Aware First, Think Less: Dynamic Boundary Self-Awareness Drives Extreme Reasoning Efficiency in Large Language Models

先觉察，少思考：动态边界自觉性驱动大语言模型的极致推理效率

https://arxiv.org/abs/**2508**.11582

Harbin Institute of Technology；Central South University

汇报人：文胜涛　　时间：2025-9-1

1. 研究动机

随着LLM在复杂推理任务中的应用，尤其是**CoT**的引入，虽然其在推理准确性上取得了显著进展，但往往会产生**大量冗余的推理步骤（冗余token）**，导致计算效率低下，并且在实时应用中会造成显著的延迟。

现有方法的不足?

当前的方法往往依赖于**人工设计的静态困难度先验和目标长度**，而忽视了每个LLM在训练过程中推理边界的动态变化，可能导致简单的问题需要更多的思考，难的问题反而推理过程短。

目的： 本文提出了**动态推理边界自觉框架（DR. SAF）**，它能让模型根据任务的复杂性和自身的推理能力，**动态调整推理的深度。**

2. Background and Related Works

核心背景问题：长链推理往往带来大量冗余token，计算效率低，延迟高。

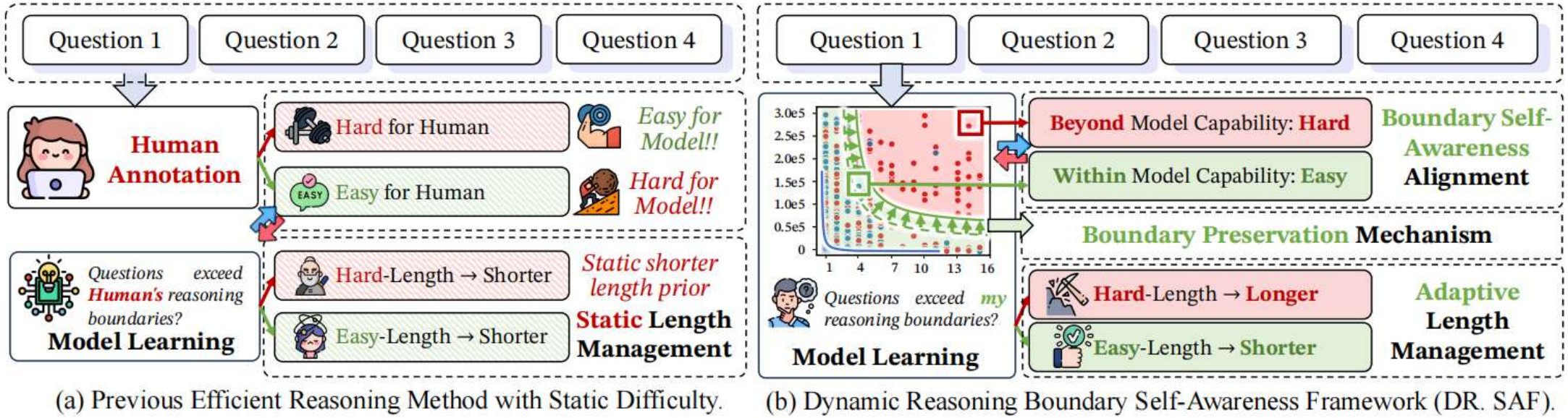

Figure 1: Traditional efficient reasoning training methods (a) primarily determine the difficulty of questions based on human-defined priors, while our dynamic reasoning boundary self-awareness framework (b) judged the difficulty of questions based on model self-awared reasoning boundary.

2. Background and Related Works

基于置信度的动态推理

使用**输出概率**或**结构置信度**决定是否继续推理，典型例子有动态提前退出、置信度触发器等。

在部分任务上节省计算，但依赖外部信号，泛化有限。

基于人工先验的长度控制

通过**人工设定难度和token预算**（如AdaptThink, DAST），在简单问题上缩短推理，复杂问题上延长推理。

例如Ling等人在训练过程中嵌入长度惩罚，以根据人类先验平衡简洁性和推理深度等

长度惩罚与自适应奖励

设置一个**惩罚loss**，**难度感知**——>**分段奖励。**通过将CoT长度与问题难度相关联，一定程度上减少延迟，但仍然依赖固定或人为设定的难度基准。

Traditional efficiency-target methods predominantly **focus on optimizing reasoning paths based on fixed or human-defined difficulty levels**. In contrast, **DR.SAF** introduces a **self-aware system capable of dynamically adjusting the depth of reasoning according to the model's internal capabilities and the real-time complexity of the task**. This approach enhances both efficiency and accuracy.

