Probability Weighted Policy Optimization

Kun Dong[1,2], **Jian Xue**[1], **Hongjuan Pei**[1], **Yuqiu Li**[1], **Qingyuan Liu**[1,2], **Shun Mao**[1,2]
Chuang Jia[1], **Zehai Niu**[1], **Ke Lu**[1,3]*

[1]University of Chinese Academy of Sciences, No.19A Yuquan Road, Beijing 100049, China
[2]State Key Laboratory of Communication Content Cognition, People's Daily Online,
No. 2 Jintai West Road, Beijing 100733, China.
[3]Peng Cheng Laboratory, Xili Street, Shenzhen 518055, China

Abstract

Group Relative Policy Optimization (GRPO) uses the group's average reward as a baseline, eliminating the need for a value model and substantially boosting large language models (LLMs) reasoning. However, vanilla GRPO assigns uniform weight to all rollout samples for relative advantage, overlooking their generation probability information. This can result in inaccurate estimation of relative advantage, especially with limited rollouts, leading to suboptimal performance. To address this limitation, we propose Probability Weighted Policy Optimization (PWPO), which explicitly incorporates the sample generation probability into the calculation of relative advantage. This probability-aware training mechanism enables the dynamic adjustment of each sample's weight based on its generation probability. Experimental results on five mathematical reasoning benchmarks demonstrate the superiority of our method.

Introduction

Reinforcement Learning (RL) has become a crucial technique for advancing Large Language Models (LLMs) (Wen et al. 2025a; He et al. 2025; Xiaomi et al. 2025), enabling them to surpass the limitations of Supervised Fine-Tuning (SFT). Currently, top-performing models, such as OpenAI o1 (OpenAI 2024) and Kimi K1.5 (Team et al. 2025), all leverage RL-based fine-tuning to enhance their capabilities in complex reasoning tasks.

In early research, the Proximal Policy Optimization (PPO) algorithm (Schulman et al. 2017) was widely adopted to optimize decision-making in intermediate step generation. This approach involves training a value model for estimating state values, which serve as the baseline to compute relative advantage, guiding the policy's updates. More recent studies, exemplified by DeepSeek (Guo et al. 2025), have explored a verifiable reward-based reinforcement learning approach (Wen et al. 2025b), adopting the Group Relative Policy Optimization (GRPO) algorithm (Shao et al. 2024) to simplify training process. Specifically, this approach samples multiple responses for a given input and utilizes the average reward of the response group as the baseline, eliminating the need for a value model. Consequently, GRPO has rapidly become a focal point for enhancing LLMs capabilities, particularly in domains like mathematics.

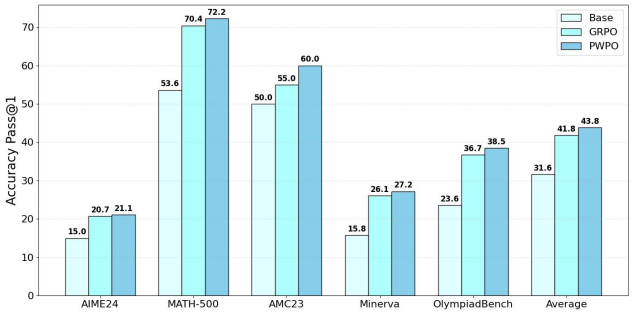

Figure 1: Performance comparison on five mathematical reasoning benchmarks. We use Qwen-Math-7B as the base model and employ GRPO and our PWPO algorithms to enhance reasoning capability, respectively. Our PWPO achieves consistent performance gains over the GRPO baseline across all mathematical benchmarks, demonstrating its effectiveness for reasoning.

Despite promising advances (Xue et al. 2025; Chu et al. 2024; Yao et al. 2024; Bai et al. 2025), the GRPO algorithm still encounters certain limitations. Specifically, for a given problem, the response reward of a certain LLM follows a corresponding probability distribution. However, due to practical constraints like computational resources or time efficiency considerations, the feasible number of responses that can be sampled in real training scenarios is quite limited. Consequently, the sampled rewards may not accurately represent the true reward distribution. This sampling restriction leads to an inaccurate advantage baseline, thereby adversely affecting group-wise advantage estimations and subsequent policy optimization.

