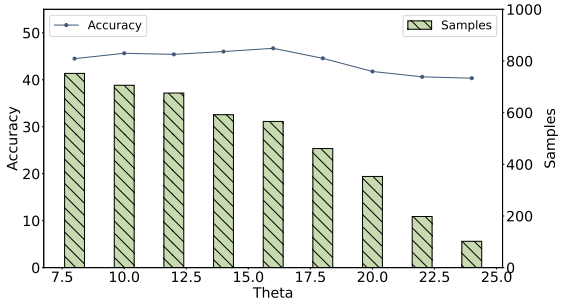


Figure 8: Threshold Analysis on GSM8K dataset.

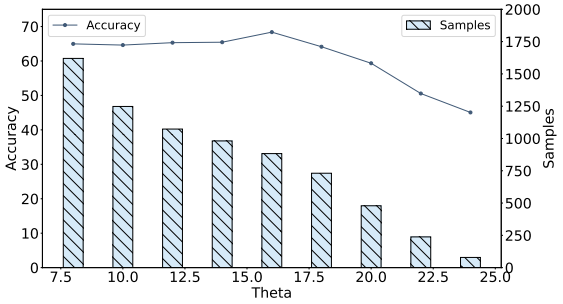


Figure 9: Threshold Analysis on StrategyQA dataset.

this setting are shown in 5.3, which we analyze in more detail in C.1. In the main experiment, we set the threshold θ to 16, and we analyzed the different threshold selections in C.2. In the clustering phase, we use `text-embedding-3-large` to obtain the corresponding embedding. We use NLL loss to react to the uncertainty of the model and use the model's respective tokenizer to get the number of generated tokens. Benefiting from the uncertainty assessment, we are able to first complete inference on a batch of confident samples. These samples are used as examples to bootstrap questions that are interrupted due to high uncertainty, similar to active learning, allowing our approach to be applied in test question only scenarios.

C Extended Analysis

C.1 Further Analysis of Relevance and Originality

In Figure 7, we examine the influence of relevance and originality weights on performance, utilizing the Mistral-7B model. We maintain the constraint $\lambda_1 + \lambda_2 = 1$, where λ_1 varies from 0 to 1, denoting an increasing emphasis on relevance. Our findings indicate that an increment in λ_1 corresponds with a gradual improvement in model performance, underscoring the positive role of relevance in enhancing model reasoning. However, beyond $\lambda_1 > 0.6$, there is a notable decline in performance, suggesting an overreliance on correlation and the detrimental impact of reasoning errors from similar samples. The model exhibits optimal performance when $\lambda_1 = \lambda_2 = 0.5$, striking a balance between relevance and originality. This balanced setting is adopted for our subsequent experiments, reflecting its effectiveness in optimizing model performance.

C.2 Threshold θ

In Figure 8 and Figure 9, we delve into the influence of threshold θ on the performance in the GSM8K and StrategyQA datasets. The histograms display the distribution of samples requiring exemplar introduction at various thresholds. The accompanying curves illustrate the corresponding accuracy at each threshold level. It is observed that the model's accuracy escalates with an in-

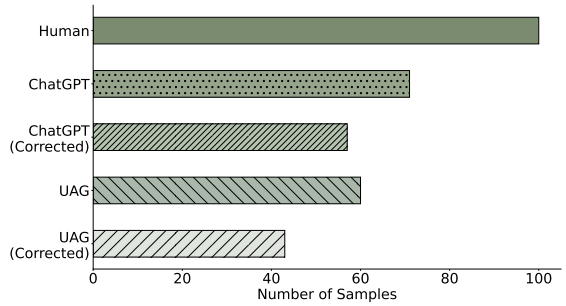


Figure 10: Error Identification and Correction on GSM8K dataset. UAG exhibits comparable error identification capabilities to ChatGPT.

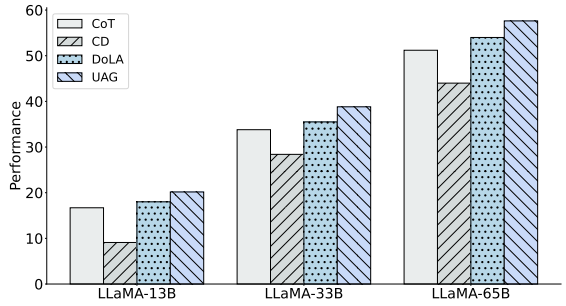


Figure 11: Performance comparison of UAG and various Decoding Enhancement Methods on GSM8K dataset. Using LLaMA-1 backbone (Touvron et al., 2023a), UAG consistently enhances performance across models of different parameter sizes.

