Ada-R1: Hybrid CoT via Bi-Level Adaptive Reasoning Optimization

Haotian Luo[1,2]*, **Haiying He**[3]*, **Yibo Wang**[4], **Jinluan Yang**[5], **Rui Liu**[6]
Naiqiang Tan[6], **Xiaochun Cao**[1], **Dacheng Tao**[7], **Li Shen**[1,2]†
[1] Shenzhen Campus of Sun Yat-sen University;
[2] Center for AI Theoretical Foundation and Systems, Shenzhen Loop Area Institute;
[3] China Agricultural University; [4] Tsinghua University; [5] Zhejiang University;
[6] Didichuxing Co. Ltd; [7] Nanyang Technological University

Abstract

Recently, long-thought reasoning models achieve strong performance on complex reasoning tasks, but often incur substantial inference overhead, making efficiency a critical concern. Our empirical analysis reveals that the benefit of using Long-CoT varies across problems: while some problems require elaborate reasoning, others show no improvement—or even degraded accuracy. This motivates adaptive reasoning strategies that tailor reasoning depth to the input. However, prior work primarily reduces redundancy within long reasoning paths, limiting exploration of more efficient strategies beyond the Long-CoT paradigm. To address this, we propose a novel two-stage framework for adaptive and efficient reasoning. First, we construct a hybrid reasoning model by merging long and short CoT models to enable diverse reasoning styles. Second, we apply bi-level preference training to guide the model to select suitable reasoning styles (group-level), and prefer concise and correct reasoning within each style group (instance-level). Experiments demonstrate that our method significantly reduces inference costs compared to other baseline approaches, while maintaining performance. Notably, on five mathematical datasets, the average length of reasoning is reduced by more than 50%, highlighting the potential of adaptive strategies to optimize reasoning efficiency in large language models. Our code is coming soon at https://github.com/StarDewXXX/AdaR1

1 Introduction

Recent large language models (LLMs) such as OpenAI's O1[1] and Deepseek's R1[2] adopt extended and structured reasoning processes (Long-CoT) for LLMs to enhance problem-solving, achieving strong performance through human-like deliberation. Similarly, [3] and [4] enable MLLMs to possess such capabilities as well. However, the improved reasoning comes with high inference costs, including increased latency and resource consumption[5–7], which limits deployment in real-time or resource-constrained scenarios. Existing efficiency-oriented methods operate within the Long-CoT distribution, aiming to reduce redundancy through pruning or compression[8, 9, 6]. While effective to some extent, these approaches do not question whether long reasoning is necessary, overlooking potential gains from fundamentally shorter reasoning strategies. CoT-Valve[10] enables both long and short outputs but lacks adaptive selection based on input complexity, leading to suboptimal results.

Our investigation (presented in Section 3) about the benefit of Long-CoT reasoning reveals a crucial insight: the utility of long, elaborate reasoning chains is highly problem-dependent. While complex

*Equal contribution
†Corresponding Author: Li Shen (shenli6@mail.sysu.edu.cn)

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

 https://doi.org/10.52202/085713-1986

problems genuinely benefit from detailed, step-by-step derivations, many other problems can be solved accurately and more efficiently with shorter, more direct reasoning paths. In fact, for simpler problems, forcing a Long-CoT process might not only be wasteful but can sometimes even introduce errors or degrade performance. This observation strongly motivates the need for adaptive reasoning strategies – systems that can tailor the depth and style of their reasoning process to the specific demands of the input problem.

Inspired by these limitations, we propose a two-stage framework for efficient and adaptive reasoning by enabling models to choose between distinct reasoning strategies. The first stage constructs a hybrid model capable of generating both Long-CoT and Short-CoT outputs. The second introduces Bi-Level Adaptive Reasoning Optimization , a training method comprising: (i) Group-Level Preference, guiding the model to select an appropriate reasoning style based on input complexity, and (ii) Instance-Level Preference, encouraging concise yet accurate reasoning within the chosen style. This dual-level adaptation allows dynamic allocation of computational resources, yielding substantial efficiency gains without sacrificing performance. On MATH[11], our method reduces reasoning length by 58% with no accuracy loss, and on GSM8K[12], by 74% with improved accuracy. These results highlight the effectiveness of adaptive reasoning in balancing quality and efficiency in large-scale models.

Our contributions can be summarized as follows:

- We conduct an empirical analysis investigating the benefits of long Chain-of-Thought (CoT) reasoning relative to shorter CoT approaches, identifying the conditions under which extended reasoning paths offer tangible advantages.
- We propose using Adaptive Hybrid Reasoning Model to enhance inference efficiency, accompanied by a novel training pipeline (Ada-R1). Comprehensive experiments demonstrate that our proposed method achieves excellent performance, significantly improving efficiency while maintaining high accuracy.
- We perform further analyses on the resulting Adaptive Hybrid Reasoning Model to gain deeper insights into its characteristics and operational behavior. And we will release the model weights of the Adaptive Hybrid Reasoning Model to the public to encourage further research and application by the community.

2 Related Work

Model Merging Model merging [13] is an emerging technique that fuses parameters from multiple trained models into one without access to original training data. Recent methods include parameter interpolation [14] and alignment-based strategies [15], with applications in LLMs, multimodal models, and other machine learning subfields. Beyond simple linear averaging, advanced methods such as DARE [16], TIES-Merging [17], and AdaMerging [18] have been proposed. DARE reduces redundancy by dropping and rescaling delta parameters. TIES-Merging mitigates interference by trimming and aligning parameter signs. AdaMerging improves performance via entropy-based layer or task weighting on unlabeled data. In contrast to traditional model merging that consolidates capabilities from multiple models, our work enables a single model to adaptively choose between Long-CoT and Short-CoT reasoning for each instance, aiming to optimize computational efficiency rather than multi-task performance.

Efficient Reasoning A variety of methods have been proposed for improved reasoning efficiency. Several techniques apply post-training strategies to shorten reasoning paths. [6] constructs preference datasets using DPO and SimPO, guiding models toward concise reasoning through preference-based fine-tuning. O1-Pruner[8] samples CoTs to build baselines for length and accuracy, then applies offline optimization to reduce reasoning length without harming performance. Similarly, [19] leverages simple fine-tuning on self-generated concise CoTs obtained via best-of-N sampling and few-shot prompting. Some approaches focus on token-level compression. TokenSkip[20], for instance, removes tokens selectively based on their estimated importance within the CoT. CoT-Valve[10], in contrast, manipulates the parameter space to produce CoTs with varying degrees of compression. Besides, various methods adopt different reasoning paradigms for efficiency. For instance, COCONUT[21] and CCOT[5] enable reasoning within the latent space, reducing the need for explicit token-level generation. Speculative Thinking[22] enhances small model inference by allowing large models to guide them during reasoning. Similarly, LightThinker[23] achieves efficiency by dynamically compressing intermediate thoughts throughout the reasoning process. Also,

some works ([24],[25], [26], [27], [28], [29]) design novel reasoning paradigms for efficiency. [30] also explores model merging technical for reasoning efficiency. Different from most works, our work solves reasoning efficiency in a novel adaptive reasoning perspective.

3 Motivation

3.1 Problem Setup

Chain-of-Thought (CoT) prompting has emerged as a powerful technique for enhancing the reasoning capabilities of large language models. Within the CoT paradigm, a distinction can be made between Long-CoT, which involves generating detailed and extensive thinking steps, and Short-CoT, which directly generate solving steps.

3.2 When Do We Need Long-CoT?

Simply applying Long-CoT to all problems introduces unnecessary overhead, especially for easier tasks where detailed reasoning brings little or no benefit. To understand when Long-CoT is truly needed, we empirically analyze its effectiveness across different problem types. We compare Long-CoT and Short-CoT on a mixed dataset (MixMathematics) composed of samples from AIME[31], MATH, and GSM8K (details in Section 5.1). We use DeepSeek-R1-Distill-Qwen-7B for Long-CoT, and fine-tune it with 2,000 Short-CoT samples from Qwen2.5-Math-7B-Instruct[32] to create a consistent Short-CoT model. We avoid using Qwen2.5 directly due to its differing training format, which may affect later merging and sampling. From 2,500 problems, we generate 12 responses per model per question and remove cases where both models fail completely. We then calculate accuracy gains (Long-CoT accuracy minus Short-CoT accuracy).

