Adaptive Group Policy Optimization: Towards Stable Training and Token-Efficient Reasoning

Anonymous ACL submission

Abstract

Since DeepSeek-R1 popularized, Group Relative Policy Optimization (GRPO) has become the core part of training Reasoning LLMs. However, we find some deficiency that influences RL stability and inference efficiency, like zero-variance in advantage estimation. Thus, we propose Adaptive Group Policy Optimization (AGPO) which contains a simple but effective modification: a revised objective function to mitigate training fluctuation and zero advantage. The experiments demonstrate our method achieves more stable training and superior performance with significantly fewer tokens in reasoning steps.

1 Introduction

Large Language Models (LLMs)(Bommasani et al., 2021; Wei et al., 2022; Zhao et al., 2023) have achieved impressive performance through extensive pre-training and post-training processes. However, effectively generating desired model responses often necessitates aligning outputs with specific downstream tasks and human preferences(Wang et al., 2023; Wolf et al., 2023).

For alignment challenges, reinforcement learning from human feedback (RLHF)(Bai et al., 2022; Kaufmann et al., 2023) is introduced as a prominent post-training strategy, adopted by notable LLMs including GPT-4, Claude, Gemini, and DeepSeek. They have explored various optimization techniques such as Proximal Policy Optimization (PPO) (Schulman et al., 2017) and Direct Preference Optimization (DPO) (Rafailov et al., 2023). Recently, to significantly reduce computational and memory overhead associated with PPO, DeepSeek eliminated the value model and proposed Group Relative Policy Optimization (GRPO)(Guo et al., 2025), which achieved high computational efficiency and excellent reasoning performance, surpassing other open-source models ranging from 7B to 70B.

Despite the demonstrated success of GRPO, it introduces challenges that can affect stable training and inference efficiency.

(1) Confusing Training Signal: Negative losses happen in RL training, but in this scenario it is not always beneficial. Higher group accuracy may have lower advantage thus influencing loss estimation. Besides, when all rewards within a group are identical, the normalized advantage approaches 0, causing the loss signal to vanish, which potentially stalling training.

(2) Inefficient CoT Length: Since GRPO lacks mechanisms to discourage excessively long chain-of-thought (CoT), models tend to produce overly verbose explanations. A refined approach that rewards concise and effective reasoning is essential to improve token efficiency.

To address these issues, we propose an enhanced GRPO training algorithm, Adaptive Group Policy Optimization (AGPO). Our main contributions are summarized as follows:

- **Training Efficiency:** By identifying the limitations of GRPO's advantage, we introduce an adaptive loss function that addresses negative loss and zero advantage scenarios, ensuring continuous and effective learning.

- **Token Efficiency:** Our adaptive loss implicitly improves token efficiency. Compared with GRPO baselines, our approach achieves better performance with significantly fewer response tokens.

2 Background

2.1 Policy Gradient

Policy gradient method is one of the most fundamental RL algorithm that directly model and optimize the policy. For any differentiable policy, the

policy gradient is :

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) A_t\right] \quad (1)$$

Where advantage A_t is the most crucial part for policy gradient method, which defines how much better a specific action a_t is, compared to average action given a state s_t.

2.2 Proximal Policy Optimization (PPO)

PPO (Schulman et al., 2017) is one of the policy gradient methods which uses clipped surrogate objective for policy optimization. Specifically, it maximize the following objective:

$$J_{\text{PPO}}(\theta) = \mathbb{E}_{\pi_{\theta_{\text{old}}}}\left[\min\left(r_t A_t, \text{clip}(r_t, 1-\epsilon, 1+\epsilon) A_t\right)\right] \quad (2)$$

Where ϵ is a hyper-parameter used for tuning clipping range. A_t is the advantage, which generally will be computed by utilizing Generalized Advantage Estimation (GAE) (Schulman et al., 2015) in PPO. r_t is the probability ratio of predicting token o_t for a given question q before and after the policy update:

$$r_t(\theta) = \frac{\pi_\theta(o_t|q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t|q, o_{<t})} \quad (3)$$

2.3 Group Relative Policy Optimization (GRPO)

Compared to PPO, GRPO (Shao et al., 2024) significantly saves the training cost through eliminating the critic model in PPO. This is achieved by approximating the advantage A_i as group-normalized reward:

$$A_i = \frac{r_i - \text{mean}(\{r_1, r_2, \ldots, r_G\})}{\text{std}(\{r_1, r_2, \ldots, r_G\})} \quad (4)$$

Where $\text{mean}(\{r_1, r_2, \ldots, r_G\})$ and $\text{std}(\{r_1, r_2, \ldots, r_G\})$ denotes the within-group mean and standard deviation respectively.

