Proceedings of the 4th International Conference on Mathematical Physics and Computational Simulation
DOI: 10.54254/2753-8818/2026.33465

Stability Analysis of Reinforcement Learning Training for Large Language Models

Jia Hu

Faculty of Science and Engineering, University of Nottingham Ningbo China, Ningbo, China
scyjh13@nottingham.edu.cn

Abstract. As large language models have gradually acquired the ability to deal with complex logical reasoning and long text generation tasks, reinforcement learning has become a key technology to achieve model output and align human preferences. However, during the training process, the model often falls into the dilemma of policy collapse, performance degradation or reward hacking. Therefore, this paper systematically discusses the training stability problem faced by large language models in reinforcement learning. Based on the detailed analysis of the estimation error of the value function and other issues, this paper summarize several key reasons for the training instability. It is found that the misinitialization and signal attenuation of the value function for long chain of thought tasks will lead to errors in the results of the advantage function, and due to the lack of random search, it is easy to fall into the local optimal state in the process of policy gradient. In summary, this paper mainly discusses the shortcomings of the current mainstream algorithms in dealing with these stability, and provides prospects and development directions for future development.

Keywords: large language models, reinforcement learning, training stability, value function, policy optimization

1. Introduction

The development of large language Models (LLMS) marks the rise of natural language processing technology from simple statistical modeling to higher-dimensional semantic alignment. Although the pre-training process provides the model with a rich amount of knowledge and information, due to the randomness and unpredictability of the output results, the model cannot be directly used in rigorous production scenarios. To solve the above problems, researchers mainly use reinforcement learning technology to guide the generation process of language models by training a new reward model. Through the exploration and feedback mechanism, reinforcement learning enables the model to find the generation sequence that best meets the requirements of a specific target from a large number of Token combinations. This process plays an irreplaceable role in large model alignment, but its training complexity is far more complex than the traditional supervised fine-tuning mode.

At present, the instability of reinforcement learning training for large language models is the main factor hindering their application in academia and industry. Moreover, reinforcement learning iterations on models with hundreds of billions of parameters require extremely high computational

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2753-8818/2026.33465

resources and numerical control. This instability is mainly reflected in two aspects. On the one hand, the model performance may fluctuate sharply at some training nodes, even making the logic ability completely lost. On the other hand, the content generated by the model tends to be template and single, that is, the model will tend to find the logical holes of the reward model, and generate content that can obtain high scores but has no practical significance. This problem can cause the model to become unstable, especially in tasks with CoT characteristics of long chains of thought. For a long reasoning link, the choice of each step will have a large impact on the result of the whole reasoning process, so subtle gradient errors will accumulate in each hidden layer, leading to the collapse of the overall strategy.

This paper aims to deeply explore the root cause of the instability of reinforcement learning training for large language models. Starting from the evaluation risk of the value function, this paper analyze the distortion phenomenon of the critic network when facing long text sequences. Then, according to the change of entropy in the process of policy update, and analyze whether the balance between exploration and exploitation has shifted. In addition, this paper also compares and analyzes the performance of different algorithm architectures in the face of different tasks, so as to explore the bad impact of algorithm design defects on the macro stability of training. Through in-depth analysis of these core issues, this study attempts to provide a relatively complete perspective for understanding large-model reinforcement learning.

2. Value function problem

For large language model alignment frameworks based on the Actor-Critic architecture, the Value Function assumes the central responsibility of estimating the sum of future rewards that can be obtained for a given state. As a reinforcement learning algorithm of numerical basis, value function is not only the advantage function calculation, the key to reduce policy gradient estimation variance, but also ACTS as a current movement and the key role of links between long-term rewards. But the problem is that because of language model itself has a high dimension and the characteristics of discrete space, and therefore its value function has higher uncertainty, which affects the stability of the reinforcement learning fine-tuning the overall training result.

2.1. Value initialization bias and its cascading effects

The initial state of an LLM is typically determined by a supervised fine-tuned SFT model with an action space the size of a vocabulary in the tens of thousands. At the early stage of training, the Critic network, as a new learnable parameter block, is not well explored for such a large potential output sequence. On the one hand, the value function faces the problem of "cold start". Due to the large size of the vocabulary, it is difficult for a critic with random weights to estimate the complex state value function correctly at the beginning. If this error is not corrected quickly, then the actor will receive a wrong reward, and its gradient will be updated in a completely different direction [1]. On the other hand, this deviation can have a significant cumulative effect during the generation of long texts. When dealing with long Chain of Thought (CoT) tasks, logical reasoning often involves tens or even hundreds of reasoning steps. The value estimation error of each step will be continuously enlarged with the increase of sequence depth, forming a chain error propagation [2]. This initial estimation error will have an additive effect, causing the model's policy distribution to prematurely converge on invalid or low-quality actions before high-quality solutions have been found [3]. This "false convergence" caused by unreasonable initialization is the main reason for the

DOI: 10.54254/2753-8818/2026.33465

early collapse of large model training, which exposes the insufficient evaluation ability of the critic network when dealing with ultra-high-dimensional discrete space.