As shown in Figure 1 (left), nearly half the samples show no improvement from Long-CoT, and some even suffer performance drops. Further analysis (Figure 1, right) groups samples by the average length of their Long-CoT outputs—longer CoTs tend to correspond to harder problems. We find that Long-CoT significantly improves accuracy on complex questions but provides little or no benefit for simpler ones.

Gain	Ratio
>0	48%
=0	45%
<0	7%

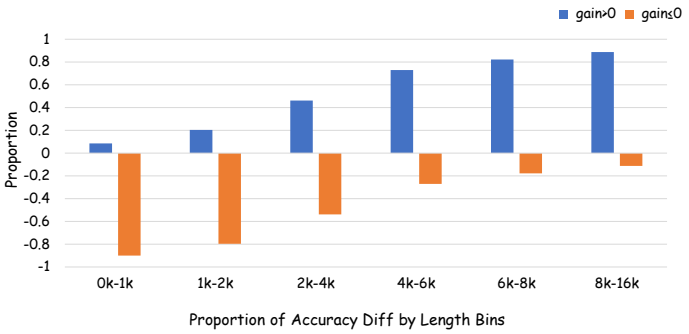

Figure 1: The proportion of gain in the data (left) and the relationship between CoT length and accuracy improvement (right), Long-CoT reasoning improves accuracy on difficult problems but has little effect or harms performance on easy ones.

3.3 A New Perspective on CoT Efficiency

Prior methods (Table 1), such as Overthinking [6], kimi-1.5 [33], and O1-Pruner, typically operate within a limited optimization scope but generally maintain performance stability or incur only a slight drop, with O1-Pruner notably achieving no performance decrease. In contrast, methods designed for a broad optimization scope, including Model Merge and CoT-Valve, did not consider how to tackle easy and different problems, rendering the model incapable of determining its reasoning depth according to the inherent difficulty of the task. Thus they frequently result in significant performance degradation. In a nutshell, methods with a restricted optimization can generally preserve performance but lose the chance to utilize shorter CoT. However, approaches capable of utilize broader CoT

https://doi.org/10.52202/085713-1986

Method	CoT Optimization Scope	Performance (Accuracy)
Overthinking[6]	Limited ×	Slightly Dropped ✓
kimi-1.5[33]	Limited ×	Slightly Dropped ✓
O1-Pruner	Limited ×	Not Dropped ✓
Naive Merge	Broad ✓	(mostly) Dropped ×
CoT-Valve	Broad ✓	Dropped ×
Ada-R1(Ours)	Broad ✓	Slightly Dropped ✓

Table 1: Comparison of Different Methods. "Limited" indicates optimization within the Long-CoT distribution, restricting efficiency. "Broader" covers both Long- and Short-CoT, enabling shorter, more efficient responses. "Slightly dropped" means accuracy decreased by less than 3%, while "dropped" refers to a decrease greater than 3%.

distribution have struggled to maintain accuracy due to their inability to adapt adequate reasoning depth to problem complexity.

The finding mentioned in last section motivates us to address the efficiency challenge of Long-CoT models from a novel perspective: enabling the reasoning model to adaptively select an appropriate reasoning mode (long or short CoT) for different problems, and then generate a correct and concise CoT in the determined mode. Our proposed method (Ada-R1) differentiates itself by successfully achieving a broad optimization scope while incurring only a marginal performance decrement. This demonstrates a more favorable trade-off between efficiency and accuracy compared to existing broad-scope optimization techniques.

4 Bi-Level Adaptive Reasoning Optimization

4.1 Problem Setup

We consider a LLM parameterized by θ and denoted as π_θ. In the context of math problem solving, the LLM accepts a sequence $x = [x^1, \ldots, x^n]$, commonly termed as the problem, and then generate a corresponding solution $y = [y^1, \ldots, y^m]$. Hence, the solution y is construed as a sample drawn from the conditional probability distribution $\pi_\theta(\cdot|x)$. The conditional probability distribution $\pi_\theta(y|x)$ can be decomposed as follows:

$$\pi_\theta(y|x) = \prod_{j=1}^{m} \pi_\theta(y^j|x, y^{<j}). \tag{1}$$

We consider two LLMs: one trained to generate long, reflective Chain-of-Thought (CoT) reasoning (*Long-CoT model*, denoted as θ_L) and the other trained for short and concise reasoning paths (*Short-CoT model*, denoted as θ_S). These two models are typically fine-tuned with different CoT and demonstrate distinct reasoning patterns.

4.2 Method Overview

Our method consists of two stages, shown in Figure 2. First, we merge a Long-CoT model and a Short-CoT model to obtain a unified reasoning model capable of generating both types of reasoning paths. This allows exploration over a broader CoT distribution. In the second stage, we apply Bi-Level Preference Training: for group-level preference, the model learns to choose between long and short reasoning group based on the input; for instance-level preference, it learns to compress the reasoning path to improve efficiency within the chosen group determined by group-level preference.

4.3 Stage I: Long-and-Short Reasoning Merge

To enable flexible reasoning behaviors within a single model, we first perform model merging with long and short models. We adopt a simple yet effective strategy of linearly merging their parameters. Given two models with parameters θ_L and θ_S, we compute the merged model as:

$$\theta_H = \alpha\theta_L + (1 - \alpha)\theta_S, \tag{2}$$

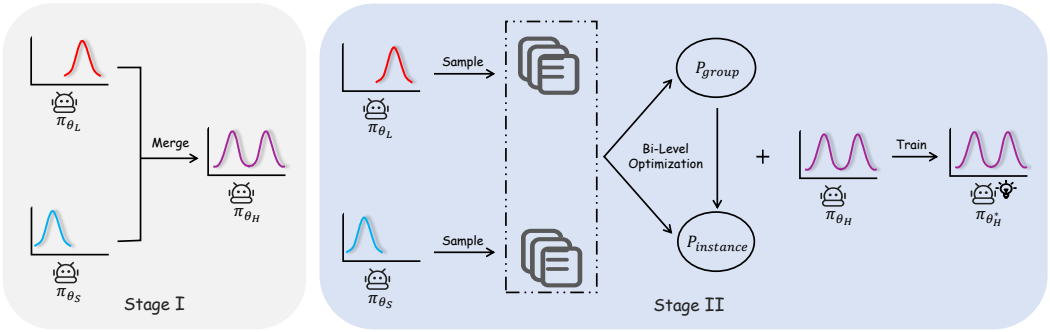

Figure 2: Pipeline of Ada-R1. At Stage I, we fused the models to obtain π_{θ_H}. In Stage II, we sample from both long and short models and then elicit the group-level and instance-level preference. After this, we optimize π_{θ_H} at both group and instance level to obtain a hybrid adaptive reasoning model.

where $\alpha \in [0, 1]$ is a merging coefficient that balances the contribution from each model. The resulting hybrid reasoning model, π_{θ_H}, inherits the capacity to generate both long and short CoT depending on the input.

This merged model expands the diversity of the CoT distribution it can produce, laying the foundation for adaptive reasoning. By combining the strengths of both reasoning styles, it enables the model to potentially match different problem types with suitable reasoning strategies, which is key to improving efficiency in the next stage.

4.4 Stage II: Bi-Level Preference Training

In this stage, we introduce a Bi-Level Preference Training strategy to fine-tune the model toward efficient reasoning. The core idea is to train the model to: (1) select the appropriate reasoning style (long or short) for each problem (*group-level preference*) and (2) further compress the reasoning within the determined chosen group (*instance-level preference*).

Group Labels. We define a group label g to denote the reasoning style of a response group. Let g_L denote the *long reasoning group* and g_S denote the *short reasoning group*. For a given input problem x, a generated resposne (solution) y belongs to one of the two groups. We use $\{y_i\}_{g=g_L}$ to denote the set of K Long-CoT responses generated by the Long-CoT model θ_L, and $\{y_j\}_{g=g_S}$ for the corresponding short responses from the Short-CoT model θ_S.