With the estimated advantage and a KL divergence penalty term, GRPO generates a group of outputs $\{o_i\}_{i=1}^{G}$ based on $\pi_{\theta_{\text{old}}}$ for each question q and update π_θ with following objective:

$$J_{\text{GRPO}}(\theta) = \mathbb{E}_{(q)\sim P(Q),\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)}\left[\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)} A_i,\right.\right.\right.$$
$$\left.\left.\left.\text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\text{old}}}(o_i|q)}, 1-\epsilon, 1+\epsilon\right) A_i\right) - \beta D_{\text{KL}}(\pi_\theta \parallel \pi_{\text{ref}})\right)\right] \quad (5)$$

Where ϵ and β are hyper-parameters.

3 Adaptive Group Policy Optimization

During RL training with the GRPO algorithm, we find a useful modification to improve performance and efficiency.

3.1 Adaptive Loss

It is easy to find that if the rewards of the group are all equal, like all correct or all wrong, then the advantages become all zero. Therefore, the corresponding sample in the batch has no effect in the training. As model training progresses, this situation increases. We believe that it is helpful to mask out these positions from the loss calculation.

Another issue is entropy collapse occurred in GRPO training. We observed the same phenomenon as in other works, like DAPO (Yu et al., 2025). The entropy drops quickly leading the policy model to give up exploring. However, we consider it as the problem of objective design rather than clip ratio. Current negative loss value tends to be overconfident since a group with higher accuracy may have higher deviation of rewards which may cause lower advantage compared to other groups in batch. We propose an adaptive loss to tackle these issues:

$$L_{(q)\sim P(Q)} = \begin{cases} masked, & \text{if } \{o_i\}_{i=1}^{G} \text{ all correct or wrong} \\ max(0, -J_{\text{GRPO}}(\theta)), & \text{otherwise} \end{cases} \quad (6)$$

Usually in RL training, the model is updated through a mini batch size of training data. Therefore, the loss mask in Equation 6 takes place in the mean operation of losses of questions q in the batch. Negative loss values are also clipped for the mean loss of the batch in order to control fluctuation and maintain a suitable training level. We believe the clip avoids the model getting stuck in a local optimum.

By replacing the original objective, our method focuses on useful information in the batch and normalizes the loss for stable training, which would bring performance improvement and token efficiency.

4 Experiment

We conduct a few experiments for evaluating how our method affects the RL training of reasoning models.

4.1 Implementation Details

We use Qwen2.5-7B and Qwen2.5-14B as the base models. All experiments are conducted on our cu-

Model	MATH-500 (Pass@1)
Qwen2.5-7B	44.0
Qwen2.5-7B-GRPO	73.2
Qwen2.5-7B-AGPO (w/o loss mask)	73.4
Qwen2.5-7B-AGPO (w/o loss clip)	73.0
Qwen2.5-7B-AGPO	74.6
Qwen2.5-14B	59.8
Qwen2.5-14B-GRPO	75.4
Qwen2.5-14B-AGPO (w/o loss mask)	75.0
Qwen2.5-14B-AGPO (w/o loss clip)	75.2
Qwen2.5-14B-AGPO	77.2

Table 1: Performance of different RL techniques on MATH-500.

Model	Average Response Tokens
Qwen2.5-7B	571
Qwen2.5-7B-GRPO	699
Qwen2.5-7B-AGPO	533
Qwen2.5-14B	772
Qwen2.5-14B-GRPO	574
Qwen2.5-14B-AGPO	521

Table 2: Average response length of different RL techniques on MATH-500.

rated dataset, which is constructed by mixing data from MATH train set (Hendrycks et al., 2021) and DAPO train set (Yu et al., 2025). For MATH train set, only data where difficulty levels are greater than or equal to 3 are selected. Similarly, for DAPO train set, data are retained only if solution rates achieved by Qwen3-32B model (Yang et al., 2025) are fall between 0.5 and 0.8 inclusively. These filtering techniques are to ensure the difficulty distribution across the obtained dataset is balanced.