3. Method

Completely Feasible Reasoning Boundary
Partily Feasible Reasoning Boundary

DR.SAF由三大板块构成：

1. 边界自觉对齐（Boundary Self-Awareness Alignment）： 使模型能够评估问题的难度。

LLM自己判断问题的难度，自认为好解决——>**CFRB**，自认为不好解决——>**PFRB**

易
CFRB
PFRB
难

$$R_{\text{Aware}}(y|x) = \begin{cases} +\alpha_1, & \text{if } \text{Acc}(\mathcal{Y}|x) \geq 90\% \wedge \text{Aware}(x) < \text{CFRB}; \\ +\alpha_1, & \text{if } \text{Acc}(\mathcal{Y}|x) < 90\% \wedge \text{CFRB} \leq \text{Aware}(x) \leq \text{PFRB}; \\ -\alpha_2, & \text{otherwise}, \end{cases}$$

2. 自适应长度管理（Adaptive Length Management）

已经掌握的CFRB问题，模型会得到**压缩奖励**，否则给予**扩展奖励**。

$$R_{\text{Len}}(y|x) = \begin{cases} \delta_{\text{comp}} & \text{if } \text{Acc}(\mathcal{Y}|x) > 90\% \wedge \ell \leq \bar{\ell}_{\text{CFRB}} \\ \delta_{\text{ext}} & \text{if } \text{Acc}(\mathcal{Y}|x) < 10\% \wedge \ell > \bar{\ell}_{\text{CFRB}} \\ 0 & \text{otherwise} \end{cases}$$

3. 边界保持机制（Boundary Preservation Mechanism）

通过调整优势来稳定优化过程。 $REff(yi|x) = RAcc(yi|x) + RLen(yi|x) + RAware(yi|x)$

3. Method

DR.SAF由三大板块构成：

3. 边界保持机制（Boundary Preservation Mechanism）

边界崩溃。当模型过度压缩推理路径时，可能会导致**正确答案的优势（奖励）变得低于零**，从而破坏了推理的有效性。为了避免这种情况，边界保持机制强制所有正确的响应都保持非负优势，从而稳定推理过程，避免推理过程的崩溃。