Group-Level Preference. For each math problem x in the dataset $\mathcal{D}$, we sample K solutions from both the long and short reasoning models. Let $\{y_i^L\}_{i=1}^K$ and $\{y_j^S\}_{j=1}^K$ be the respective sample sets. We define the approximated accuracy expectation for each group as:

$$\hat{\mathbb{E}}[C^L(x)] = \frac{1}{K}\sum_{i=1}^{K} \mathbb{1}[\text{Correct}(y_i^L)], \quad \hat{\mathbb{E}}[C^S(x)] = \frac{1}{K}\sum_{j=1}^{K} \mathbb{1}[\text{Correct}(y_j^S)], \tag{3}$$

where $\mathbb{1}[\cdot]$ is the indicator function. Then we introduce a preference margin threshold $\epsilon > 0$. The group-level preference for x is then determined as:

$$\begin{cases} g_L \succ g_S \mid x & \text{if} \quad \hat{\mathbb{E}}[C^L(x)] - \hat{\mathbb{E}}[C^S(x)] > \epsilon, \\ g_S \succ g_L \mid x & \text{if} \quad \hat{\mathbb{E}}[C^L(x)] - \hat{\mathbb{E}}[C^S(x)] \leq \epsilon. \end{cases}$$

Given the group-level preference for an input x, we form training pairs from the Cartesian product of the two groups. For example, if $g_L \succ g_S \mid x$, we construct the preference pairs as:

$$\mathcal{P}_{\text{group}}(x) = \left\{ (x, y_i^L, y_j^S) \;\middle|\; i \in [1, K], j \in [1, K] \right\}. \tag{4}$$

From this set of pairs, we randomly sample a subset contain M_1 pairs to construct DPO training tuples (x, y_w, y_l), where y_w is the preferred (chosen) response and y_l is the less preferred (rejected).

https://doi.org/10.52202/085713-1986

For all $x \in \mathcal{D}$, we perform group-level preference assignment by comparing the sampled long and short responses as described above. These tuples are then aggregated into a new dataset $\mathcal{D}_{\text{group}} = \{(x, y_w, y_l)\}$, which serves as supervision for optimizing the DPO objective at the group level.

Instance-Level Preference. Once the preferred group $g^* \in \{g_L, g_S\}$ is determined for a given x, we further construct *instance-level preferences* within that group to encourage more concise reasoning. We compare response pairs (y_a, y_b) such that both belong to the same group (e.g., $y_a, y_b \in \{y_i^L\}$), and prefer the shortest correct response. For dispreferred samples, we select M_2 longest responses. Formally, for each $x \in \mathcal{D}$ with preferred group g^*, we first identify the subset of correct responses $\{y_i\}_{\text{correct}} \subseteq \{y_i\}_{g=g^*}$. Among these, we select the shortest correct response as the preferred instance:

$$y_w = \arg \min_{y \in \{y_i\}_{\text{correct}}} |y|.$$

To construct instance-level preference pairs, we then select the M_2 longest responses from the entire group $\{y_i\}_{g=g^*}$. Denote these as $\{y_{l_j}\}_{j=1}^{M_2}$. This yields a dataset of instance-level training tuples:

$$\mathcal{D}_{\text{instance}} = \left\{ (x, y_w, y_l) \;\middle|\; y_w = \arg\min_{y \in \{y_i\}_{g=g^*}^{\text{correct}}} |y|, y_l \in \arg\max_{y \in \{y_i\}_{g=g^*}}^{(M_2)} |y| \right\}$$

These instance-level preferences encourage the model not only to reason correctly, but also to do so concisely within the preferred reasoning style.

We sample such intra-group pairs and use them as additional training data for DPO to encourage the model to favor more concise reasoning within each group.

Objective. Given collelcted preference datasets $\mathcal{D}_{\text{group}}$ and $\mathcal{D}_{\text{instance}}$ sampled from p^* which contains N preference pairs (x, y_w, y_l). With a parameter β controlling the deviation from the reference model p_{ref}, DPO optimize the model by:

$$\max_{\pi_{\theta_H}} \mathbb{E}_{(x,y_w,y_l) \sim \mathcal{D}_{\text{group}} \cup \mathcal{D}_{\text{instance}}} \left[\log \sigma \left(\beta \log \frac{\pi_{\theta_H}(y_w \mid x)}{\pi_{\theta_{ref}}(y_w \mid x)} - \beta \log \frac{\pi_{\theta_H}(y_l \mid x)}{\pi_{\theta_{ref}}(y_l \mid x)} \right) \right]$$

5 Experiments

5.1 Setup

Long-CoT Models. The long thought models we chosen for our experiment are DeepSeek-R1-Distill-Qwen-7B and DeepSeek-R1-Distill-Qwen-1.5B, which have demonstrated excellent performance on most math problem-solving tasks. For both models, we utilize full-parameter fine-tuning.

Short CoT Models. Since model merging requires Shot-CoT models, we face two issues with existing Shot-CoT models: (1) they often employ templates that differ from those used in Long-CoT models; (2) they tend to exhibit substantial parameter deviations from the base model, which introduces instability during the merging process[18, 34]. To address these challenges, we fine-tune the Long-CoT models using a small number of short CoT examples to obtain the corresponding Shot-CoT models. This approach ensures consistency in template usage and maintains a closer parameter proximity between the two models.

Dataset. Following s1[35] and Light-R1[36], we construct a mixed training dataset to ensure coverage across mathematical problems of varying difficulty levels. Specifically, we combine GSM8K, MATH, and AIME datasets in a ratio of 1:3:1, resulting in a total of 2,500 diverse math problems.

Evaluation. We use the GSM8K test set, the MATH test set, and AIME25 as in-distribution evaluation data, while Olympiad[37] and Minerva[38] are employed as out-of-distribution test sets. For evaluation metrics, we consider both accuracy and sequence length. For small-scale datasets such as AIME25, we report results averaged over four independent runs to reduce randomness and improve reliability. Additionally, we report the average accuracy degrade rate and the average length reduction rate across all test sets.

Table 2: Accuracy (shown above) and length (shown below) of models and methods on different benchmarks. Avg represents the change in length and accuracy compared to the Long model (+ for increase, - for decrease).

Model \ Bench	AIME25	MATH500	GSM8K	Olympiad	Minerva	Avg.(%)
7B Models						
Long(R1-distill)	38.3 (11005)	90.2 (3534)	88.9 (1014)	54.4 (7492)	35.7 (4533)	- -
Short	10.0 (957)	78.6 (591)	89.5 (272)	39.4 (910)	28.6 (579)	−19.97% (-84.57%)
Merge	21.7 (9079)	79.4 (916)	88.4 (236)	41.2 (3743)	25.7 (1734)	−18.63% (-56.02%)
DPO	35.8 (9976)	89.4 (2334)	86.0 (360)	55.2 (5309)	35.6 (3281)	-3.56% (-33.26%)
O1-Pruner	40.0 (9353)	92.4 (2212)	89.4 (377)	55.3 (5295)	35.3 (3259)	+2.48% (-34.53%)
CoT-Valve	22.5 (5024)	78.6 (747)	87.9 (235)	39.6 (2313)	29.4 (629)	−18.41% (-73.06%)
Ada-R1(Ours)	35.8 (8426)	90.2 (1468)	90.3 (260)	52.4 (4889)	34.1 (1647)	-1.65% (-50.93%)
1.5B Models						
Long(R1-distill)	23.3 (12307)	81.0 (4416)	80.9 (1481)	41.6 (7687)	26.1 (5789)	- -
Short	9.0 (1098)	69.4 (740)	78.2 (269)	30.7 (1373)	22.4 (725)	−26.34% (-85.15%)
Merge	20.8 (9226)	71.8 (1740)	74.2 (251)	28.6 (3767)	20.0 (1399)	−10.12% (-59.10%)
DPO	20.8 (10224)	81.4 (3055)	74.8 (374)	42.8 (6319)	24.3 (3905)	−5.93% (-34.57%)
O1-Pruner	23.3 (9496)	82.6 (2782)	84.6 (726)	44.7 (5658)	28.3 (3964)	+2.18% (-33.75%)
CoT-Valve	14.2 (7744)	69.6 (1299)	76.3 (205)	28.7 (3169)	19.5 (867)	−19.61% (-67.52%)
Ada-R1(Ours)	23.0 (9516)	80.8 (2455)	79.2 (341)	42.1 (5802)	23.5 (3021)	-1.21% (-43.28%)

5.2 Competitive Methods

DPO. DPO are widely used baselines in reasoning optimization area. Follwoing the setting of [33, 6], we choose shortest sample as chosen samples and longest sample as rejected sample.