2.2. The problem of value signal attenuation and belief assignment

When the LLM large language model performs multi-step inference tasks, belief assignment is one of the main difficulties to maintain the stability of training. In general, the reward signal is only given at the end of the entire text sequence generation, and the value function needs to reflect this sparse scalar reward to each specific Token in the sequence. This sparse reward mechanism will cause serious signal attenuation problem. As the inference chain grows, the effective information of the distal reward signal will decrease exponentially when it is backward guided through the value function. When the value function cannot provide stable guidance for the intermediate steps, the model is prone to decision shock in the middle stage of operation [4]. Since the state observed by the language model is a linguistic sequence with rich meaning, its corresponding reward value should be highly sensitive to semantic differences. In reality, it is difficult for practitioners to make the critic neural network distinguish the completely opposite amount of information contained in two highly similar intermediate states, and when the reward function fails to reflect this intrinsic semantic correlation, the model will over-converge in a non-optimal direction [5]. Related theoretical analysis points out that the existing technology has limitations when dealing with specific large model architectures, and the prediction error of the value function is difficult to be completely eliminated by simple regularization.

2.3. Reward hacking and unsteadiness of evaluation metrics

In addition, the quality of the reward model also directly affects the predictive stability of the value function. In the process of reinforcement learning, there is a bad tendency to exploit the boundaries of the reward model, which is commonly referred to as reward hacking. When the value function finds that some features that are not related to the reasoning logic of the task (e.g., special punctuation habits, long phrases, or fixed response templates) can deceive the reward model into high scores, it will mistakenly evaluate these states as "high-value" regions [6]. According to Goodhart's law, the moment an evaluation metric is used as an optimization target, it is no longer a valid metric. As a result, the value function will overfit these artificially high scoring states, leading the policy network to generate texts with ridiculous logic or empty content despite high scores [7]. This phenomenon of "decoupling score from quality" not only intensifies the instability of training, but also makes the estimation target of the value function in a continuous sliding state, which further strengthens the numerical oscillation of the policy gradient [8].

2.4. High dimensionality of state representation and the generalization bottleneck

For large language models, the state space is a high-dimensional space consisting of continuous hidden layer vectors, often with thousands or even tens of thousands of dimensions. The value function needs to learn a smooth and correct mapping in this vast space. But the distribution of states that can be accessed during the online sampling process of reinforcement learning occupies only a very small part of this space. On the one hand, the generalization ability of the value function on the unseen state is poor, and the valuation oscillation is prone to occur. On the other hand, due to the drastic parameter update in the fine-tuning process of the large model, the state representation itself is also in dynamic change [9]. This "non-stationarity" makes it impossible to establish a lasting

DOI: 10.54254/2753-8818/2026.33465

and stable evaluation standard for the value function [10]. The information characterizing the instability is fed back to the policy layer through the advantage function, which is finally manifested as the lack of consistency and stability of the result output. This study propose that the difficulty in the semantic representation of the value function in very large dimensions limits the fundamental technical problem of achieving alignment stability in large language models [11].

3. Policy optimization problem

Policy optimization is one of the most intuitive ways of aligning large language models to achieve behavioral changes and preferences. Due to the extremely high number of parameters and a large number of discrete action options, the optimization of the policy function of large language models is highly non-convex and unpredictable, and the training process is prone to be unstable. This chapter focus on the entropy collapse, inefficient sampling rate and the negative impact of random exploration on training stability in policy-based optimization methods.

3.1. Analysis of entropy collapse mechanism

Entropy collapse is the most common instability phenomenon in reinforcement learning fine-tuning of large language models. In the process of reinforcement learning, in order to maximize the cumulative reward, the model will tend to output those fixed sequences that it knows will obtain high scores. As a result, the probability distribution of the policy is too narrow, and its information entropy decreases rapidly.