$$REff(yi|x) = RAcc(yi|x) + RLen(yi|x) + RAware(yi|x)$$

→ **可能为负**

GRPO每个输出的优势是：

$$A(y|x) = \frac{R_{\text{Eff}}(y|x) - \mu_R}{\sigma_R + \epsilon}$$

μ_R 是组内平均奖励

BPM通过**截断均值**的方式对齐进行限制：

$$\mu_R^{trunc} = \max(\mu_R, \min_{y_i \in C} R_{Eff}(y_i|x))$$

取两者之间的最大值。

3. Method

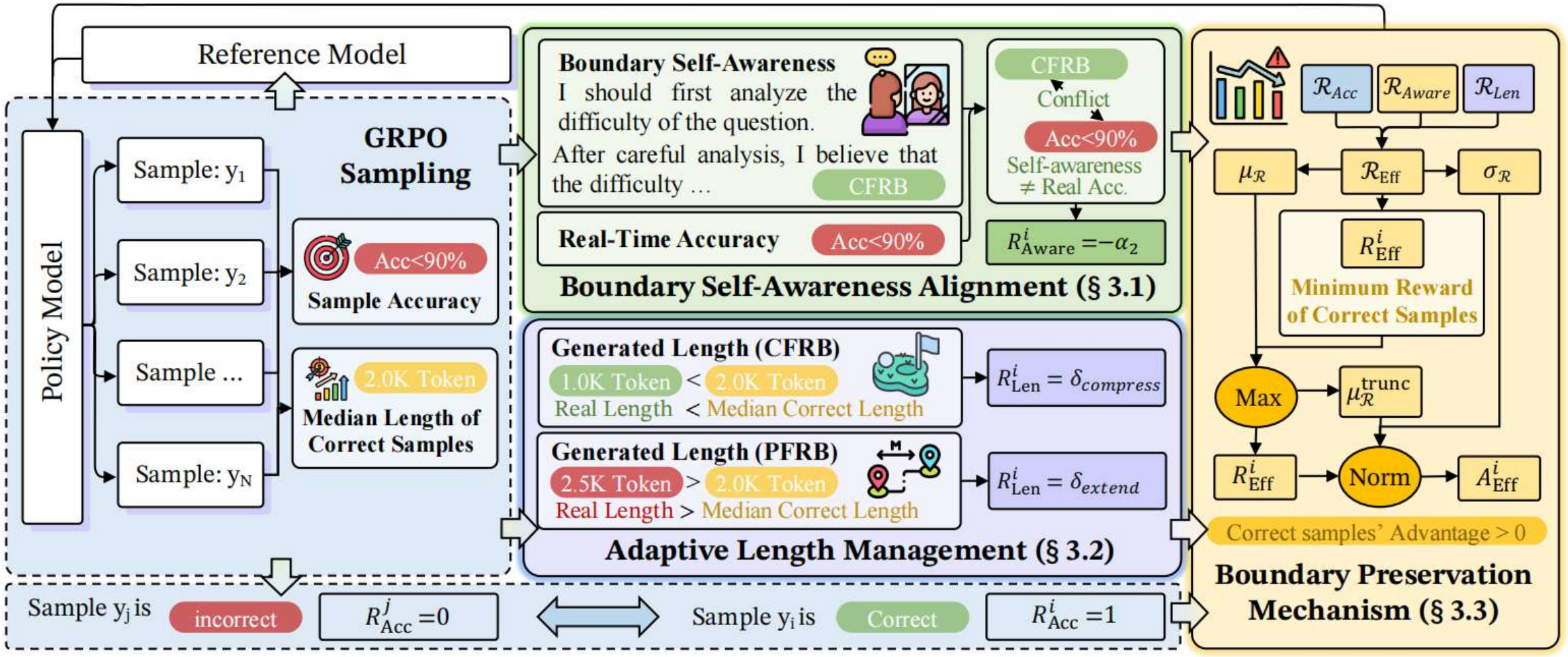

Figure 2: Main pipeline of Dynamic Reasoning-Boundary Self-Awareness Framework (DR. SAF), including **Boundary Self-Awareness Alignment (BSA)**, **Adaptive Length Management (ALM),** and **Boundary Preservation Mechanism (BPM).**

3. Method

给定问题x：一个班有 36 个学生，分成 6 个小组，每组人数相等。每组有多少人？

模型（策略 π_θ）在训练时，会针对同一个问题 x 采样出 $k = 4$ 个答案，设为：

① 36 ÷ 6 = 6. 答案：6。 （**正确，推理短**）

② 先把36分成两半=18，再把18分成3组=6。答案：6。 （**正确，但推理更长**）

③ 36 ÷ 3 = 12，每组12人。 （**错误**）

④ 36个学生，可能不能平均分。答案：5或6。 （**错误，逻辑混乱**）

3. Method

一、先计算 R_{ACC}

答案1： 奖励 1

答案2： 奖励 1

答案3： 奖励 0

答案4： 奖励 0

给定问题x：一个班有 36 个学生，分成 6 个小组，每组人数相等。每组有多少人？

模型（策略 π_θ）在训练时，会针对同一个问题 x 采样出 $k=4$ 个答案，比如：

① $36 \div 6 = 6$. 答案：6。 **（正确，推理短）**

② 先把36分成两半=18，再把18分成3组=6。答案：6。 **（正确，但推理更长）**

③ $36 \div 3 = 12$，每组12人。 **（错误）**

④ 36个学生，可能不能平均分。答案：5或6。 **（错误，逻辑混乱）**

二、计算边界自觉奖励 R_{Aware} ——BSA

模型先自评：这题是简单算术 → 标为 CFRB（完全可行边界）

检验：确实 Acc > 90%（假设大部分答对了）。

$$R_{\text{Aware}}(y|x) = \begin{cases} +\alpha_1, & \text{if } \text{Acc}(\mathcal{Y}|x) \geq 90\% \wedge \text{Aware}(x) < \text{CFRB}; \\ +\alpha_1, & \text{if } \text{Acc}(\mathcal{Y}|x) < 90\% \wedge \text{CFRB} \leq \text{Aware}(x) \leq \text{PFRB}; \\ -\alpha_2, & \text{otherwise}, \end{cases}$$