CoT-Valve. CoT-Valve enables dynamic control of Chain-of-Thought length using a single model by identifying and leveraging a controllable direction in the model's parameter space to generate compressed CoT.

O1-Pruner. O1-Pruner is a method designed to reduce reasoning overhead while maintaining model accuracy. It begins by establishing a baseline through pre-sampling, and then applies reinforcement learning-based finetuning.

5.3 Main Results

We can be seen from the Table 2 that: the Short and Merge models achieve the most significant length reduction compared to the Long Model. However, this efficiency gain is accompanied by a notable degradation in accuracy, exceeding 10 percentage points. Among the models that do not suffer significant accuracy degradation, our method achieves the best length reduction performance,

https://doi.org/10.52202/085713-1986

Table 3: Ablation study of each component on several benchmarks, showing that the Merge + bi-level achieves the best trade-off, with a 52.08% average length reduction and a minimal 0.51% accuracy degradation compared to others.

Model \ Bench	AIME25	MATH500	GSM8K	Avg.(%)
Long(R1-distill)	38.3 (11005)	90.2 (3534)	88.9 (1014)	- -
Merge	21.7 (9079)	79.4 (916)	88.4 (236)	−12.83% (-56.10%)
Merge + SFT	35.8 (11222)	84.6 (2314)	88.7 (375)	-3.82% (-31.86%)
Merge + instance level	24.2 (8514)	81.6 (886)	88.0 (212)	−10.86% (-38.20%)
Merge + group level	30.8 (9049)	87.8 (1565)	91.6 (359)	-3.31% (-46.03%)
Merge + bi level	35.8 (8426)	90.2 (1468)	90.3 (260)	-0.51% (-52.08%)

reaching 50.93% for the 7B model and 43.28% for the 1.5B model. Compared to DPO, our approach demonstrates both more substantial length reduction and significantly less accuracy degradation. While O1-Pruner maintains high accuracy, its length reduction effect is considerably weaker than that of our method.

5.4 Ablation Study

To assess each component's impact in our framework, we conduct an ablation study on AIME25, MATH500, and GSM8K. As shown in Table 3, the Merge model reduces average output length by 56.10%, but with a notable 12.83% drop in accuracy.

Supervised Fine-Tuning (SFT) on the merged model (using the chosen sample in our group level preference dataset), helps recover a significant portion of the lost accuracy, bringing the average degradation down to 3.82%. However, its average length reduction is less pronounced (31.86%) compared to the Merge model without further training.

Although optimization at the instance level leads to a substantial reduction in sequence length (38%), the model fails to distinguish which problems require long reasoning and which can be solved with shorter responses, resulting in more than a 10% drop in accuracy.

Introducing the group-level preference training after merging (Merge + group level) yields better results than SFT and only instance level. It achieves a higher average length reduction (46.03%) and a slightly better accuracy recovery, with only a 3.31% average degradation relative to the baseline. This indicates that training the model to select the appropriate reasoning style is effective in balancing efficiency and accuracy.

The full method (Merge + bi level), combining group and instance level preference training, offers the best trade-off: 52.08% length reduction with only 0.51% accuracy loss. This result highlights the complementary benefits of the bi-level training approach: the group level guides the model towards suitable reasoning styles, and the instance level further refines the chosen style by favoring concise and correct responses, leading to a highly efficient and accurate hybrid reasoning model.

6 Further Evaluation

6.1 Thinking Ratio Study

To investigate the thinking characteristics of different models, we propose the "Thinking Ratio" metric. This metric is designed to detect whether a response constitutes a deep thinking (Long-CoT) sample. Long-CoT responses typically include unique keywords (e.g., 'wait', 'recheck'). By detecting the

presence of these keywords in a response, we can determine if it is a deep thinking sample. This detection method is more generalizable than relying solely on response length. We use a subset of Math Testset. Using the method described above, we analyzed the proportion of deep thinking samples for each model. Furthermore, for each category (thinking/non-thinking samples), we also calculated their accuracy.

The results are shown in Figure 3. The baseline Long-CoT model predominantly employs deep thinking (0.98), yielding high accuracy. In contrast, the Naive Merge model drastically shifts towards non-thinking responses (0.94) but suffers significant accuracy degradation on both thinking (0.68) and non-thinking (0.81) paths. DPO shows a moderate shift to non-thinking (0.34) while preserving accuracy. Our Ada-R1 model achieves a more significant shift towards non-thinking (0.72) than DPO, yet crucially maintains high accuracy for these dominant non-thinking responses (0.96), unlike the Naive Merge. This demonstrates Ada-R1's effective adaptation, utilizing efficient shorter paths without substantial accuracy loss.

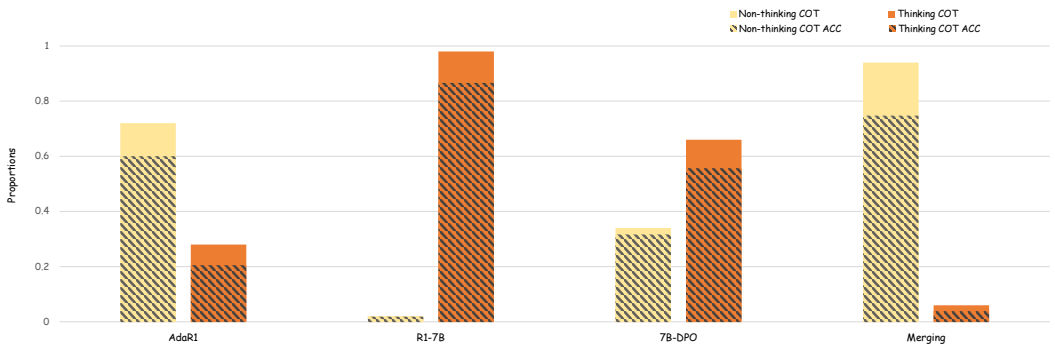

Figure 3: The proportion and accuracy of thinking and non-thinking in different methods, Ada-R1 can achieve a good balance and accuracy between thinking and non-thinking.

6.2 Adaptive Reasoning Study

This section evaluates the adaptive reasoning ability of Ada-R1 (7B) on the MATH dataset, which is divided into five difficulty levels (Level 1–5). We analyze both the model's thinking ratio (Long-CoT usage) and its average accuracy across these levels. As shown in the left part of Figure 4, the thinking ratio increases significantly with task difficulty. Level 1 problems have the lowest Long-CoT usage, while Level 5 shows the highest, indicating that Ada-R1 adaptively chooses to think more on harder problems. In terms of accuracy (Figure 4, right), Ada-R1 achieves strong performance across difficulty levels. Its accuracy is comparable to that of a full Long-CoT model (Deepseek-R1-Qwen-7B-Distill) and consistently higher than the Short-CoT model, especially on Levels 3 to 5. These results support our hypothesis from Section 3: Ada-R1 can selectively apply Long-CoT when needed, achieving a better balance between accuracy and efficiency.

6.3 Merge Model Analysis

To verify the effectiveness of the model merging strategy, we compute the average loss of the merged model on both Long-CoT and Short-CoT samples, and compare these values with those of the original Long-CoT and Short-CoT models. As shown in Table 4, the merged model exhibits similar losses to the corresponding original models on both types of samples. This indicates that the merged model preserves the reasoning capabilities of both individual models, thereby enabling it to fully leverage population-level preference signals during the second-stage fine-tuning. These results support the effectiveness of our dual-level preference training setup, where the merged model benefits from a broader reasoning space while being guided by meaningful preference supervision.