Firstly, driven by the goal of reward maximization, the model tends to reduce the exploration range. Once there are some templates or sentence patterns that are guaranteed to be accepted by the reward model, most of the probability mass will be concentrated on this line [12]. Second, a too low exploration range makes the model inflexible. If the information entropy is below a certain threshold, the model will not be able to provide diverse answers. Singleness means that the strategy is not diversified enough, so that if it fails, it is difficult to try another one and fall into a local optimum. Too high certainty means that the strategy is too certain, and the model is very sensitive to small perturbations in the input. If the cue word changes slightly, the quality of the output result will be greatly reduced. Although the optimization strategy based on agent entropy balance can partially alleviate this problem, however, in the actual use process, how to dynamically divide the critical value of entropy is also a problem in the current research [13].

3.2. Sample efficiency and decision noise bottleneck

Large language model reinforcement learning is also characterized by low sample efficiency and high level of decision noise. The action space of the model is too large, including the entire vocabulary, which means that a decision at each step requires a choice among tens of thousands of tokens. The contribution of most generated tokens to the reward signal in the long text generation task is close to zero, which means that not all tokens play an important role in the final reward in the long sequence generation, and this redundant information can obscure more important decision points. This sparsity makes it difficult for the model to identify which are the key nodes that really determine the success or failure of inference [14]. If all tokens are uniformly sampled during policy optimization, the training gradient will be filled with a lot of invalid noise. It is found that when dealing with complex reasoning tasks, the model is prone to the probability change of minor tokens (such as connectives and function words), which will cause the whole policy to be deflated. In a

DOI: 10.54254/2753-8818/2026.33465

long sequence, a small probability change of the small word may affect the large word. In the process of backward transmission, these noises will affect the final result and lead to large jumps in the process of multiple iterations of the strategy. This instability not only reduces the training speed, but also deeps the risk of model collapse [14].

3.3. Mathematical mechanism analysis of policy collapse

From a mathematical point of view, entropy collapse is essentially the result of the absolute dominance of the gradient of the dominant term in the objective function. For large language models, the action space corresponds to a vocabulary whose policy distribution is characterized by a policy function. $\boldsymbol{\pi_\theta(ws)}$ The mathematical expression of Shannon entropy is defined as follows.

$$H(\pi_{(\theta(\cdot|s))}) = -\sum_{w\in V}\pi_{(\theta(w|s))}log\pi_{(\theta(w|s))} \tag{1}$$

In the Softmax output layer commonly adopted by large language models, the probability distribution is expressed as

$$\pi_\theta(w|s) = \frac{exp(logit_w)}{\sum_i exp(logit_i)} \tag{2}$$

When a particular Token sequence receives a high reward, the policy gradient will force the value corresponding to that sequence to be increased. $\boldsymbol{logit_{w^*}}$ This increase leads to a rapid approach to 1 due to the exponential amplification effect of the Softmax function. $\boldsymbol{\pi_\theta\left(w^*s\right)}$

This extreme sharpening of the probability distribution directly triggers the vanishing gradient problem. According to the derivative law,

$$\frac{\partial\pi_i}{\partial logit_j} = \pi_i(\delta_{ij} - \pi_j) \tag{3}$$

When the policy enters a collapsed state, it makes the derivative term almost zero, which causes the update of the model weights to stop. $\boldsymbol{\pi_j \approx 1}$ This phenomenon is especially severe when dealing with tasks related to long sequences in the chain of thought. Since the cumulative probability of a long sequence changes exponentially with the number of steps, a small deviation in the initial probability may produce a large gradient variance in the subsequent steps, resulting in high randomness and uncertainty in the training process. This study propose that the deterministic bias caused by the dominance function is the core incentive that leads to the logical regression of large models in the alignment process.

4. Algorithm design issues

4.1. Parameter sensitivity and architectural limitations

In the relevant experiments for testing the inference ability of large language models, PPO, GRPO, and DAPO algorithms show different training characteristics. Although PPO algorithm has established a complete theoretical framework of Actor-Critic, the pressure of its dual network and video memory overhead bring relatively unstable factors in large parameters. In contrast, the generalized reinforcement strategy optimizes GRPO to cancel the critic network, and uses the relative scores within the group as the feedback signal, which simplifies the training process to a certain extent. However, literature shows that they are very sensitive to hyperparameters such as

DOI: 10.54254/2753-8818/2026.33465

learning rate and group sampling size. In a distributed training environment, small parameter jitters may be amplified by gradient accumulation, resulting in completely different convergence trajectories of different models on the same task [15].