所以正确样本（答案1、2）得到 正奖励 α；如果模型把这种题错判成“难题”或答错，就会扣分。

三、计算长度奖励 R_{Len}——ALM

正确答案集合 C={1, 2}。

正确样本的平均长度：

答案1（短，8 token 左右）

答案2（长，20 token 左右）

→ 中位数 $\bar{\ell}_{CFRB} \approx 14$。

答案1：长度短于 14 → 获得压缩奖励 δ_{comp}。

答案2：长度大于 14 → 没有额外奖励（因为已经正确但太啰嗦）。

错误答案（3,4）：不在 CFRB，奖励 0。

3. Method

四、计算总奖励

答案1：1（Acc） + α_1（Aware） + δ_{comp}（Len） → 高分

答案2：1（Acc） + α_1（Aware） → 次高分

答案3：0 → 低分

答案4：0 → 低分

给定问题x：一个班有 36 个学生，分成 6 个小组，每组人数相等。每组有多少人？

模型（策略 π_θ）在训练时，会针对同一个问题 x 采样出 $k=4$ 个答案，比如：

① 36 ÷ 6 = 6. 答案：6。 **（正确，推理短）**

② 先把36分成两半=18，再把18分成3组=6。答案：6。 **（正确，但推理更长）**

③ 36 ÷ 3 = 12，每组12人。 **（错误）**

④ 36个学生，可能不能平均分。答案：5或6。 **（错误，逻辑混乱）**

五、边界保持——BPM

可能出现的问题：如果压缩奖励太大，正确答案2（虽然长，但正确）可能优势变负。

BPM 会把组均值基线抬高，确保正确答案（1,2）的优势 ≥ 0。

→ 答案1 有最高优势，答案2 保持非负，答案3/4 优势为负。

$$REff(yi|x) = RAcc(yi|x) + RLen(yi|x) + RAware(yi|x)$$

$$A(y|x) = \frac{R_{\text{Eff}}(y|x) - \mu_R}{\sigma_R + \epsilon}$$

$$\mu_R^{trunc} = \max(\mu_R, \min_{y_i \in C} R_{Eff}(y_i|x))$$

六、GRPO 更新

把每个样本的奖励转成 组内归一化优势：

A_1：正，大

A_2：正，中

A_3、A_4：负

用这些优势加权 log 概率更新模型 → **答案1 的模式会被放大**，答案2 保持可行但不鼓励啰嗦，答案3/4 被压制。

4. Experimental Setup

训练集：DeepMath103K随机抽取了 5,000 个实例作为训练集

验证集：AIME24、GSM8K、Math-500、AMC23、OlympiadBench、AIME25

LLM基座：R1-distill-Qwen-2.5-7B 和 R1-distill-Qwen-3-8B

评价指标

- Accuracy (ACC in %)
- average response token length (LEN)
- token efficiency (EFF)

Baseline

- **Prompting Strategies：** Dynasor-CoT、DEER、ThinkSwitcher
- **Offline Strategies：** OverThink、Spirit、ConCISE-SimPO、DAST、AdaptThink
- **Online Strategies：** Length-Penalty FEDH

4. Experimental Results

Model Name	GSM8K			MATH500			AIME24			AMC			OlymBench			AIME25		
	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF	ACC	LEN	EFF
Qwen2.5-7B-Ins	90.9	279	32.58	74.2	567	13.09	12.0	1016	1.18	47.5	801	5.93	39.2	827	4.74	7.6	1240	0.61
Qwen2.5-7B-Math	93.2	439	21.23	63.4	740	8.57	19.0	1429	1.33	62.5	1022	6.12	31.5	1037	3.04	4.0	2562	0.16
Qwen2.5-7B-Math-Ins	95.2	323	29.47	81.4	670	12.15	10.3	1363	0.76	60.0	1029	5.83	38.9	1027	3.79	9.3	2087	0.45
R1-Distill-Qwen2.5-7B	92.4	1833	5.04	90.8	3854	2.36	49.2	10200	0.48	90.0	6476	1.39	66.1	7789	0.85	35.0	10518	0.33
+ ThinkSwitcher	**92.5**	1389	6.66	91.3	3495	2.61	48.3	7936	0.61	–	–	–	57.0	5147	1.11	37.5	6955	0.54
+ Dynasor-CoT	89.6	1285	6.97	89.4	2661	3.36	46.7	12695	0.37	85.0	5980	1.42	–	–	–	–	–	–
+ DEER	90.6	917	9.88	89.8	2143	4.19	49.2	9839	0.50	85.0	4451	1.91	–	–	–	–	–	–
+ OverThink	91.4	879	10.39	**92.9**	2405	3.86	50.0	9603	0.52	–	–	–	–	–	–	–	–	–
+ Spirit	87.2	687	12.68	90.8	1765	5.14	38.3	6926	0.55	–	–	–	–	–	–	–	–	–
+ ConCISE-SimPO	92.1	715	12.88	91.0	1945	4.68	48.3	7745	0.62	–	–	–	–	–	–	–	–	–
+ DAST	86.7	459	18.89	89.6	2162	4.14	45.6	7578	0.60	–	–	–	–	–	–	–	–	–
+ AdaptThink	91.0	309	29.45	92.0	1875	4.91	**55.6**	8599	0.65	85.0*	4265*	1.99*	58.4*	5988*	0.98*	**38.3***	10380*	0.37*
+ Length-Penalty	87.2	263	33.16	89.1	2121	4.20	51.9	7464	0.70	82.5	4411	1.87	**59.8**	4919	1.22	33.3	8902	0.37
+ FEDH	90.1	218	41.33	88.5	1306	6.50	42.3	7242	0.58	–	–	–	–	–	–	–	–	–
+ **DR. SAF**	88.1	**162**	**54.38**	88.3	**1061**	**8.32**	50.6	**6288**	**0.80**	**90.0**	**3096**	**2.91**	59.4	**3259**	**1.82**	38.2	**6764**	**0.56**
R1-Distill-Qwen3-8B	94.2	2135	4.41	90.6	7051	1.28	**67.9**	20155	0.34	83.5	11931	0.70	60.1	12895	0.47	**62.9**	20992	0.30
+ FEDH*	**94.4**	2014	4.69	92.6	6761	1.37	61.3	13463	0.42	94.7	11928	0.79	63.5	12353	0.51	46.7	14730	0.32
+ Length-Penalty*	93.3	604	15.45	92.4	2581	3.58	63.7	12303	0.52	89.2	6166	1.45	68.4	7383	0.93	54.7	12446	0.44
+ **DR. SAF**	92.3	**521**	**17.72**	**93.3**	**2168**	**4.30**	66.0	**9807**	**0.67**	**95.6**	**4003**	**2.39**	**71.3**	**5766**	**1.24**	57.9	**10692**	**0.54**