6.4 Out of Domain Test

To further assess the effectiveness of our method, we additionally evaluate it on non-mathematical reasoning benchmarks, including LogiQA[39], GPQA[40], and MMLU[41](we report results on both

https://doi.org/10.52202/085713-1986

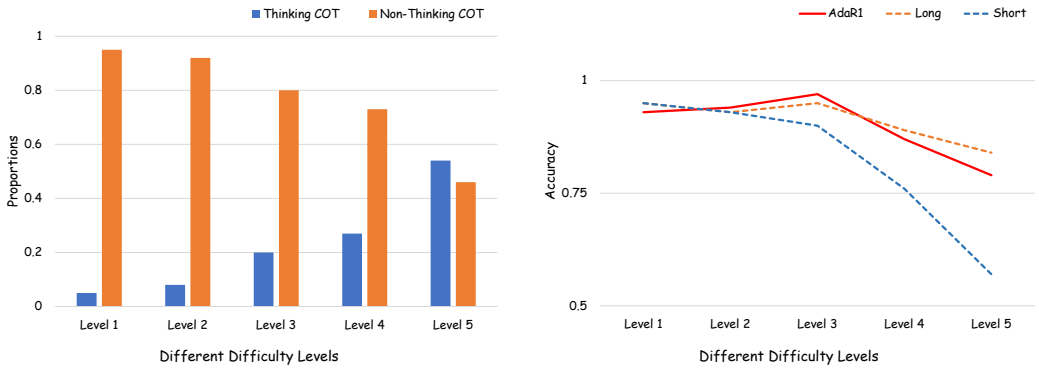

Figure 4: The ratio of thinking and non-thinking CoTs of Ada-R1-7B on different MATH levels (left) and the accuracy on different MATH levels of different models (right). As the difficulty increases, Ada-R1 is able to think more on harder problems and maintain higher accuracy.

Table 4: Average losses of the merged model compared with the original Long-CoT and Short-CoT models, showing that the merged model retains both reasoning abilities.

Model	Long-CoT Loss	Short-CoT Loss
Long (R1-distill)	0.2106	–
Short	–	0.1570
Merge	0.2141	0.2028

Table 5: Evaluation results of Ada-R1 on non-mathematical reasoning tasks.

Model \ Bench	LogiQA	GPQA	MMLU(stem)	MMLU(humanities)	Avg.(%)
Long(R1-distill)	45.1 (3269)	30.3 (7302)	43.3 (2336)	16.3 (1476)	- -
DPO	44.2 (2650)	31.3 (6592)	48.3 (1577)	21.0 (1270)	+7.26% (-15.95%)
Ada-R1(ours)	48.6 (2794)	32.8 (5101)	54.6 (1909)	30.9 (930)	+23.63% (-25.37%)

STEM and Humanities subsets), as shown in Table 5. The results indicate that Ada-R1 maintains strong efficiency–accuracy trade-offs beyond purely mathematical problems, demonstrating its generalizability to broader reasoning domains.

7 Conclusion

In this paper, we demonstrate through empirical analysis that the benefits of Long-CoT reasoning vary significantly depending on the problem. Motivated by this, we propose a novel two-stage training framework for adaptive reasoning. Experiments show that model trained with our method can reason adaptively to different problems. And our method significantly reduces inference costs while preserving performance, highlighting the promise of adaptive strategies for optimizing reasoning efficiency in large language models.

8 Acknowledgement

This work is supported by National Key R&D Projects (NO. 2024YFC3307100), NSFC Grant (No. 62576364), National Natural Science Foundation of China (No.62025604, 62411540034), Shenzhen Basic Research Project (Natural Science Foundation) Basic Research Key Project (NO. JCYJ20241202124430041), CCF-DiDi GAIA Collaborative Research Funds (NO. CCF-DiDi GAIA 202419 and CCF-DiDi GAIA 202519).

References

[1] OpenAI. Learning to reason with llms. https://openai.com/index/learning-to-reason-with-llms/, 2024. [Accessed 19-09-2024].

[2] DeepSeek-AI team. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.

[3] Huanjin Yao, Qixiang Yin, Jingyi Zhang, Min Yang, Yibo Wang, Wenhao Wu, Fei Su, Li Shen, Minghui Qiu, Dacheng Tao, and Jiaxing Huang. R1-sharevl: Incentivizing reasoning capability of multimodal large language models via share-grpo, 2025.

[4] Huanjin Yao, Jiaxing Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang, Yuxin Song, Haocheng Feng, Li Shen, and Dacheng Tao. Mulberry: Empowering mllm with o1-like reasoning and reflection via collective monte carlo tree search, 2024.

[5] Jeffrey Cheng and Benjamin Van Durme. Compressed chain of thought: Efficient reasoning through dense representations, 2024.

[6] Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do not think that much for 2+3=? on the overthinking of o1-like llms, 2025.

[7] Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Hanjie Chen, and Xia Hu. Stop overthinking: A survey on efficient reasoning for large language models, 2025.

[8] Haotian Luo, Li Shen, Haiying He, Yibo Wang, Shiwei Liu, Wei Li, Naiqiang Tan, Xiaochun Cao, and Dacheng Tao. O1-pruner: Length-harmonizing fine-tuning for o1-like reasoning pruning, 2025.

[9] Daman Arora and Andrea Zanette. Training language models to reason efficiently, 2025.

[10] Xinyin Ma, Guangnian Wan, Runpeng Yu, Gongfan Fang, and Xinchao Wang. Cot-valve: Length-compressible chain-of-thought tuning, 2025.

[11] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

[12] 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. Training verifiers to solve math word problems, 2021.

[13] Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications and opportunities, 2024.

[14] Yue Zhou, Yi Chang, and Yuan Wu. Mixup model merge: Enhancing model merging performance through randomized linear interpolation, 2025.

[15] Rishabh Bhardwaj, Do Duc Anh, and Soujanya Poria. Language models are homer simpson! safety re-alignment of fine-tuned language models through task arithmetic, 2024.

[16] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch, 2024.

 https://doi.org/10.52202/085713-1986

[17] Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models, 2023.

[18] Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning, 2024.

[19] Tergel Munkhbat, Namgyu Ho, Seo Hyun Kim, Yongjin Yang, Yujin Kim, and Se-Young Yun. Self-training elicits concise reasoning in large language models, 2025.

[20] Heming Xia, Yongqi Li, Chak Tou Leong, Wenjie Wang, and Wenjie Li. Tokenskip: Controllable chain-of-thought compression in llms, 2025.

[21] Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space, 2024.

[22] Wang Yang, Xiang Yue, Vipin Chaudhary, and Xiaotian Han. Speculative thinking: Enhancing small-model reasoning with large model guidance at inference time, 2025.

[23] Jintian Zhang, Yuqi Zhu, Mengshu Sun, Yujie Luo, Shuofei Qiao, Lun Du, Da Zheng, Huajun Chen, and Ningyu Zhang. Lightthinker: Thinking step-by-step compression, 2025.

[24] Chenxu Yang, Qingyi Si, Yongjie Duan, Zheliang Zhu, Chenyu Zhu, Zheng Lin, Li Cao, and Weiping Wang. Dynamic early exit in reasoning models, 2025.

[25] Jiayi Pan, Xiuyu Li, Long Lian, Charlie Snell, Yifei Zhou, Adam Yala, Trevor Darrell, Kurt Keutzer, and Alane Suhr. Learning adaptive parallel reasoning with language models, 2025.

[26] Wenjie Ma, Jingxuan He, Charlie Snell, Tyler Griggs, Sewon Min, and Matei Zaharia. Reasoning models can be effective without thinking, 2025.

[27] Ziqing Qiao, Yongheng Deng, Jiali Zeng, Dong Wang, Lai Wei, Fandong Meng, Jie Zhou, Ju Ren, and Yaoxue Zhang. Concise: Confidence-guided compression in step-by-step efficient reasoning, 2025.

[28] Ren Zhuang, Ben Wang, and Shuifa Sun. Accelerating chain-of-thought reasoning: When goal-gradient importance meets dynamic skipping, 2025.

[29] Junjie Yang, Ke Lin, and Xing Yu. Think when you need: Self-adaptive chain-of-thought learning, 2025.