To address this limitation, we propose Probability Weighted Policy Optimization (PWPO), which explicitly incorporates the sample generation probability into the calculation of relative advantage. Unlike the vanilla GRPO algorithm, which assigns uniform weight to all rollout samples, our method dynamically adjusts sample weights based on the probability information of their generation. This probability-aware training mechanism enables more robust advantage

*Corresponding author.

estimation. Crucially, our method integrates seamlessly with the GRPO algorithm without introducing modification to the policy networks or extra learning components. Before delving into details, we sum up our core contributions in this work as follows:

- We introduce PWPO, a novel approach designed for enhanced advantage estimation, requiring no additional learning parameters.
- We evaluate our approach by comparing it to the standard group-mean approach and demonstrate its superiority on five mathematical reasoning benchmarks.

Related Work

Large Reasoning Models. Recent research in Large Language Models (LLMs) has yielded models such as OpenAI o1 (OpenAI 2024) and Google Gemini (Team et al. 2023), which exhibit exceptional proficiency in complex problem-solving and reasoning. These modern methods move beyond earlier techniques—such as Chain-of-Thought (CoT) (Wei et al. 2022), Buffer of Thought (BoT) (Yang et al. 2024b), Tree-of-Thoughts (TOT) (Yao et al. 2023), or Graph-of-Thought (GOT) (Besta et al. 2024)—by utilizing verifiable rewards to incentivize LLM learning through self-exploration. This approach has enabled significant breakthroughs in the scaling of reinforcement learning training. Inspired by the Group Relative Policy Optimization (GRPO) algorithm (Shao et al. 2024) employed in DeepSeek-R1 (Guo et al. 2025), the research community is actively developing further techniques to enhance large-scale reinforcement learning.

Reinforcement Learning. Reinforcement Learning (RL) has become a key technology for optimizing Large Language Models (LLMs). Early works focus on Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al. 2022), aligning LLMs with human values by training reward models on extensive human preference data. However, this method proved to be highly resource-intensive due to the need for manual annotation (Touvron et al. 2023). Subsequent innovations like Direct Preference Optimization (DPO) (Rafailov et al. 2023) streamlined the process by directly training the policy on preference data, but still contended with scalability limitations and potential biases stemming from the preference models (Gao, Schulman, and Hilton 2023).

Recently, RL with Verifiable Rewards (RLVR) emerged as a powerful alternative (Xiaomi et al. 2025; Team et al. 2025), especially for tasks with objective correctness criteria, such as mathematical reasoning. RLVR bypasses the complexities of learned models by deriving reward signals from rule-based verification mechanisms, leading to breakthroughs of cutting-edge reasoning capabilities. Further leveraging the technical lineage of model-free RL—including Policy Gradients (Kakade 2001), TRPO (Schulman et al. 2015a), and PPO (Schulman et al. 2017)—the GRPO algorithm (Shao et al. 2024) was introduced. GRPO specifically adapts RL for LLM fine-tuning, consolidating the application of reinforcement learning in complex reasoning tasks by incorporating robust policy optimization mechanisms.

Preliminaries

Proximal Policy Optimization (PPO)

PPO (Schulman et al. 2017) stands as a milestone algorithm in reinforcement learning. By introducing the importance sampling ratio $r_t(\theta)$, PPO enables efficient reuse of data collected under an old policy $\pi_{\theta_{\text{old}}}$ to update the current policy π_θ, thereby significantly improving sample efficiency. Moreover, through a clipping operation, PPO restricts policy updates to remain within a proximal region around the previous policy. This prevents excessive divergence between successive policies and ensures training stability. For a data distribution $\mathcal{D}$, PPO updates the policy by maximizing the following objective:

$$\mathcal{J}_{\text{PPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},o_{\le t}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\Big(\min(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta),1-\varepsilon,1+\varepsilon)\hat{A}_t)\Big)\right], \tag{1}$$

where

$$r_t(\theta) = \frac{\pi_\theta(o_t \mid q, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t \mid q, o_{<t})}. \tag{2}$$

$\hat{A}_t$ is the estimated advantage at timestep t, computed via Generalized Advantage Estimation (GAE) (Schulman et al. 2015b) using the reward function R and the value model V:

$$\hat{A}_t^{\text{GAE}(\gamma,\lambda)} = \sum_{l=0}^{\infty}(\gamma\lambda)^l\delta_{t+l}, \tag{3}$$

where

$$\delta_l = R_l + \gamma V(s_{l+1}) - V(s_l), \quad 0 \le \gamma,\lambda \le 1. \tag{4}$$