Prompt 策略

Offline 策略

Online 策略

1. **offline策略**尽管在**ACC**上有较高的表现，但相比于**online策略**，它们的**token Efficiency**较低；
2. DR. SAF 在**token Efficiency**方面表现出色，且**ACC**几乎没有下降；
3. DR. SAF 显著提高了 LLM 相较于静态困难度推理的**token Efficiency**；
4. DR. SAF 在更强大的 LLM 上显示出显著的性能提升。

4. More Analyses

Analysis 1: DR. SAF 在所有基准上都能实现与传统指令模型相媲美，甚至更优的token Efficiency。

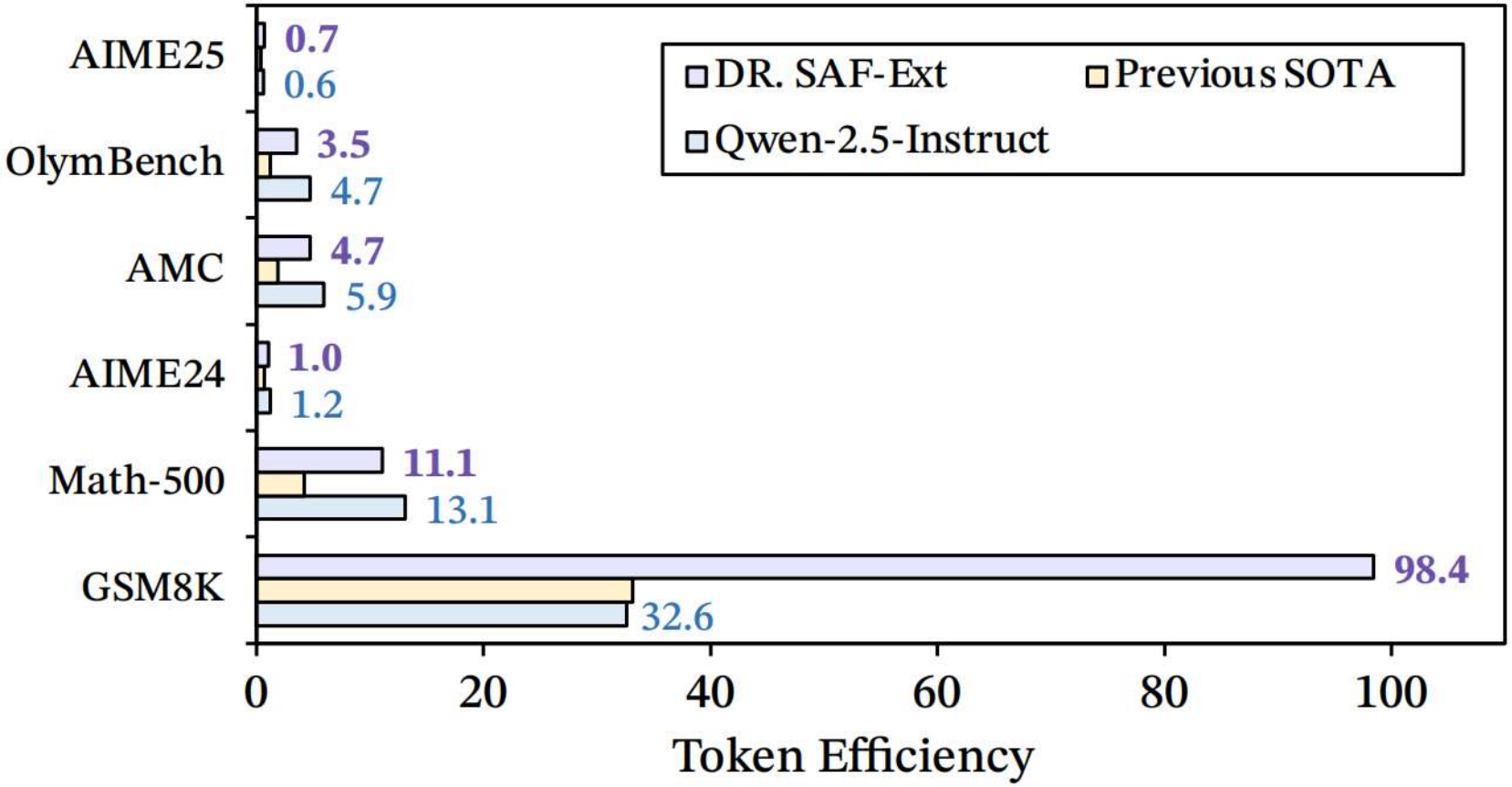

Figure 4: Comparing the extreme efficiency of **DR. SAF (DR. SAF-Ext)** with **traditional instruction models** and current **SOTA reasoning efficient techniques**.