[30] Han Wu, Yuxuan Yao, Shuqi Liu, Zehua Liu, Xiaojin Fu, Xiongwei Han, Xing Li, Hui-Ling Zhen, Tao Zhong, and Mingxuan Yuan. Unlocking efficient long-to-short llm reasoning with model merging, 2025.

[31] MAA. American invitational mathematics examination - aime. In *American Invitational Mathematics Examination - AIME 2024*, February 2024.

[32] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

[33] Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, Chuning Tang, Congcong Wang, Dehao Zhang, Enming Yuan, Enzhe Lu, Fengxiang Tang, Flood Sung, Guangda Wei, Guokun Lai, Haiqing Guo, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haotian Yao, Haotian Zhao, Haoyu Lu, Haoze Li, Haozhen Yu, Hongcheng Gao, Huabin Zheng, Huan Yuan, Jia Chen, Jianhang Guo, Jianlin Su, Jianzhou Wang, Jie Zhao, Jin Zhang, Jingyuan Liu, Junjie Yan, Junyan Wu, Lidong Shi, Ling Ye, Longhui Yu, Mengnan Dong, Neo Zhang, Ningchen Ma, Qiwei Pan, Qucheng Gong, Shaowei Liu, Shengling Ma, Shupeng Wei, Sihan Cao, Siying Huang, Tao Jiang, Weihao Gao, Weimin Xiong, Weiran He, Weixiao Huang, Wenhao Wu, Wenyang He,

Xianghui Wei, Xianqing Jia, Xingzhe Wu, Xinran Xu, Xinxing Zu, Xinyu Zhou, Xuehai Pan, Y. Charles, Yang Li, Yangyang Hu, Yangyang Liu, Yanru Chen, Yejie Wang, Yibo Liu, Yidao Qin, Yifeng Liu, Ying Yang, Yiping Bao, Yulun Du, Yuxin Wu, Yuzhi Wang, Zaida Zhou, Zhaoji Wang, Zhaowei Li, Zhen Zhu, Zheng Zhang, Zhexu Wang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Ziyao Xu, and Zonghan Yang. Kimi k1.5: Scaling reinforcement learning with llms, 2025.

[34] Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. Metagpt: Merging large language models using model exclusive task arithmetic, 2024.

[35] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025.

[36] Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, Haosheng Zou, Yongchao Deng, Shousheng Jia, and Xiangzheng Zhang. Light-r1: Curriculum sft, dpo and rl for long cot from scratch and beyond, 2025.

[37] Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024.

[38] Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models, 2022.

[39] Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning, 2020.

[40] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023.

[41] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021.

 https://doi.org/10.52202/085713-1986

A Selection of α in Stage I (Model Merge)

To select an appropriate value of α during the merge phase, we adopted the following approach. We randomly sampled 100 problems from the AIME exams (AIME 2025 is excluded). Using the 7B model as mentioned in our experiment, we evaluated performance under different values of α (0.9, 0.8, 0.7), computing both accuracy and thinking ratio for each setting. We selected $\alpha = 0.8$ as a balanced choice for Stage II training as it has relatively high accuracy and moderate thinking ratio.

Table 6: Performance of merged 7B models on 100 AIME problems.

α	**Accuracy**	**Thinking Ratio**
0.9	54.0	93%
0.8	40.0	48%
0.7	27.0	9%

B Training Details

For both models, we selected 2,500 problems from the mixed Mathematics as training data. For each problem, we sample 12 times. From each set of solutions, we randomly selected 2 solutions for training. After computing the rewards, we normalized the reward values. Both models are trained with 8 * A800-80G GPUs. The other hyperparameters used in the training process are presented in the table below.

Table 7: Hyperparameters for the Deepseek-Distill-1.5B and Deepseek-Distill-7B.

Hyperparameter	**Deepseek-Distill-1.5B**	**Deepseek-Distill-7B.**
cutoff_len	4096	4096
batch_size	32	32
learning_rate	5.0e-7	5.0e-7
num_train_epochs	2.0	2.0
lr_scheduler_type	constant	constant
M_1	4	4
M_2	2	2
beta	0.05	0.1

C Further Evaluation of Different Methods

We further evaluate the performance and efficiency of different methods (Ada-R1, DPO, O1-Pruner) across varying levels of problem difficulty, as illustrated in Figure 5 and Figure 6. Figure 5 presents the accuracy ratio of each method relative to a baseline model across different difficulty levels within the MATH dataset. The results indicate that while performance trends may vary, our proposed Ada-R1 method demonstrates strong robustness. Specifically, as the inherent difficulty of the mathematical problems increases, Ada-R1 is able to consistently maintain a high accuracy ratio.

Figure 6 show the ratio of average tokens consumed by each method to solve problems across the same difficulty spectrum. As expected, solving more difficult problems generally requires more reasoning steps and thus more tokens. However, Figure 6 reveals that Ada-R1 exhibits favorable token efficiency. Critically, when faced with increasing problem difficulty, Ada-R1 manages to solve these complex problems while utilizing relatively fewer tokens compared to other evaluated methods, showcasing its ability to achieve efficient reasoning even for demanding tasks.

Collectively, these figures highlight Ada-R1's ability to strike a beneficial balance between accuracy and efficiency. It not only maintains high performance on challenging problems (Figure 5) but also does so in a computationally efficient manner, particularly evident in its lower token usage for difficult instances (Figure 6), addressing limitations observed in prior methods.

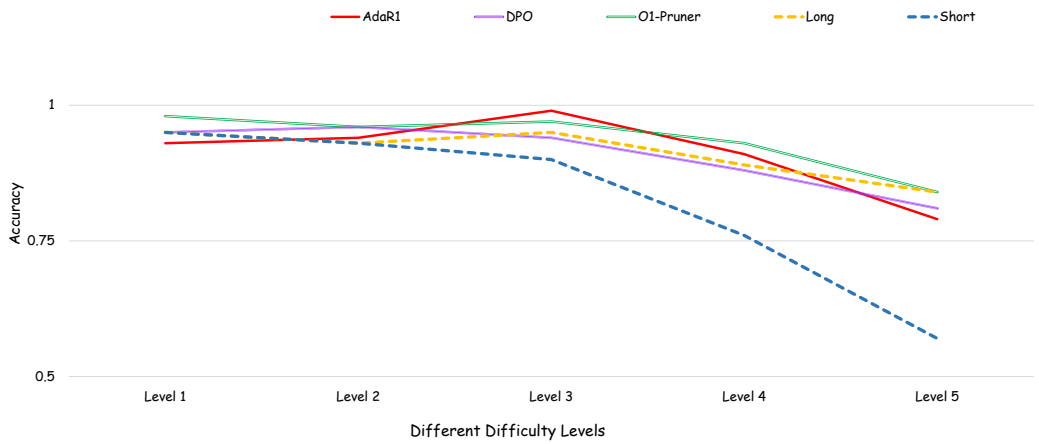

Figure 5: The ratio of accuracy at different MATH levels on different models. As the difficulty increases, Ada-R1 is able to maintain high accuracy.

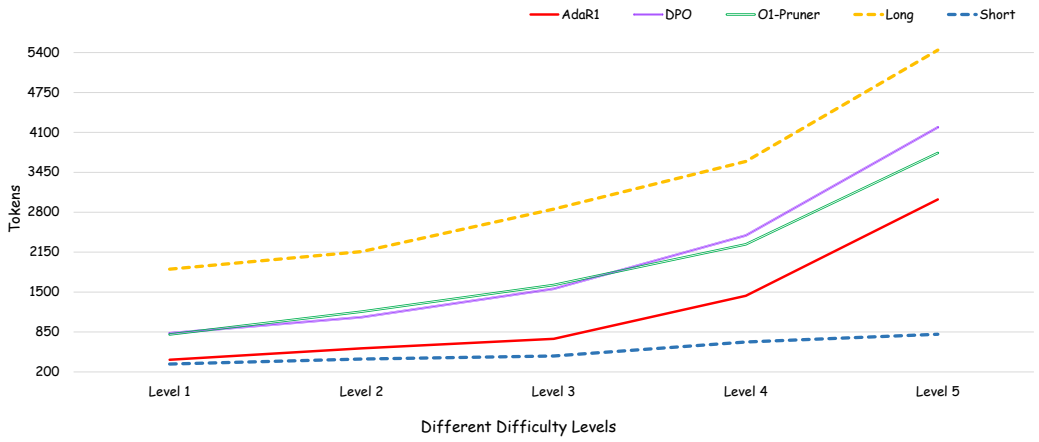

Figure 6: The ratio of average tokens on different models. As the difficulty increases, Ada-R1 is able to use relatively fewer tokens to solve difficult problems.