Group Relative Policy Optimization (GRPO)

As shown in Formula 4, the advantage computation in PPO involves training an extra value model V to estimate state values as baseline. In contrast, GRPO (Shao et al. 2024) estimates the advantage in a group-relative manner, eliminating the need for a value model. Given a question q, the policy network $\pi_{\theta_{\text{old}}}$ samples a group of G responses $\{o_i\}_{i=1}^G$. The advantage of the i-th response is then derived by normalizing the group-level rewards $\{R_i\}_{i=1}^G$:

$$\hat{A}_{i,t} = \frac{R_i - \text{mean}(\{R_i\}_{i=1}^G)}{\text{std}(\{R_i\}_{i=1}^G) + \eta}, \tag{5}$$

where η is a small constant.

Following PPO, GRPO also employs a clipped objective, complemented by a KL penalty:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\{o_i\}_{i=1}^G\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \Big(\min(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}(r_{i,t}(\theta),1-\varepsilon,1+\varepsilon)\hat{A}_{i,t}) - \beta D_{\text{KL}}(\pi_\theta||\pi_{\text{ref}})\Big)\Bigg], \tag{6}$$

where

$$r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}. \tag{7}$$

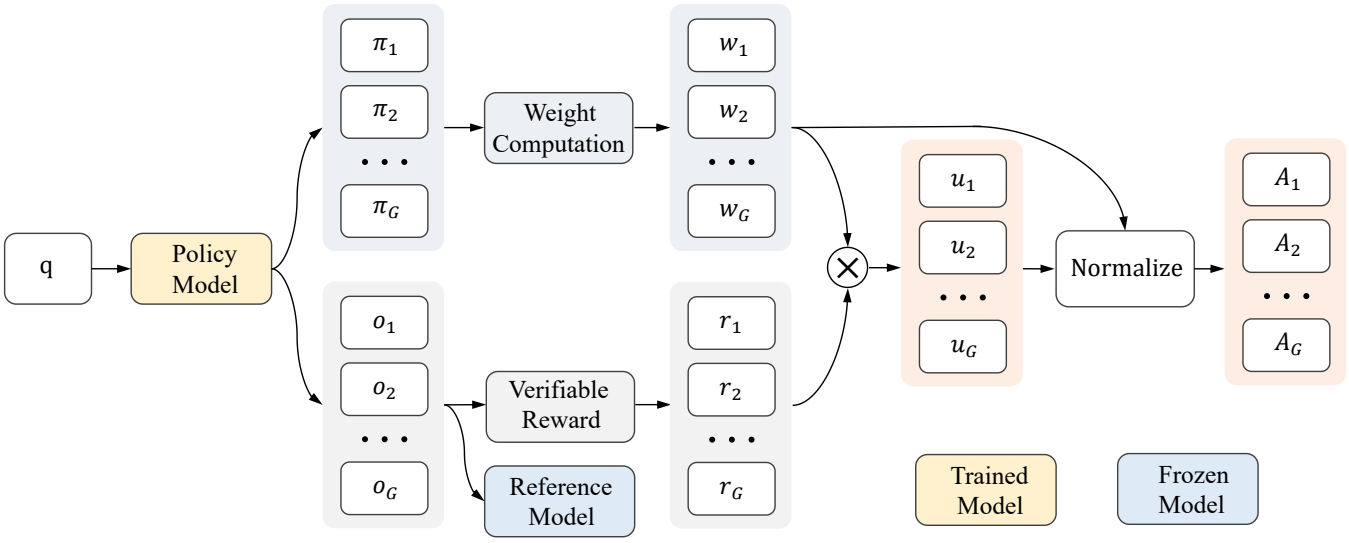

Figure 2: Overview of PWPO. It explicitly incorporates the sample generation probability into the calculation of relative advantage, enabling the dynamic adjustment of each sample's weight based on its generation probability.