4. More Analyses

Analysis 2: DR. SAF 显著加快了压缩训练的速度。

Analysis 3: DR. SAF 在压缩过程中提升了性能，而基于长度惩罚的方法(FEDH)则在压缩时导致性能下降（图5a、5c

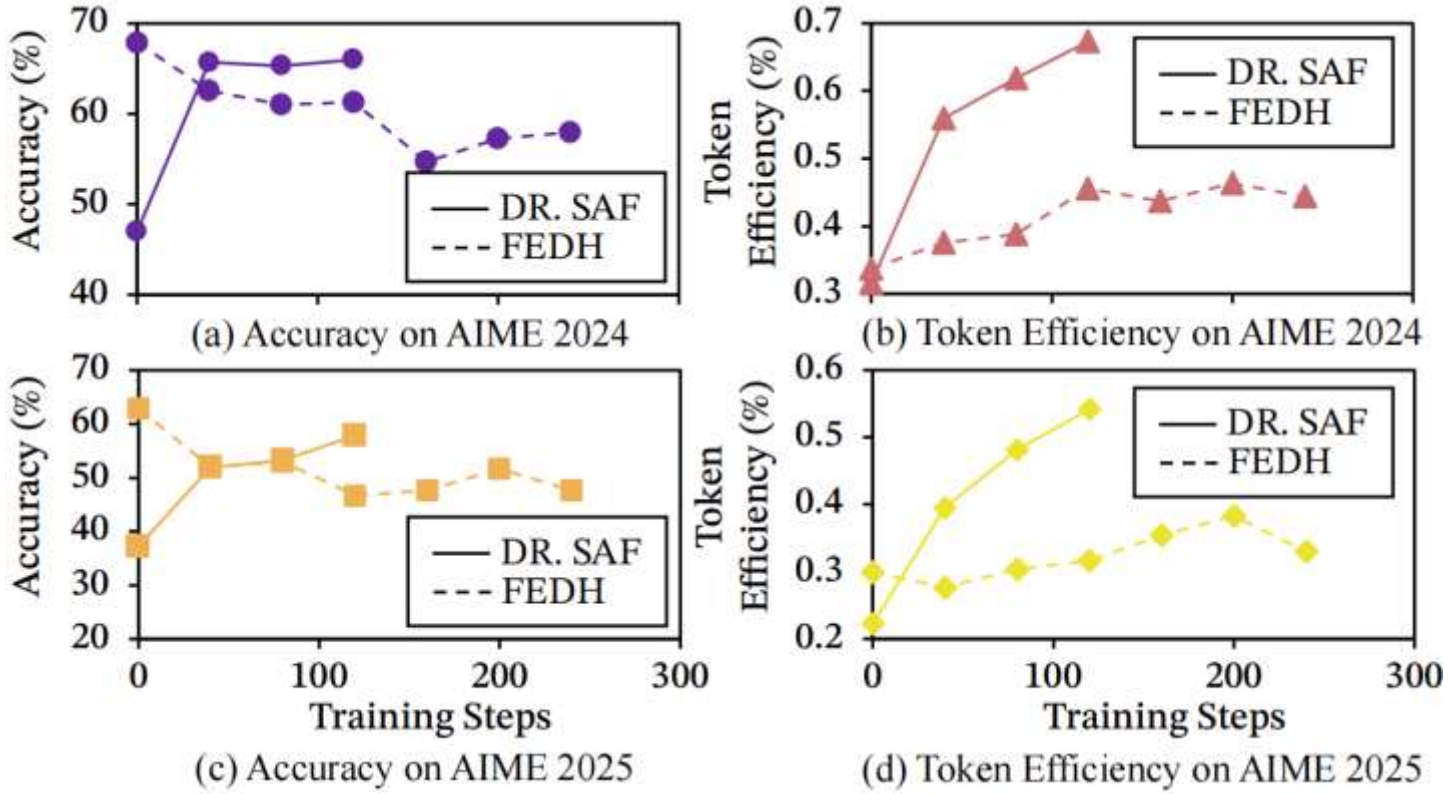

Figure 5: Training efficiency comparison of **DR. SAF** vs. **FEDH** on R1-Distill-Qwen-3-8B

5. Ablation experiments

边界自适应对齐BSA 是否增强了模型的边界自觉性，从而提高了令牌效率?

Model Name	ACC_{AVG}	LEN_{AVG}	EFF_{AVG}
DR. SAF	75.28	2773.2	13.65
w/o BSA	74.98	3219.9	8.04
w/o ALM	67.54	2105.1	11.96
w/o BPM	67.87	2543.6	13.65

Answer1: Boundary Self-Awareness Alignment is crucial for DR. SAF efficiency. We assess the effectiveness of the Boundary Self-Awareness Mechanism by ablating it from the DR. SAF. As shown in Table 2, token efficiency decreases by more than 40% without this component. Further, Figure 3 reveals that, during training, the model's predicted task difficulty progressively aligns with its actual reasoning accuracy. Notably, the models with the highest and second-highest alignment scores also achieve the highest and second-highest levels of token efficiency, respectively.