VeRL (Sheng et al., 2024) is utilized to perform RL training with a train batch size of 32, a PPO mini batch size of 8 and a learning rate of $1e-6$. The number of group rollout is 8. Temperature for generation is set to 1. As for reward settings, the $r_{correct}$ and r_{wrong} for accuracy reward are set to 0 and 1 respectively. Trained checkpoints that achieve best performance on MATH-500 (Lightman et al., 2023) with the metric of Pass@1 are selected for further evaluation with respect to token efficiency.

It is worth noting that KL divergence penalty is not applied for all experiments. This is based on the observation that model distribution can vary significantly compared to reference model during long CoT training. Therefore, removing KL divergence has been adopted as common practice in the domain.

4.2 Main Results

Table 1 shows the performance of different models on the benchmark. Both GRPO and AGPO acquire huge performance gains compared with the base models. As for Qwen2.5-7B experiments, we observe a clear improvement on MATH-500 from 73.2 to 74.6 at the best checkpoint. Qwen2.5-14B experiments also induce similar conclusion that the adaptive loss further refines the model by 1.8 percentage. It is obvious that both loss clip and loss mask are important for our method. If we train without the mask, the performance drops even lower than that of GRPO for all model sizes and it is same for loss clip.

Table 2 illustrates the token efficiency of different models on the benchmark. Qwen2.5-7B-AGPO only takes 533 tokens on average for solving MATH-500 problems while Qwen2.5-7B-GRPO consumes 699 tokens that is 31% higher. In terms of 14B models, our AGPO also ranks first in token efficiency which uses 521 tokens averagely.

4.3 Training Dynamics

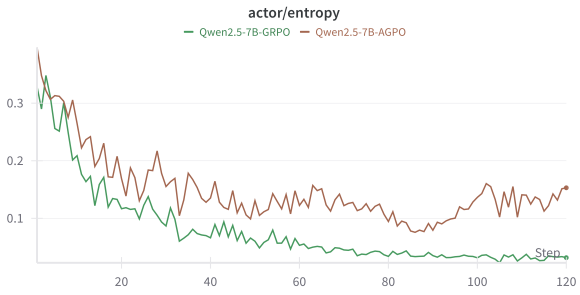

Figure 1: Actor entropy curves of GRPO and AGPO for Qwen-2.5-7B

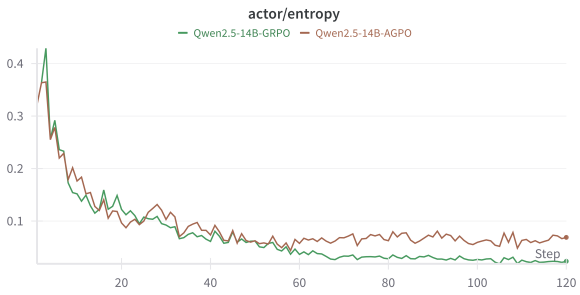

Figure 2: Actor entropy curves of GRPO and AGPO for Qwen-2.5-14B

We examine several variations of training metrics after applying AGPO.

As shown in Figure 1 and Figure 2, a significant enhancement in entropy is observed for both Qwen2.5-7B and Qwen2.5-14B actor models after the application of AGPO. This observation can be attributed to the lower loss clip operation in AGPO, where only positive training loss from below-average actions is maintained while negative training loss is overconfident and controlled in the gradient update. Consequently, the probability distribution is drifted upward asymmetrically by adequate gradient without being drifted downward meanwhile, which manifests higher measured entropy during training. The higher entropy eventually facilitates the generation of more diversified samples within the batch, which is essential for large-scale RL training.