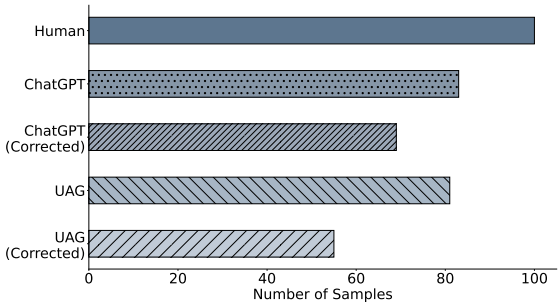


Figure 12: Error Identification and Correction on StrategyQA dataset. UAG exhibits comparable error identification capabilities to ChatGPT.

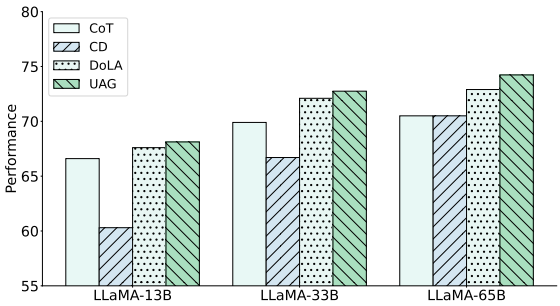


Figure 13: Performance comparison of UAG and various Decoding Enhancement Methods on StrategyQA dataset. Using LLaMA-1 backbone (Touvron et al., 2023a), UAG consistently enhances performance across models of different parameter sizes.

crease in θ, peaking before a rapid decline post $\theta > 16$. To understand this phenomenon, we examine the frequency of samples meeting the criteria of Eq 7 at varying thresholds, depicted through a histogram. This analysis reveals a decrease in the number of samples satisfying Eq 7 as θ ascends. A lower θ entails a larger subset of samples undergoing reasoning adjustment, which, given that many samples are correctly inferred initially, could inadvertently introduce errors. Conversely, a higher θ may fail to identify samples with reasoning errors, gradually like an exemplar-free Zero-Shot-CoT approach (Kojima et al., 2022) and resulting in a marked degradation in performance.

C.3 Correlation between Reasoning Errors and Model Uncertainty

In Figure 1, we observe a correlation between reasoning errors and model uncertainty, a phenomenon previously substantiated by various studies (Yuan et al., 2021; Fu et al., 2023a; Manakul et al., 2023). To delve deeper, we analyze 100 error-containing samples from the GSM8K and StrategyQA datasets, identified through labeled correct answers, as illustrate in Figure 10 and Figure **??**. These samples are processed using both ChatGPT and our UAG method for error localization and correction. Employing the reasoning process generated by Mistral-7B, we utilized `gpt-3.5-turbo-1106`, following the method outlined in Xie et al. (2023). ChatGPT exhibits a substantial error identification rate, correctly pinpointing 71% and 83% of errors in the datasets and successfully amending 57% and 69%, respectively. UAG demonstrates a comparable proficiency, correctly identifying 60% and 81% of the errors and effecting corrections in 43% and 55% of the cases. Notably, UAG capitalizes on the model's inherent uncertainty for judgment, obviating the need for reevaluation. This approach not only significantly reduces computational overhead but also remains versatile across models of varying parameter sizes.

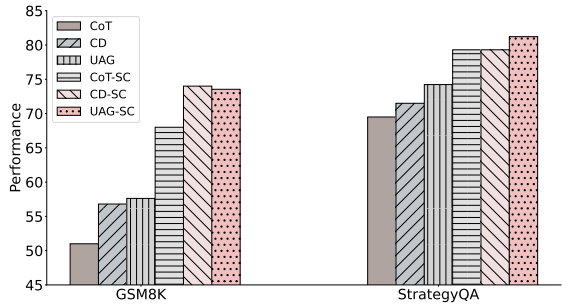


Figure 14: Performance Comparison in Multiple Reasoning Chains Scenario. Adhering to the configuration outlined by O'Brien and Lewis (2023), we utilize the LLaMA-65B backbone (Touvron et al., 2023a).

C.4 Comparison to Decoding Enhancement Method

In Section 2, we discuss a variety of reasoning enhancement methods. Due to experimental constraints, it was impractical to compare our approach with all notable methods. Consequently, we select two representative decoding enhancement methods for comparison: Contrastive Decoding (CD) (Li et al., 2023a) and Decoding by Contrasting Layers (DoLA) (Chuang et al., 2023). Utilizing the LLaMA-1 backbone (Touvron et al., 2023a), with baseline results sourced from Chuang et al. (2023), Figure 11 and Figure 13 illustrates that UAG consistently outperforms the DoLA method across all model sizes on both the GSM8K and StrategyQA datasets. O'Brien and Lewis (2023) highlight that meticulous hyperparameter selection can notably enhance the performance of CD methods in reasoning tasks. In Figure 14, our comparison with these carefully tuned CD results reveals that UAG not only demonstrates comparable performance but also exhibits a distinct edge in commonsense reasoning tasks. This comparison underscores UAG's ability to achieve significant performance improvements across a broader array of tasks.