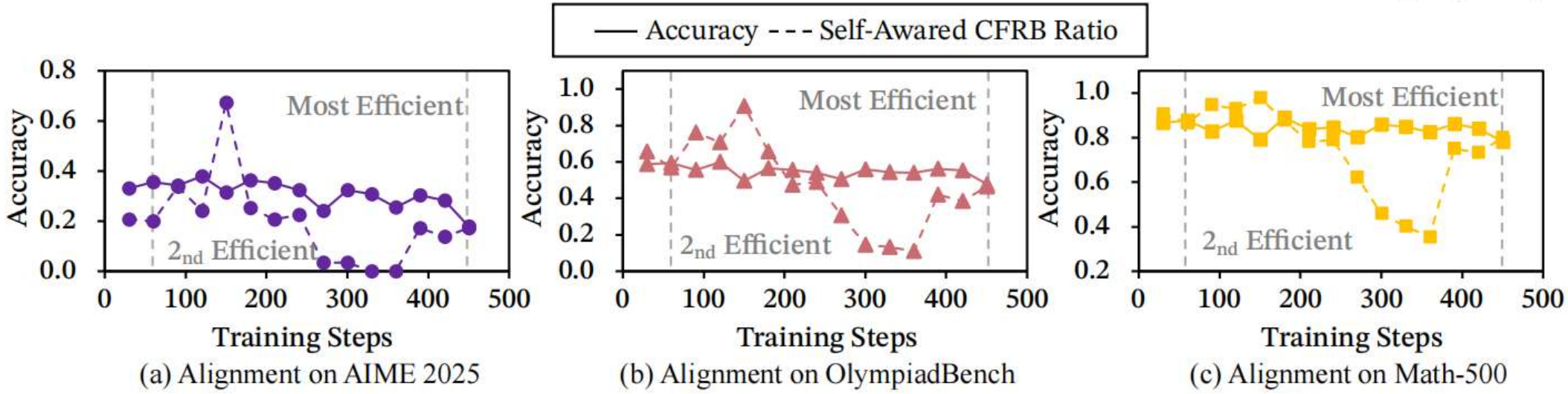

Figure 3: Training trajectory of BSA, shown as the predicted CFRB ratio plotted against the training steps.

5. Ablation experiments

自适应长度管理ALM 是否通过自适应地控制令牌长度同时提高了准确性和效率?