Probability Weighted Policy Optimization (PWPO)

While the group-mean calculation offers an effective approach to estimate advantage, it encounters certain limitations in real training scenarios. Specifically, practical constraints like limited hardware or time efficiency often restrict the feasible number of sampled responses, resulting in an inaccurate baseline. Consider a toy case where a model's probability of answering a question correctly is $P(\text{correct}) = 0.7$, and its probability of answering incorrectly is $P(\text{incorrect}) = 0.3$. Assuming only two answers can be sampled per question, a hypothetical sample set yields one correct answer (a) and one incorrect answer (b). If uniform weights are assigned to both answers, the resulting sample mean is $\frac{(1\cdot 1)+(0\cdot 1)}{1+1} = 0.5$. This deviates from the true expected mean of 0.7. Since the model is more likely to answer correctly, the probability of generating the correct answer (a) is typically higher than that of generating the incorrect answer (b). We can use the generated probability as the weight to calculate a weighted mean. For instance, if the generation probability for a is 0.3 and for b is 0.2, the weighted mean is $\frac{(1\cdot 0.3)+(0\cdot 0.2)}{0.3+0.2} = 0.6$, which is closer to the true expected mean of 0.7.

Given the above motivations, we propose the Probability Weighted Policy Optimization (PWPO), which explicitly incorporates the generation probability into the calculation of relative advantage. Given a question q, we can sample a group of G responses $\{o_i\}_{i=1}^{G}$ and obtain their corresponding generation probability $\pi_\theta(o_i|q)$. Then we define the weight of the i-th response as

$$\begin{aligned} w_i &= (\pi_\theta(o_i|q))^{\frac{1}{|o_i|}} \\ &= \exp\left(\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\frac{\log \pi_\theta(o_{i,t}|q,o_{i,<t})}{\tau}\right), \end{aligned} \tag{8}$$

where τ controls the influence magnitude of generation probability on weight. Subsequently, we can calculate a weighted mean as

$$m = \frac{\sum_{i=1}^{G} w_i \cdot R_i}{\sum_{i=1}^{G} w_i}. \tag{9}$$

The weighted variance is calculated as follows

$$\sigma^2 = \frac{\sum_{i=1}^{G} w_i (R_i - m)^2}{S-1}, \tag{10}$$

where

$$S = \frac{(\sum_{i=1}^{G} w_i)^2}{\sum_{i=1}^{G} w_i^2}. \tag{11}$$

Therefore, we can ultilize m and σ to derive the group-wise advantage of the i-th response as follows

$$\hat{A}_{i,t} = \frac{R_i - m}{\sigma + \eta}. \tag{12}$$

With this probability-aware training mechanism, we can dynamically adjust each sample's weight based on its generation probability, enabling more robust advantage estimation. During training, we only enable the accuracy reward for simplicity.

Experiments

Task Setting

We evaluate our method on mathematical reasoning task, training on the MATH-lighteval dataset (Hendrycks et al. 2021) (7.5k problem-answer pairs). Model performance is measured by the pass@1 metric across five benchmarks: AIME 2024, MATH 500 (Lightman et al. 2023), AMC 2023, Minerva (Lewkowycz et al. 2022), and Olympiad Bench (Huang et al. 2024). We first conduct ablation studies with the Qwen-Math-1.5B model (Yang et al. 2024a) to select optimal hyperparameters. We then scale our approach to the Qwen-Math-7B model to validate its effectiveness.

Table 1: Comparison under GRPO and our PWPO algorithms on the Qwen-Math-7B model.

Group Size	Model/Method	AIME 2024	MATH 500	AMC 2023	Minerva	OlympiadBench	Average
-	Qwen-Math-7B	15.0	53.6	50.0	15.8	23.6	31.6
4	GRPO	20.1	67.7	52.5	23.9	33.2	39.5
	PWPO	20.7	69.8	55.0	25.7	35.3	41.3
8	GRPO	20.7	70.4	55.0	26.1	36.7	41.8
	PWPO	21.1	72.2	60.0	27.2	38.5	43.8

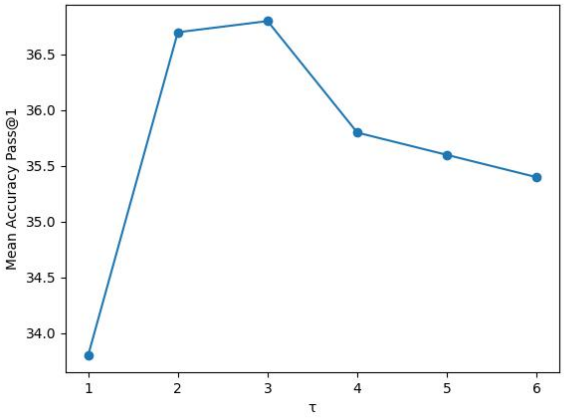

Figure 3: Ablation study on the Temperature τ. The best performance is achieved when we set τ to 3.

