APPENDIX

Anonymous authors
Paper under double-blind review

A RELATED WORK

Critic-based RL4LLM algorithms Shao et al. (2024) first demonstrated that large-scale reinforcement learning (RL) with outcome-based rewards can unlock long-tail reasoning, beginning from an unaligned base model. This finding has led to numerous variations of the Proximal Policy Optimization (PPO) algorithm. As far as we know, most algorithm research is mainly based on the baseline normalized advantage calculation method (Hu, 2025; Liu et al., 2025b; Chen et al., 2025).

On the other hand, value-based algorithm innovations are relatively few, Yuan et al. (2025b) argued that the decay factor is not well-suited for complex reasoning tasks that require long chains of thought (CoT). Yue et al. (2025); Zhu et al. (2025); Zhao et al. (2025) proposed novel mechanisms to enhance the robustness of the critic model when faced with noisy reward signals. Open-Reasoner-Zero (Hu et al., 2025) argues that, within this regime, vanilla PPO without KL regularization suffices to scale training stably. T-PPO (Fan et al., 2025) uses critic to enhance the stability of policy training in the long-tail asynchronous setting (Fu et al., 2025). Another similar research line to introduce critic-like models is done with the introduction of Implicit PRM (Yuan et al., 2025a). This approach is also able to provide token-level supervision for scalable RL training. PRIME (Cui et al., 2025) adapted a specific reward model formulation to directly generate token-level rewards. However, current mainstream RL4LLM algorithms primarily emphasize critic-free optimization (Zhang et al., 2025). In this context, our research aim to underscore the importance of the critic in RL4LLM scenarios and try to address the deployment limitations associated with critics.

Asymmetric architecture. In the realm of continuous deep RL, recent studies have investigated the potential of asymmetric network structures by reducing the capacity of the actor network. For example, Mastikhina et al. (2025); Mysore et al. (2021) suggest that the actor can function effectively with a significantly smaller capacity compared to the critic. Empirical evidence from Tan et al. (2022) supports this idea, demonstrating that sparsifying the policy network can enhance effective policy learning while significantly improving both inference and training speeds. Additionally, Liu et al. (2025a) found that pruning the actor network's topology based on trial gradients can yield better performance. Similarly, Ma et al. (2025) revealed that even random pruning of the actor network can maintain performance within the SimBa network architecture (Lee et al., 2024). These contributions highlight the adaptability of RL in accommodating asymmetric designs, providing valuable insights for our research. However, existing works primarily concentrate on reducing the actor's size within simple network frameworks. In contrast, our paper pioneers the exploration of effectively guiding a small critic to inform a larger actor by optimizing the PPO algorithm within the RL4LLM scenario.

B THE PERFORMANCE GAIN OF AsyPPO ON THE SMALL MODEL STRATEGY

Policy model	Base model	Symmetric PPO	AsyPPO
Qwen3-4b-Base	30.5%	47.3%	53.1% +6.1%
Qwen3-4b-Base	31.7%	50.6%	53.8% +3.2%

Table 1: Peak accuracy comparison of Symmetric PPO and AsyPPO under high data reuse setting (UTD=4) over six benchmarks. Score calculation same as **??** (b). Purple score denotes the improvement compare to Symmetric PPO.

We set both the classic symmetrical PPO and our AsyPPO to the optimal Settings. AsyPPO uniformly initializes mini-critics using the Qwen3-1.7b-Base model. AsyPPO employs two mini-critics with advantage masking at 20%. And use the open source hard training dataset in (Liu et al., 2025c), which is selected from DeepMath-103k (He et al., 2025) with sampling probability proportional to each entry's assigned difficulty level. We report the average@4 across six challenging benchmarks, i.e., MATH-500, OlympiadBench, MinervaMath, and AMC 2023, AIME 2025, AIME 2024.

Overall, Table 1 shows that AsyPPO effectively enhances the reasoning capabilities of two small models of different sizes, achieving respective improvements of 22.6% and 22.1% over their original performance. Compared to symmetric PPO, our algorithm delivers gains of 6.1% and 3.2%, while maintaining lightweight deployment. Upon analyzing specific benchmarks, our approach demonstrates notable advancements. For instance, on AIME 2025, we observed respective increases of approximately 4% (4B) and 6% (8B) compared to symmetric PPO. Similarly, on MATH-500, the improvements were around 3% (4B) and 2% (8B), and on MinervaMath, the gains were approximately 2% (4B) and 4% (8B). In the remaining three tasks, our method maintained performance levels comparable to those of symmetric PPO.