D Why Does Ada-R1 Work?

D.1 Early Mode Selection Assumption

While Ada-R1 significantly reduces inference cost by adaptively selecting a reasoning strategy during the inference stage, its design relies on an important assumption: the model determines the reasoning mode (Long-CoT or Short-CoT) immediately after receiving the problem input, without relying on any intermediate computation or external signals. In other words, the model is expected to assess the complexity of the problem and select an appropriate reasoning path before beginning the actual problem-solving process.

D.2 Visualization Setup

To investigate this question and better understand how Ada-R1 works, we design an experiment. We randomly select 500 problems from the training data and evaluate them using the 7B models (R1, and Ada-R1). For each problem, we extract the hidden states of the final token in the input sequence and use the last layer's hidden states as the internal representation of the problem. Based on previously computed group-level preferences (i.e., whether the problem should be solved using Long-CoT or Short-CoT), we assign a color label to each sample—red for problems requiring Long-CoT and blue

https://doi.org/10.52202/085713-1986

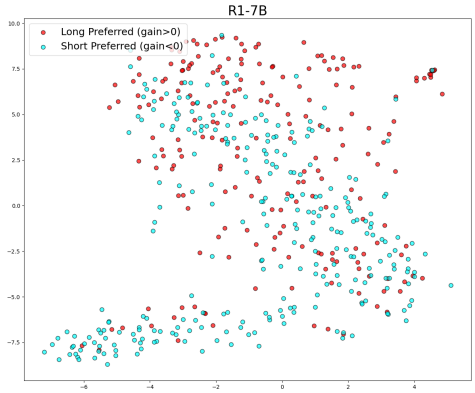

Figure 7: Visualization of R1 model

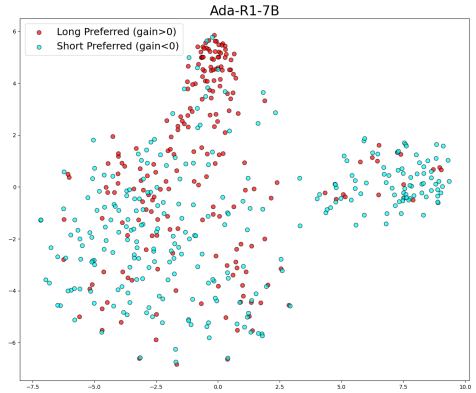

Figure 8: Visualization of Ada-R1 model

for those suitable for Short-CoT. We then apply t-SNE to project the high-dimensional hidden states into a two-dimensional space for visualization.

D.3 Ada-R1 Learns an Implicit Problem Classifier

From the visualization, we observe that after preference-based training, Ada-R1 is able to partially separate problems that require Long-CoT from those that do not, based solely on their internal representations. This suggests that the model learns to encode problem complexity in representation space, enabling early and efficient reasoning mode selection. Such a capability underpins the effectiveness of Ada-R1: by making an informed decision on the reasoning strategy at the problem stage, the model avoids unnecessary computation for simpler problems while retaining full reasoning capacity for more complex ones.

E Case Study

We select two cases (9 and 10) of R1 and Ada-R1 to better understand our proposed method. In case 9, Ada-R1 can infer answers to simple problems with a small amount of CoT; in case 10, Ada-R1 can infer answers to difficult problems with relatively few long CoT.

F Limitations

While our proposed adaptive hybrid reasoning framework shows strong performance on benchmark datasets, its applicability in real-world scenarios presents new challenges. In particular, real-world tasks often involve diverse input distributions, domain-specific knowledge, and evolving requirements that differ from curated academic datasets such as MATH or GSM8K. The current design assumes clear patterns in problem complexity that can guide reasoning strategy selection; however, in practical applications, this complexity may be harder to estimate reliably without additional context or metadata. We view this as a promising direction for future research to bridge the gap between controlled evaluation settings and real-world reasoning applications.

Question: Convert the point $(0, 3)$ in rectangular coordinates to polar coordinates. Enter your answer in the form (r, θ), where $r > 0$ and $0 \leq \theta < 2\pi$. Let's think step by step and output the final answer within boxed{}. Ground Truth: $(3, \frac{\pi}{2})$

Ada-R1: To convert the rectangular coordinate $(0, 3)$ to polar coordinates, we use the formulas: $r = \sqrt{x^2 + y^2}$ and $\theta = \arctan\left(\frac{y}{x}\right)$. For the point $(0, 3)$: - $r = \sqrt{0^2 + 3^2} = \sqrt{9} = 3$ - Since the point lies on the positive y-axis, the angle θ is $\frac{\pi}{2}$. Thus, the polar coordinates are $(3, \frac{\pi}{2})$. Final Answer: $\boxed{(3, \frac{\pi}{2})}$ ✓

R1: Okay, so I need to convert the rectangular coordinate $(0, 3)$ to polar coordinates. Hmm, polar coordinates are represented as (r, θ), where r is the radius (distance from the origin) and θ is the angle made with the positive x-axis. Let me recall the formulas to convert from rectangular to polar coordinates.

First, I remember that the radius r can be found using the Pythagorean theorem. Since in rectangular coordinates, the point is (x, y), the distance from the origin is $r = \sqrt{x^2 + y^2}$. So in this case, x is 0 and y is 3. Plugging that into the formula, r should be $\sqrt{0^2 + 3^2}$ which is $\sqrt{0 + 9} = \sqrt{9} = 3$. Okay, that seems straightforward.

Now, the angle θ. I know that θ is calculated using the arctangent function because $\tan\theta = \frac{y}{x}$. But wait, in this case, x is 0. Hmm, arctangent of $\frac{y}{0}$. But dividing by zero is undefined. I need to think about what angle θ corresponds to when x is 0.

In rectangular coordinates, when x is 0, the point lies on the y-axis. Since y is positive here ($y = 3$), the point is on the positive y-axis. So, in terms of polar coordinates, the angle θ should be the angle that points directly upwards along the y-axis. I think that's $\frac{\pi}{2}$ radians, which is 90 degrees. Because starting from the positive x-axis, if we rotate $\frac{\pi}{2}$ radians counterclockwise, we end up pointing straight up along the positive y-axis.

Let me double-check that. If θ is $\frac{\pi}{2}$, then converting back to rectangular coordinates, x should be $r\cos\theta = 3\cos(\frac{\pi}{2})$ and y should be $r\sin\theta = 3\sin(\frac{\pi}{2})$. $\cos(\frac{\pi}{2})$ is 0 and $\sin(\frac{\pi}{2})$ is 1, so that gives $x = 0$ and $y = 3$, which is exactly the point we started with. So that makes sense.

Wait, but could θ also be $\frac{3\pi}{2}$? No, because $\frac{3\pi}{2}$ is pointing downward along the negative y-axis, which would give us $y = -3$, not 3. So θ has to be in the range where the point is on the positive y-axis, which is $\frac{\pi}{2}$.

Just to make sure, let's think about the unit circle. At $\theta = 0$, we're on the positive x-axis; at $\theta = \frac{\pi}{2}$, positive y-axis; at $\theta = \pi$, negative x-axis; at $\theta = \frac{3\pi}{2}$, negative y-axis. So yeah, $\theta = \frac{\pi}{2}$ is correct for the positive y-axis.

So, putting it all together, r is 3 and θ is $\frac{\pi}{2}$. Therefore, the polar coordinates should be $(3, \frac{\pi}{2})$. Let me just write that out to make sure I didn't make any mistakes.

Wait, another thought: sometimes people use different ranges for θ, like from $-\pi$ to π. But in this case, the problem specifies that θ should be between 0 and 2π, so $\frac{\pi}{2}$ is within that range. So, I don't need to adjust θ or anything.