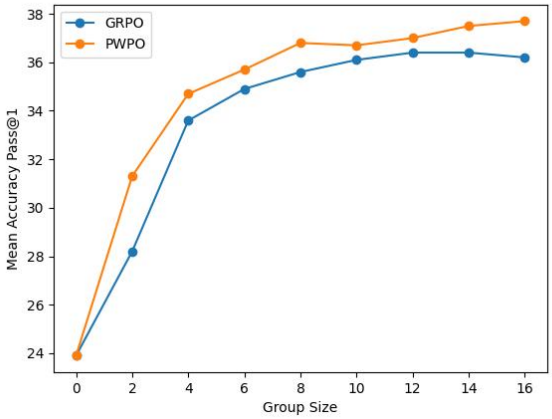

Figure 4: Model performance under different group sizes. The results improve consistently as the group size increases.

Implementation Details

During training, all models were optimized for one epoch using the AdamW optimizer (Loshchilov and Hutter 2017). We applied a learning rate of 3e-6 for the 1.5B model and 1e-6 for the 7B model. The learning rate schedules follow cosine decay pattern (Loshchilov and Hutter 2016), incorporating a warmup over the first 10% of training steps. The training configuration use a global batch size of 512. The maximum response length is limited to 2048 and 4096 tokens for training and evaluation.

The Influence of Temperature τ

We first investigate the influence of temperature τ on the Qwen-Math-1.5B model to select the optimal value. This parameter directly influences the weighting of samples. Specifically, a lower temperature τ increases the impact of sample generation probability on its weight. As shown in Figure 3, we can get the best mean accuracy across five benchmarks when τ is set to 3. Therefore, we opt for this value for subsequent experiments.

Consistent Improvement across Group Sizes

Here we vary the number of sampled responses per problem during training to evaluate its impact on the performance of the Qwen-Math-1.5B model. As Figure 4 illustrates, model performance improves consistently as the group size increases. Crucially, our proposed PWPO method demonstrates superiority compared to the vanilla GRPO algorithm, delivering consistent performance improvements across different group sizes .

Robustness to Larger Model

To further verify the effectiveness of our method, we conduct experiments using a larger Qwen-Math-7B model. Specifically, we evaluate the performance of both the GRPO and our proposed PWPO algorithms at group sizes of 4 and 8. As can be seen in Table 1, our PWPO outperforms the vanilla GRPO algorithm under both settings, further demonstrating the robustness of our approach.

Conclusion

In this paper, we propose Probability Weighted Policy Optimization (PWPO), a novel reinforcement learning algorithm for training LLMs. PWPO enhances the vanilla GRPO algorithm by explicitly incorporating the sample generation probability into the calculation of relative advantage. This allows for the dynamic adjustment of each sample's weight based on its generation probability. Experiments on mathematical reasoning tasks demonstrate the superiority of our approach over the standard group-mean method.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (U23A20388, 62320106007), the Beijing Natural Science Foundation (L254018), and the Joint Fund of the Science and Technology R&D Program of Henan (235200810031).

References

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; et al. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 17682–17690.

Chu, X.; Qiao, L.; Zhang, X.; Xu, S.; Wei, F.; Yang, Y.; Sun, X.; Hu, Y.; Lin, X.; Zhang, B.; et al. 2024. Mobilevlm v2: Faster and stronger baseline for vision language model. *arXiv preprint arXiv:2402.03766*.

Gao, L.; Schulman, J.; and Hilton, J. 2023. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, 10835–10866. PMLR.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

He, J.; Liu, J.; Liu, C. Y.; Yan, R.; Wang, C.; Cheng, P.; Zhang, X.; Zhang, F.; Xu, J.; Shen, W.; et al. 2025. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*.