C Detailed Experimental Setup

C.1 Plot setup

To ensure clarity and intuitiveness in the qualitative analysis, all curves are consistently smoothed using identical parameters. Specifically, the mean values are computed using an 11-step moving window with an exponential smoothing factor of 0.6. The smooth window set as 4 and 2.

C.2 Hyperparameters

We employ ROLL, a user-friendly and efficient open-source reinforcement learning framework, to implement our pipeline. Subsequently, the key parameters observed during the training process are presented as follows. See our code config file for more details on the parameters. For the 14b policy training. We uniformly arrange the actors on (0,16) and the critics on (16,32) GPUs. For other small models, we uniformly place the actor at (0,8) and the critic at (8,16) GPU. Detailed settings can be found in next page.

```
# We use below setup for 4b and 8b policy
seed: 42
max_steps: 500
save_steps: 500
logging_steps: 1
eval_steps: 1
gamma: 1.0  # discount factor
lambd: 1.0  # GAE lambda
rollout_batch_size: 64
prompt_length: 1024
response_length: 8000
value_aggregation_strategy: "mean"
gradient_mask_percentage: 0.2 # mask 20%
entropy_loss_coef: 0.01
entropy_filter_mask_percentage: 0.2 # filter out 20%
ppo_epochs: 1 # 4 is also used in main experiments
adv_estimator: "gae"
init_kl_coef: 0.0
async_generate_level: 1

actor_train:
  training_args:
    learning_rate: 1.0e-6
    weight_decay: 0
    per_device_train_batch_size: 1
    gradient_accumulation_steps: 256
    warmup_steps: 50
    num_train_epochs: 50

critic_1:
  training_args:
    learning_rate: 1.0e-5
    weight_decay: 1.0e-2
    warmup_steps: 5
    per_device_train_batch_size: 1
    gradient_accumulation_steps: 128
    warmup_steps: 5
    num_train_epochs: 50

critic_2:
  training_args:
    learning_rate: 1.0e-5
    weight_decay: 1.0e-2
    warmup_steps: 5
    per_device_train_batch_size: 1
    gradient_accumulation_steps: 128
    warmup_steps: 5
    num_train_epochs: 50
  ...

actor_infer:
  generating_args:
    max_new_tokens: ${response_length}
    top_p: 0.99
    top_k: 100
    num_beams: 1
    temperature: 0.99
    num_return_sequences: 32
  ...
```

```
# We use below setup for 14b policy
seed: 42
max_steps: 500
save_steps: 500
logging_steps: 1
eval_steps: 1
gamma: 1.0  # discount factor
lambd: 1.0  # GAE lambda
value_aggregation_strategy: "mean"
gradient_mask_percentage: 0.2 # mask 20%
entropy_loss_coef: 0.01
entropy_filter_mask_percentage: 0.2  # filter out 20% or 0%
rollout_batch_size: 64
prompt_length: 1024
response_length: 8000
infer batch size: 4
ppo_epochs: 4
adv_estimator: "gae"
init_kl_coef: 0.0
async_generate_level: 1
actor_train:
  training_args:
    learning_rate: 1.0e-6
    weight_decay: 0
    per_device_train_batch_size: 2
    gradient_accumulation_steps: 6
    warmup_steps: 50
    num_train_epochs: 50
critic_1:
  training_args:
    learning_rate: 1.0e-5
    weight_decay: 1.0e-2
    warmup_steps: 5
    per_device_train_batch_size: 2
    gradient_accumulation_steps: 16
    warmup_steps: 5
    infer batch size: 4
    num_train_epochs: 50
critic_2:
  training_args:
    learning_rate: 1.0e-5
    weight_decay: 1.0e-2
    warmup_steps: 5
    per_device_train_batch_size: 2
    gradient_accumulation_steps: 16
    warmup_steps: 5
    infer batch size: 4
    num_train_epochs: 50
  ...
actor_infer:
  generating_args:
    max_new_tokens: ${response_length}
    top_p: 0.99
    top_k: 100
    num_beams: 1
    temperature: 0.99
    num_return_sequences: 32
  ...
```

C.3 PROMPT

In this work, we incorporate the following instruction into the system prompt to encourage the model to better demonstrate its reasoning process: **"Please reason step by step, and put your final answer within \boxed{}."** This setting is designed to guide the model to perform step-by-step reasoning and explicitly present the final answer in the form of \boxed{ }, thereby enhancing the clarity and readability of the output.