4.2. Adaptation differences in task scenarios

The robustness of the reinforcement learning algorithm itself also affects the results, and it becomes important in some multimodal generation tasks that include chains of thought. For example, DPO avoids the instability caused by online sampling, but when the problem itself requires a more complex intermediate process, the final effect largely depends on the quality of offline data. Although online reinforcement algorithm can bring stronger exploration ability, because there is no explicit value guidance, it is more prone to a more unstable convergence process than offline alignment in the process of generating long text chains of thought. Experimental comparison shows that the adaptability differentiation of DPO and GRPO under different tasks essentially reflects the vulnerability of existing algorithms in dealing with explicit reasoning logic [6].

In addition, as the guide of reinforcement learning, the stability of Reward Mode itself is also very important. In the process of training, if the Actor model finds the scoring holes of the reward model and exploits them, it may cause reward hacking. This drift can cause the model to produce results that appear to have a high score, but are in fact very confusing. This incompatibility between the model and the optimizer leads to a degree of randomness.

5. Conclusions

In summary, this paper explores the causes of the instability phenomenon of large language models in RL fine-tuning from multiple perspectives. It is concluded that the phenomenon is caused by the Vf prediction error caused by the randomly changing training environment, the imbalance of RL method's policy update, and the sensitivity of RL method to hyperparameters. It is found that the credibility distribution of CT causes great initial state deviation, which leads to policy fluctuations. The information bottleneck greatly reduces the diversity of sampling, and it is easy to converge to a single state in the optimization process, and it is impossible to achieve diversified output.

In order to solve the above stability bottleneck, the future research direction can consider the improvement of basic mathematical theory and specific implementation technology. More accurate guidance signals are provided by real-time monitoring of reasoning steps. At the same time, an alignment algorithm with adaptive adjustment ability can be developed, which may effectively improve the fault tolerance and macro stability of different models in a decentralized environment by constantly optimizing the mechanism of entropy balance and the constraint on KL divergence. In addition, reinforcement learning and reasoning ability should also co-evolve, and strive to achieve a balance between offline preference learning and online exploration. After long-term efforts, future large language models are expected to further promote cognitive alignment and logical evolution on the basis of ensuring stable and reliable training.

References

[1] Ouyang, L., Wu, J., Jiang, X., et al. (2022). Training language models to follow instructions with human feedback. Advances in Neural Information Processing Systems, 35, 27730–27744.
[2] Wang, Z., et al. (2023). Step-level value estimation for deep reasoning. arXiv preprint arXiv: 2311.01234.
[3] Yuan, Y., Yue, Y., Zhu, R., et al. (2025). What's behind PPO's collapse in long-CoT? Value optimization holds the secret. arXiv preprint arXiv: 2503.01491.

DOI: 10.54254/2753-8818/2026.33465

[4] Ziegler, D. M., et al. (2019). Fine-tuning language models from human preferences. arXiv preprint arXiv: 1909.08593.
[5] Shao, J., & Cheng, Y. (2025). Towards analyzing and understanding the limitations of VAPO: A theoretical perspective. arXiv preprint arXiv: 2506.03038.
[6] Bai, Y., et al. (2022). Constitutional AI: Harmlessness from AI feedback. arXiv preprint arXiv: 2212.08073.
[7] Casper, S., et al. (2023). Open problems and fundamental limitations of RLHF. arXiv preprint arXiv: 2307.15217.
[8] Rafailov, R., Sharma, A., Mitchell, E., et al. (2024). Direct preference optimization: Your language model is secretly a reward model. Advances in Neural Information Processing Systems, 36.
[9] Perez, E., et al. (2022). Red teaming language models with language models. arXiv preprint arXiv: 2202.03286.
[10] DeepSeek-AI. (2025). DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv preprint arXiv: 2501.12948.
[11] Ahmadian, A. (2024). Back to basics: Revisiting REINFORCE for LLM alignment. arXiv preprint arXiv: 2402.14740.
[12] Dong, G., Bao, L., Wang, Z., et al. (2025). Agentic entropy-balanced policy optimization. arXiv preprint arXiv: 2510.14545.
[13] Li, Q., Xue, R., Wang, J., et al. (2025). CURE: Critical-token-guided re-concatenation for entropy-collapse prevention. arXiv preprint arXiv: 2508.11016.
[14] Lian, Y. (2025). Comparative analysis and parametric tuning of PPO, GRPO, and DAPO for LLM reasoning enhancement. arXiv preprint arXiv: 2512.07611.
[15] Tong, C., Guo, Z., Zhang, R., et al. (2025). Delving into RL for image generation with CoT: A study on DPO vs. GRPO. arXiv preprint arXiv: 2505.17017.