Hendrycks, D.; Burns, C.; Kadavath, S.; Arora, A.; Basart, S.; Tang, E.; Song, D.; and Steinhardt, J. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Huang, Z.; Wang, Z.; Xia, S.; Li, X.; Zou, H.; Xu, R.; Fan, R.-Z.; Ye, L.; Chern, E.; Ye, Y.; et al. 2024. Olympicarena: Benchmarking multi-discipline cognitive reasoning for superintelligent ai. *Advances in Neural Information Processing Systems*, 37: 19209–19253.

Kakade, S. M. 2001. A natural policy gradient. *Advances in neural information processing systems*, 14.

Lewkowycz, A.; Andreassen, A.; Dohan, D.; Dyer, E.; Michalewski, H.; Ramasesh, V.; Slone, A.; Anil, C.; Schlag, I.; Gutman-Solo, T.; et al. 2022. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35: 3843–3857.

Lightman, H.; Kosaraju, V.; Burda, Y.; Edwards, H.; Baker, B.; Lee, T.; Leike, J.; Schulman, J.; Sutskever, I.; and Cobbe, K. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Loshchilov, I.; and Hutter, F. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

OpenAI. 2024. Learning to reason with LLMs.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36: 53728–53741.

Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M.; and Moritz, P. 2015a. Trust region policy optimization. In *International conference on machine learning*, 1889–1897. PMLR.

Schulman, J.; Moritz, P.; Levine, S.; Jordan, M.; and Abbeel, P. 2015b. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y.; Wu, Y.; et al. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Team, G.; Anil, R.; Borgeaud, S.; Alayrac, J.-B.; Yu, J.; Soricut, R.; Schalkwyk, J.; Dai, A. M.; Hauth, A.; Millican, K.; et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Team, K.; Du, A.; Gao, B.; Xing, B.; Jiang, C.; Chen, C.; Li, C.; Xiao, C.; Du, C.; Liao, C.; et al. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Wen, L.; Cai, Y.; Xiao, F.; He, X.; An, Q.; Duan, Z.; Du, Y.; Liu, J.; Tang, L.; Lv, X.; et al. 2025a. Light-r1: Curriculum sft, dpo and rl for long cot from scratch and beyond. *arXiv preprint arXiv:2503.10460*.

Wen, X.; Liu, Z.; Zheng, S.; Xu, Z.; Ye, S.; Wu, Z.; Liang, X.; Wang, Y.; Li, J.; Miao, Z.; et al. 2025b. Reinforcement Learning with Verifiable Rewards Implicitly Incentivizes Correct Reasoning in Base LLMs. *arXiv preprint arXiv:2506.14245*.

Xiaomi, L.; Xia, B.; Shen, B.; Zhu, D.; Zhang, D.; Wang, G.; Zhang, H.; Liu, H.; Xiao, J.; Dong, J.; et al. 2025. MiMo: Unlocking the Reasoning Potential of Language Model–From Pretraining to Posttraining. *arXiv preprint arXiv:2505.07608*.

Xue, Z.; Wu, J.; Gao, Y.; Kong, F.; Zhu, L.; Chen, M.; Liu, Z.; Liu, W.; Guo, Q.; Huang, W.; et al. 2025. DanceGRPO: Unleashing GRPO on Visual Generation. *arXiv preprint arXiv:2505.07818*.

Yang, A.; Zhang, B.; Hui, B.; Gao, B.; Yu, B.; Li, C.; Liu, D.; Tu, J.; Zhou, J.; Lin, J.; Lu, K.; Xue, M.; Lin, R.; Liu, T.; Ren, X.; and Zhang, Z. 2024a. Qwen2.5-Math Technical Report: Toward Mathematical Expert Model via Self-Improvement. *arXiv preprint arXiv:2409.12122*.

Yang, L.; Yu, Z.; Zhang, T.; Cao, S.; Xu, M.; Zhang, W.; Gonzalez, J. E.; and Cui, B. 2024b. Buffer of thoughts:

Thought-augmented reasoning with large language models. *Advances in Neural Information Processing Systems*, 37: 113519–113544.

Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T.; Cao, Y.; and Narasimhan, K. 2023. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36: 11809–11822.

Yao, Y.; Yu, T.; Zhang, A.; Wang, C.; Cui, J.; Zhu, H.; Cai, T.; Li, H.; Zhao, W.; He, Z.; et al. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*.