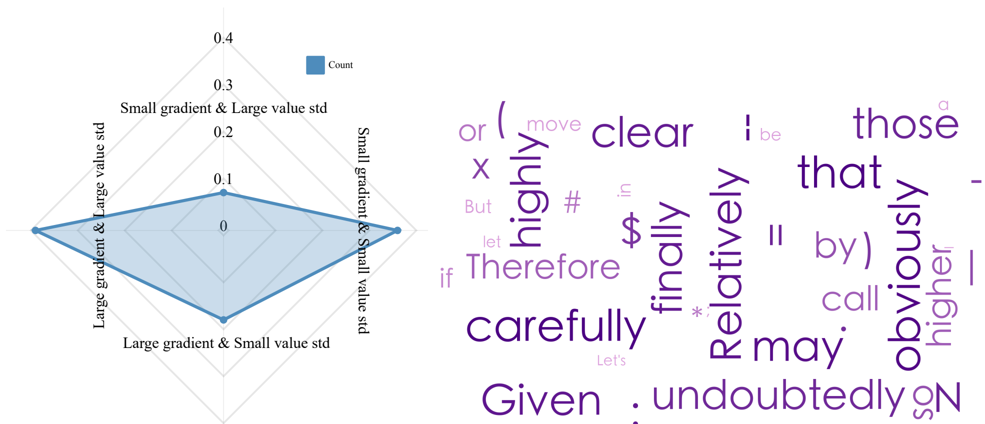

Figure 1: Left: Statistics within a mini-batch in the mid-training stage. Right: The 40 tokens that are masked most frequently in the same mini-batch.

D THE RELATIONSHIP BETWEEN VALUE STD AND STATE INFORMATION QUANTITY

Specifically, for the training scenarios of 8b actors and two 0.6b critics, we use the value-std corresponding to the global state and the median of the gradient magnitude to categorize the states into four types. Namely, large gradient & large value std, large gradient & small value std, small gradient & large value std, small gradient & small value std. The results in Figure 1 (Left) show that the vast majority of states are classified into the categories of large gradient & large value std and small gradient & small value std, thereby empirically proving the positive relationship between value std and the learning value (information quantity) of the state.

E VISUALIZATION OF WORD CLOUDS

We statistically analyzed the word clouds of the tokens with the highest mask frequency in the initial stage of `AsyPPO` training. The results in Figure 1 (Right) show that our mask mechanism tends to mask adjectives, adverbs, and some isolated symbols, with less involvement in logical transitions, except for the slightly prominent progressive word "therefore".

LLM USAGE

LLMs were used to assist paper editing and to write the code for plotting experiments.

REFERENCES

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, Zhiyuan Liu, Maosong Sun, Bowen Zhou, and Ning Ding. Process reinforcement through implicit rewards. *ArXiv*, abs/2502.01456, 2025. URL https://api.semanticscholar.org/CorpusID:276107672.

Tiantian Fan, Lingjun Liu, Yu Yue, Jiaze Chen, Chengyi Wang, Qiying Yu, Chi Zhang, Zhiqi Lin, Ruofei Zhu, Yufeng Yuan, et al. Truncated proximal policy optimization. *arXiv preprint arXiv:2506.15050*, 2025.

Wei Fu, Jiaxuan Gao, Xujie Shen, Chen Zhu, Zhiyu Mei, Chuyi He, Shusheng Xu, Guo Wei, Jun Mei, Jiashu Wang, et al. Areal: A large-scale asynchronous reinforcement learning system for language reasoning. *arXiv preprint arXiv:2505.24298*, 2025.

Zhiwei He, Tian Liang, Jiahao Xu, Qiuzhi Liu, Xingyu Chen, Yue Wang, Linfeng Song, Dian Yu, Zhenwen Liang, Wenxuan Wang, et al. Deepmath-103k: A large-scale, challenging, decontaminated, and verifiable mathematical dataset for advancing reasoning. *arXiv preprint arXiv:2504.11456*, 2025.

Jian Hu. Reinforce++: A simple and efficient approach for aligning large language models. *arXiv preprint arXiv:2501.03262*, 2025.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. *arXiv preprint arXiv:2503.24290*, 2025.