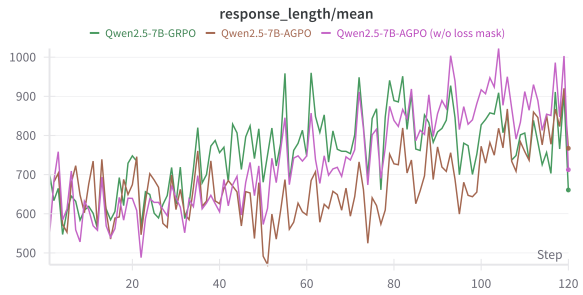

Figure 3: Response length curves of GRPO and AGPO for Qwen2.5-7B

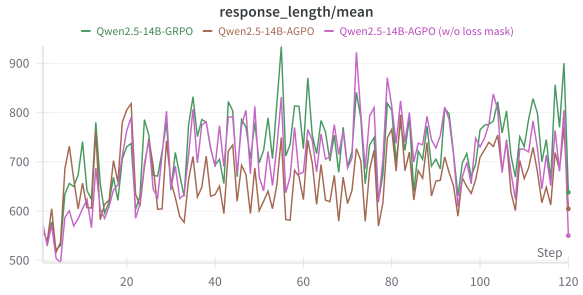

Figure 4: Response length curves of GRPO and AGPO for Qwen2.5-14B

We also find substantial reductions in response length while the training steps increase, for both models with AGPO as Figure 3 and Figure 4 show. Meanwhile, comparable accuracy performance is maintained on training set as shown in Figure 5 and Figure 6. This phenomenon can be attributed to masking loss operation in AGPO, since the effect of loss mask on response length is clearly shown in Figure 3 and Figure 4. In this operation, the losses of all correct or wrong groups are masked,

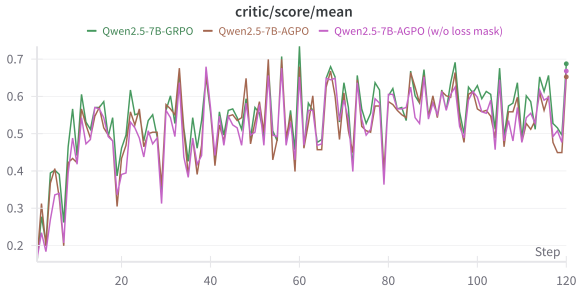

Figure 5: Reward score curves of GRPO and AGPO for Qwen2.5-7B

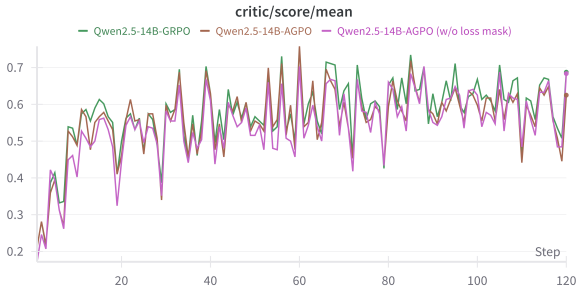

Figure 6: Reward score curves of GRPO and AGPO for Qwen2.5-14B

enhancing accurate estimation of average loss of the batch. The correction consequently amplifies gradient update towards correct direction. Therefore, in combination with the empirical observation, we believe the loss mask implicitly serves as length-based reward to constrain response length.

5 Conclusion

In this work, we propose a novel method, AGPO, to train a more powerful reasoning model. Our adaptive loss, including loss clip and mask, demonstrates noticeable improvement on both model performance and inference efficiency. Also, our method helps to avoid entropy collapse while training. We want to do more designs about the adaptive loss as future directions. For example, the current loss can be normalized to non-negative values by exponential equations.

Limitations

We will experiment on more kinds of base models and datasets in future to validate universality of our method. More ablation studies around modifications will be taken as well. It is also uncertain if our approach can produce effects together with other tricks proposed by different GRPO refinements.

References

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

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

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. 2023. A survey of reinforcement learning from human feedback. *arXiv preprint arXiv:2312.14925*, 10.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-Dimensional Continuous Control Using Generalized Advantage Estimation. *arXiv e-prints*, page arXiv:1506.02438.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv e-prints*, page arXiv:2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*.

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Yotam Wolf, Noam Wies, Oshri Avnery, Yoav Levine, and Amnon Shashua. 2023. Fundamental limitations of alignment in large language models. *arXiv preprint arXiv:2304.11082*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, et al. 2025. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2).