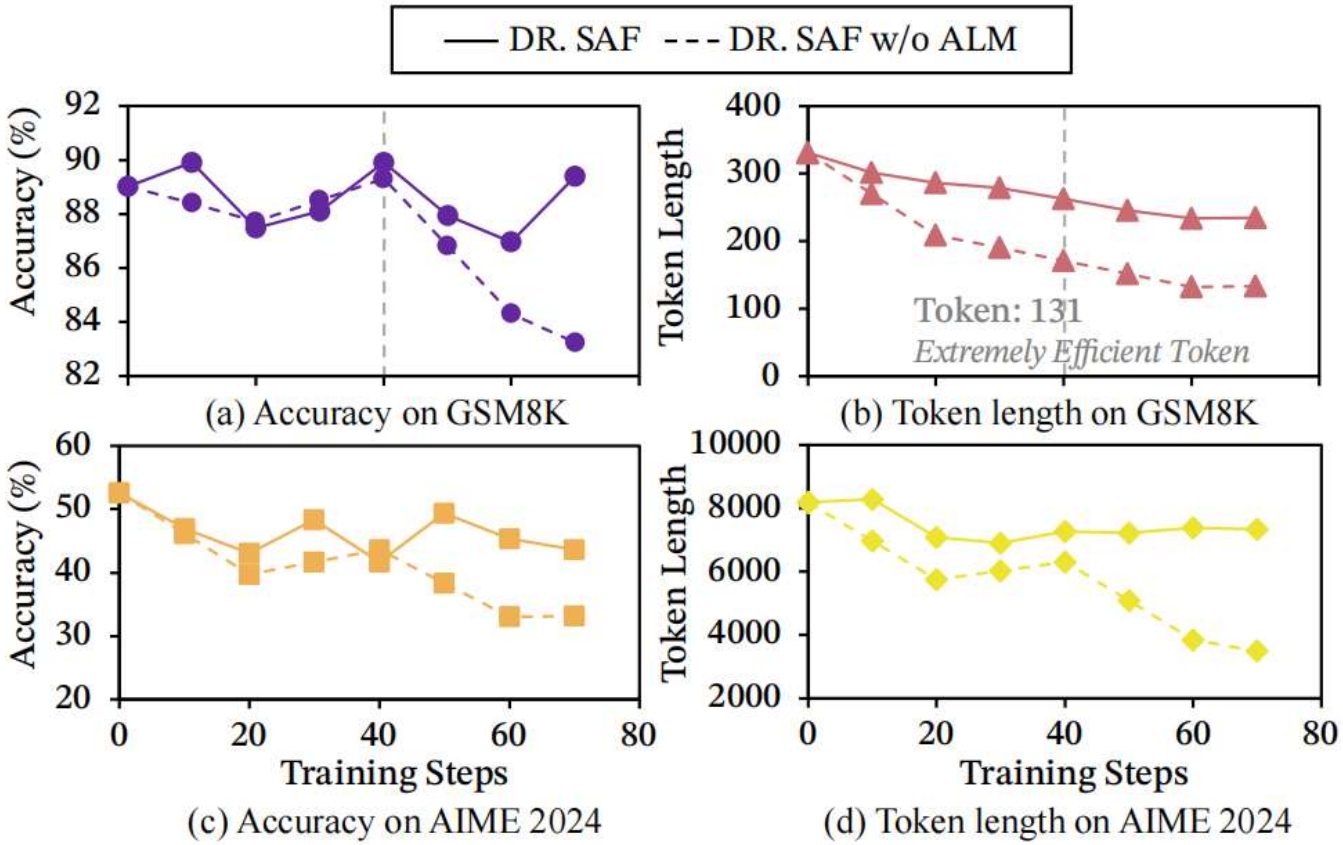

Figure 7: Trends in accuracy and response length during training with Adaptive Length Management (ALM).

5. Ablation experiments

边界保持机制BPM 是否防止了推理边界在压缩训练中的崩溃，从而保持了模型性能?

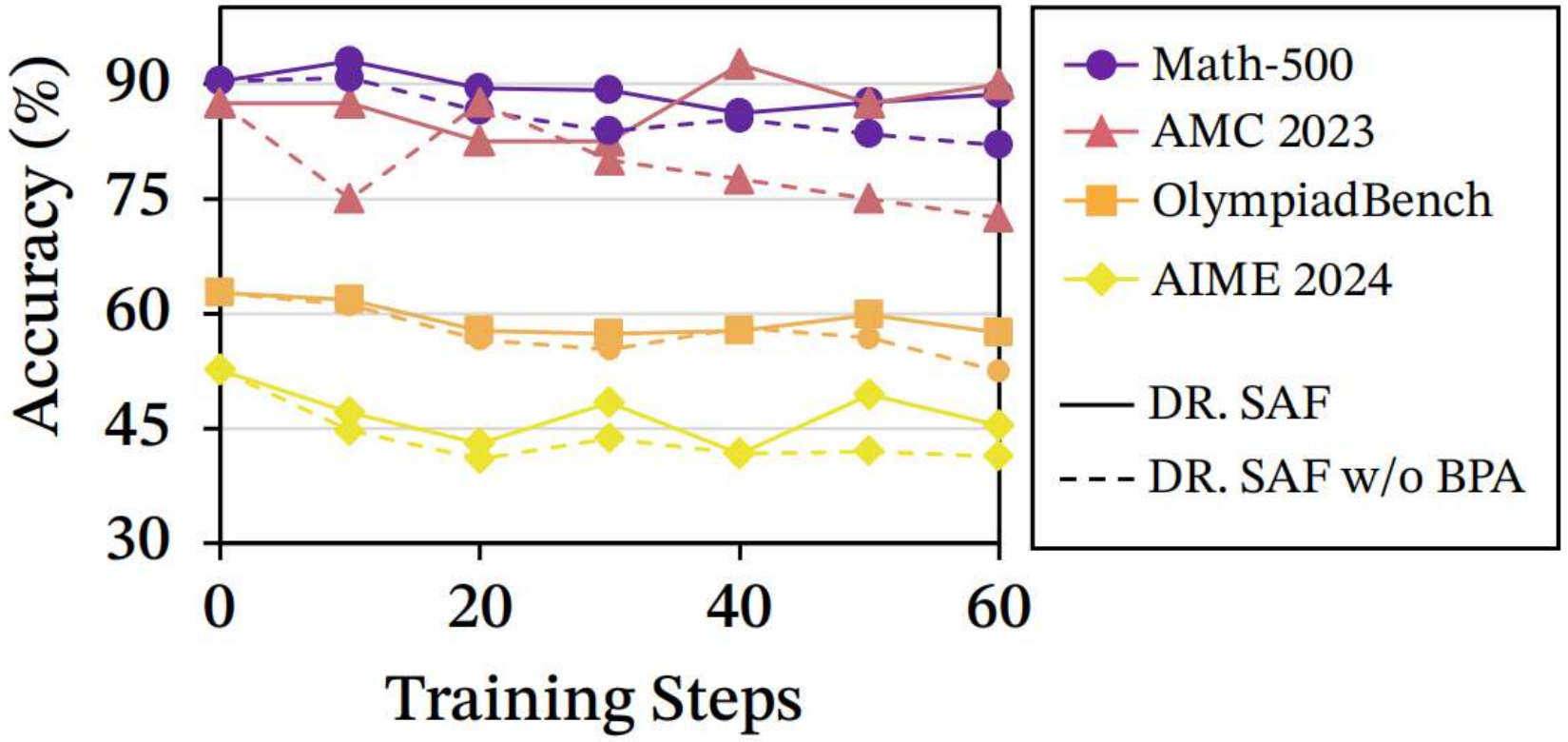

Figure 8: Accuracy trend produced during training by the boundary-preservation mechanism (BSM).

Thanks for Your Watching !

汇报人：文胜涛　　时间：2025-7-08