Hojoon Lee, Dongyoon Hwang, Donghu Kim, Hyunseung Kim, Jun Jet Tai, Kaushik Subramanian, Peter R. Wurman, Jaegul Choo, Peter Stone, and Takuma Seno. Simba: Simplicity bias for scaling up parameters in deep reinforcement learning. *ArXiv*, abs/2410.09754, 2024. URL https://api.semanticscholar.org/CorpusID:273346233.

Jiashun Liu, Johan Samir Obando Ceron, Aaron Courville, and Ling Pan. Neuroplastic expansion in deep reinforcement learning. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=20qZK2T7fa.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025b.

Zihe Liu, Jiashun Liu, Yancheng He, Weixun Wang, Jiaheng Liu, Ling Pan, Xinyu Hu, Shaopan Xiong, Ju Huang, Jian Hu, et al. Part i: Tricks or traps? a deep dive into rl for llm reasoning. *arXiv preprint arXiv:2508.08221*, 2025c.

Guozheng Ma, Lu Li, Zilin Wang, Li Shen, Pierre-Luc Bacon, and Dacheng Tao. Network sparsity unlocks the scaling potential of deep reinforcement learning. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=mIomqOskaa.

Olya Mastikhina, Dhruv Sreenivas, and Pablo Samuel Castro. Optimistic critics can empower small actors. *arXiv preprint arXiv:2506.01016*, 2025.

Siddharth Mysore, Bassel El Mabsout, Renato Mancuso, and Kate Saenko. Honey. i shrunk the actor: A case study on preserving performance with smaller actors in actor-critic rl. *2021 IEEE Conference on Games (CoG)*, pp. 01–08, 2021.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yi Xiang Marcus Tan, Pihe Hu, L. Pan, and Longbo Huang. Rlx2: Training a sparse deep reinforcement learning model from scratch. *ArXiv*, abs/2205.15043, 2022.

Lifan Yuan, Wendi Li, Huayu Chen, Ganqu Cui, Ning Ding, Kaiyan Zhang, Bowen Zhou, Zhiyuan Liu, and Hao Peng. Free process rewards without process labels. In *Forty-second International Conference on Machine Learning*, 2025a. URL https://openreview.net/forum?id=8ThnPFhGm8.

Yufeng Yuan, Yu Yue, Ruofei Zhu, Tiantian Fan, and Lin Yan. What's behind ppo's collapse in long-cot? value optimization holds the secret. *ArXiv*, abs/2503.01491, 2025b. URL https://api.semanticscholar.org/CorpusID:276766648.

Yu Yue, Yufeng Yuan, Qiying Yu, Xiaochen Zuo, Ruofei Zhu, Wenyuan Xu, Jiaze Chen, Chengyi Wang, TianTian Fan, Zhengyin Du, et al. Vapo: Efficient and reliable reinforcement learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*, 2025.

Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Peng Li, Yu Fu, Xingtai Lv, Yuchen Zhang, Sihang Zeng, Shang Qu, Hao-Si Li, Shijie Wang, Yuru Wang, Xi-Dai Long, Fangfu Liu, Xiang Xu, Jiaze Ma, Xuekai Zhu, Ermo Hua, Yihao Liu, Zonglin Li, Hua yong Chen, Xiaoye Qu, Yafu Li, Weize Chen, Zhenzhao Yuan, Junqi Gao, Dong Li, Zhiyuan Ma, Ganqu Cui, Zhiyuan Liu, Biqing Qi, Ning Ding, and Bowen Zhou. A survey of reinforcement learning for large reasoning models. 2025. URL https://api.semanticscholar.org/CorpusID:281247204.

Yuzhong Zhao, Yue Liu, Junpeng Liu, Jingye Chen, Xun Wu, Yaru Hao, Tengchao Lv, Shaohan Huang, Lei Cui, Qixiang Ye, et al. Geometric-mean policy optimization. *arXiv preprint arXiv:2507.20673*, 2025.

Dingwei Zhu, Shihan Dou, Zhiheng Xi, Senjie Jin, Guoqiang Zhang, Jiazheng Zhang, Junjie Ye, Mingxu Chai, Enyu Zhou, Ming Zhang, Caishuang Huang, Yunke Zhang, Yuran Wang, and Tao Gui. Vrpo: Rethinking value modeling for robust rl training under noisy supervision. *ArXiv*, abs/2508.03058, 2025.