Just to recap: the point is $(0, 3)$ in rectangular, so $x = 0$, $y = 3$. Calculating r: $\sqrt{0^2 + 3^2} = 3$. Calculating θ: since $x = 0$ and y is positive, θ is $\frac{\pi}{2}$. So, $(3, \frac{\pi}{2})$ is the polar coordinate.

I think that's solid. Let me visualize it too. If I plot $(0, 3)$ on the rectangular coordinate system, it's on the y-axis, three units up. In polar coordinates, that's a radius of 3 and an angle of $\frac{\pi}{2}$ from the positive x-axis. Yep, that lines up perfectly.

Just to add another check, let me compute the sine and cosine with $\theta = \frac{\pi}{2}$. $\cos(\frac{\pi}{2})$ is 0, so $x = 3 * 0 = 0$, which matches. $\sin(\frac{\pi}{2})$ is 1, so $y = 3 * 1 = 3$, which also matches. Perfect, no discrepancies there.

So, I think I'm confident that the polar coordinates are $(3, \frac{\pi}{2})$. I don't see any mistakes in my reasoning, and all the checks seem to confirm that this is the correct answer.

Final Answer: The polar coordinates are $\boxed{(3, \frac{\pi}{2})}$. ✓

Figure 9: Ada-R1 can infer answers to simple problems with a small amount of CoT, whereas the R1 model requires a long CoT.

https://doi.org/10.52202/085713-1986

Question: Define $p = \sum_{k=1}^{\infty} \frac{1}{k^2}$ and $q = \sum_{k=1}^{\infty} \frac{1}{k^3}$. Find a way to write $\sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \frac{1}{(j+k)^3}$ in terms of p and q. Let's think step by step and output the final answer within boxed{}. Ground Truth: $p - q$

Ada-R1: Okay, so I have this problem where I need to express a double sum in terms of two known series, p and q. Let me first write down what's given. We have: $p = \sum_{k=1}^{\infty} \frac{1}{k^2}$ and $q = \sum_{k=1}^{\infty} \frac{1}{k^3}$. And the expression we need to evaluate is $\sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \frac{1}{(j+k)^3}$. Hmm, so this is a double sum over j and k starting from 1 to infinity, and each term is 1 divided by $(j + k)$ cubed. I need to find a way to write this in terms of p and q. Let me think about how to approach this. Maybe I can change variables to simplify the expression. If I let $n = j + k$, then for each fixed n, j can range from 1 to n-1, and k would be n - j. So, for each n, the number of terms where j + k = n is n - 1. So, maybe I can rewrite the double sum as a single sum over n, and for each n, sum over j from 1 to n-1. Let me write that down: $\sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \frac{1}{(j+k)^3} = \sum_{n=2}^{\infty} \sum_{j=1}^{n-1} \frac{1}{n^3}$. Yes, that seems right. Because when j + k = n, j can go from 1 to n-1, and for each j, k is determined as n - j. So, the inner sum is just adding $\frac{1}{n^3}$ (n-1) times. So, the double sum becomes: $\sum_{n=2}^{\infty} \frac{n-1}{n^3}$. Simplify that expression: $\sum_{n=2}^{\infty} \frac{n-1}{n^3} = \sum_{n=2}^{\infty} \left(\frac{1}{n^2} - \frac{1}{n^3}\right)$. Yes, because $\frac{n-1}{n^3} = \frac{n}{n^3} - \frac{1}{n^3} = \frac{1}{n^2} - \frac{1}{n^3}$. So now, the sum can be split into two separate sums: $\sum_{n=2}^{\infty} \frac{1}{n^2} - \sum_{n=2}^{\infty} \frac{1}{n^3}$. But notice that these sums are just p and q minus their first term. Let's write that: The first sum is: $\sum_{n=2}^{\infty} \frac{1}{n^2} = p - \frac{1}{1^2} = p - 1$. Similarly, the second sum is: $\sum_{n=2}^{\infty} \frac{1}{n^3} = q - \frac{1}{1^3} = q - 1$. Therefore, the original ...
[OMITTED]
Thus, the original double sum becomes: $(p-1)-(q-1) = p-1-q+1 = p-q$. Therefore, the final answer is: $\boxed{p-q}$. ✓

R1: Okay, so I have this problem where I need to express a double sum in terms of two known series, p and q. Let me first write down what's given. We have: $p = \sum_{k=1}^{\infty} \frac{1}{k^2}$ and $q = \sum_{k=1}^{\infty} \frac{1}{k^3}$. And the expression we need to evaluate is $\sum_{j=1}^{\infty} \sum_{k=1}^{\infty} \frac{1}{(j+k)^3}$. Hmm, so this is a double infinite sum where both j and k start from 1 and go to infinity, and each term is 1 over $(j + k)$ cubed. I need to find a way to write this in terms of p and q. Let me think about how to approach this. Maybe I can find a substitution or change variables to simplify the expression. Since the denominator is $(j + k)^3$, maybe if I let $n = j + k$, then I can rewrite the sum in terms of n and another variable. Let's try that. Let me set $n = j + k$. Then, for each fixed n, j can range from 1 to n - 1, because k would be n - j, which also needs to be at least 1. So, for each n, j goes from 1 to n - 1, and k = n - j. So, substituting, the double sum becomes: $\sum_{n=2}^{\infty} \sum_{j=1}^{n-1} \frac{1}{n^3}$. Wait, that seems promising. Because now for each n, we have a term $\frac{1}{n^3}$ multiplied by the number of j's, which is (n - 1). So, the inner sum is just $\frac{n-1}{n^3}$. Therefore, the entire expression becomes: $\sum_{n=2}^{\infty} \frac{n-1}{n^3}$. Simplify that: $\sum_{n=2}^{\infty} \left(\frac{1}{n^2} - \frac{1}{n^3}\right)$. Which is: $\sum_{n=2}^{\infty} \frac{1}{n^2} - \sum_{n=2}^{\infty} \frac{1}{n^3}$. But I know that: $\sum_{n=1}^{\infty} \frac{1}{n^2} = p$ and $\sum_{n=1}^{\infty} \frac{1}{n^3} = q$. So, subtracting the first term from both series: $\sum_{n=2}^{\infty} \frac{1}{n^2} = p - 1$ and $\sum_{n=2}^{\infty} \frac{1}{n^3} = q - 1$. Therefore, substituting back into the expression: $(p - 1) - (q - 1) = p - 1 - q + 1 = p - q$. Wait, so the entire double sum simplifies to p - q? Let me verify my steps to make sure I didn't make a mistake. 1. I started by changing variables, setting n = j + k. Then for each n, j goes from 1 to n - 1, so k = n - j also goes from 1 to n - 1. That seems correct. 2. Then, I rewrote the double sum as a single sum over n, with each term being $\frac{n-1}{n^3}$. That makes sense because for each n, there are (n - 1) terms contributing $\frac{1}{n^3}$ each. 3. Then, I split the fraction into $\frac{1}{n^2} - \frac{1}{n^3}$. That's a valid step because $\frac{n-1}{n^3} = \frac{n}{n^3} - \frac{1}{n^3} = \frac{1}{n^2} - \frac{1}{n^3}$. 4. Then, I separated the sums into two infinite series starting from n = 2, which is correct because the original sum starts from n = 2. 5. Finally, I expressed the sums in terms of p and q by subtracting the n=1 term from each series. That's correct because the original p and q start from n=1. So, putting it all together...
[OMITTED]
Therefore, the final answer is: $\boxed{p-q}$ ✓

Figure 10: Ada-R1 can infer answers to difficult problems with relatively few long CoT, whereas the R1 model requires longer CoT.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide our contributions and scope both in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitation in the Appendix of this paper.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

https://doi.org/10.52202/085713-1986

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Yes, we provide all necessary information to reproduce the experimental results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide code and data.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The training and test details are described in the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The paper does not report error bars, confidence intervals, or standard deviations for the experimental results. But our models are evaluated with multiple benchmarks.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

https://doi.org/10.52202/085713-1986

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide this information in the paper in Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research follows the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes. All assets are properly credited and used under their respective licenses.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

https://doi.org/10.52202/085713-1986

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not provide new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: This research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

https://doi.org/10.52202/085713-1986