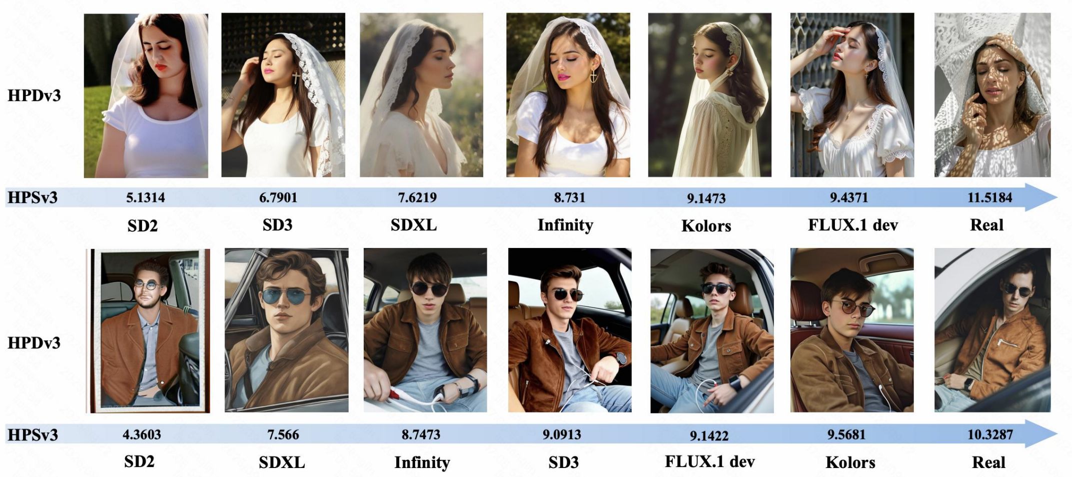
Reward Model

2025.9.25 by 刘恒霖

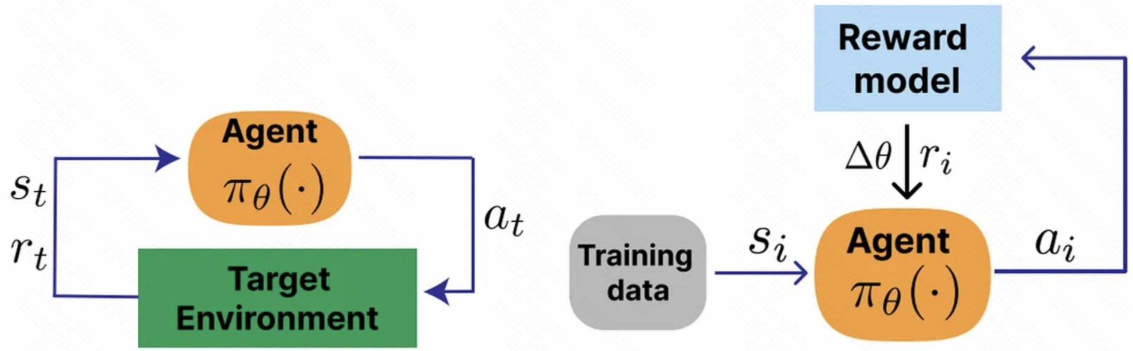
清华大学2024级硕士

作用

- 人类偏好的量化代理



- 作为强化学习的奖励信号

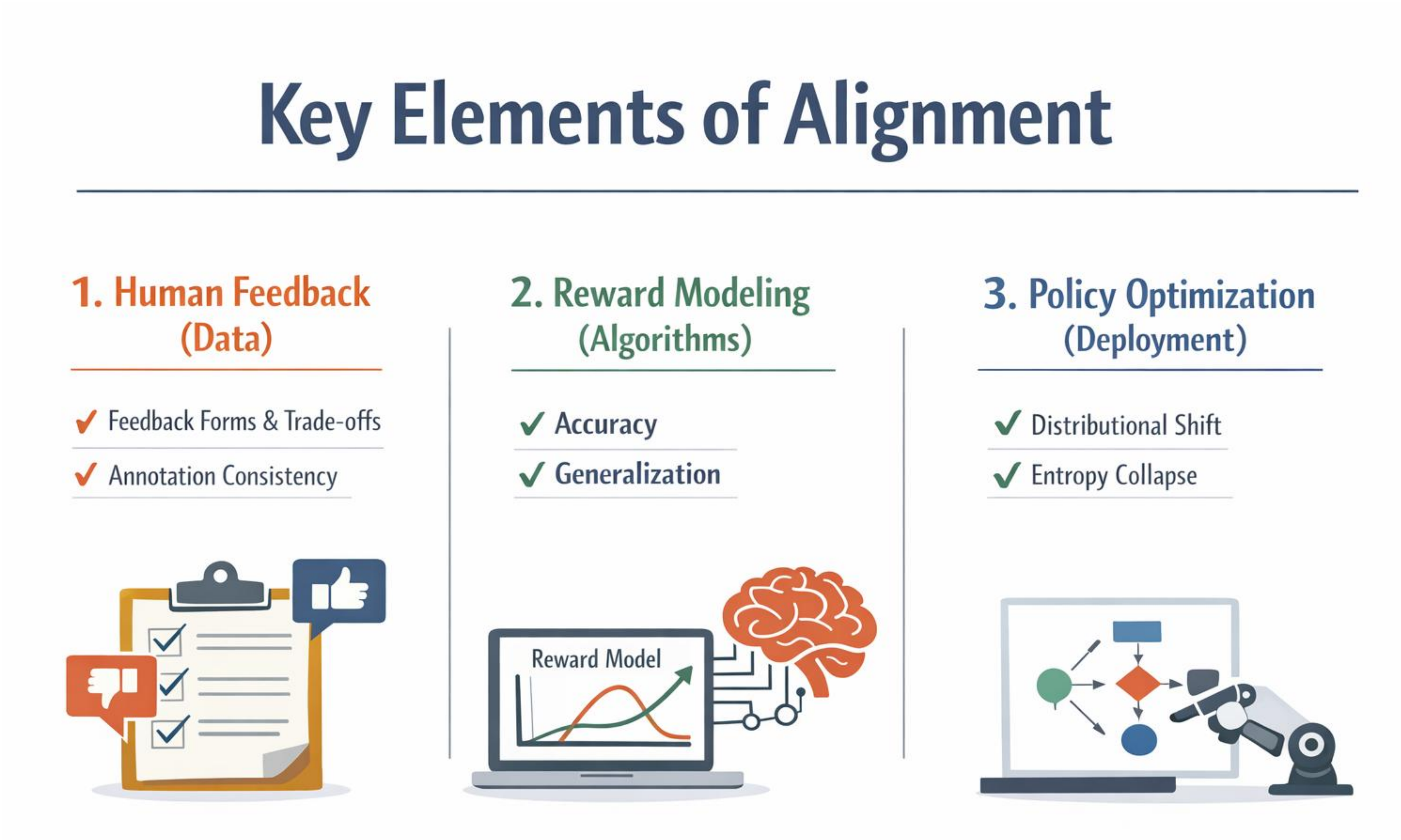


(a) Trial and error reinforcement learning.

(b) Reinforcement learning from human feedback.

Figure 2: Comparing RLHF to a traditional RL problem. (*left*) is the canonical RL problem, where an agent interacts repeatedly with an environment. (*right*) is RLHF, where an agent is optimized against a set of predetermined prompts and a learned reward model for one action per input state.

难点



Key Elements of Alignment

1. Human Feedback (Data)

- ✓ Feedback Forms & Trade-offs
- ✓ Annotation Consistency

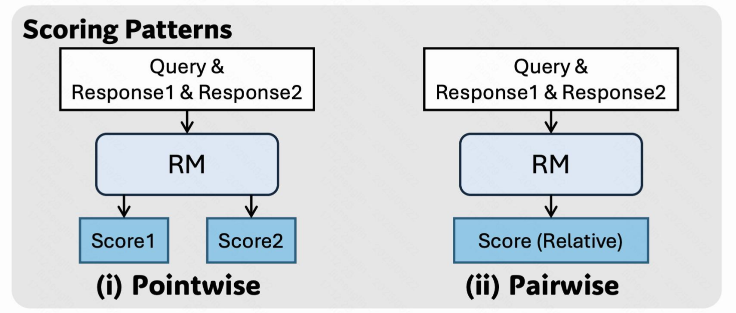
2. Reward Modeling (Algorithms)

- ✓ Accuracy
- ✓ Generalization

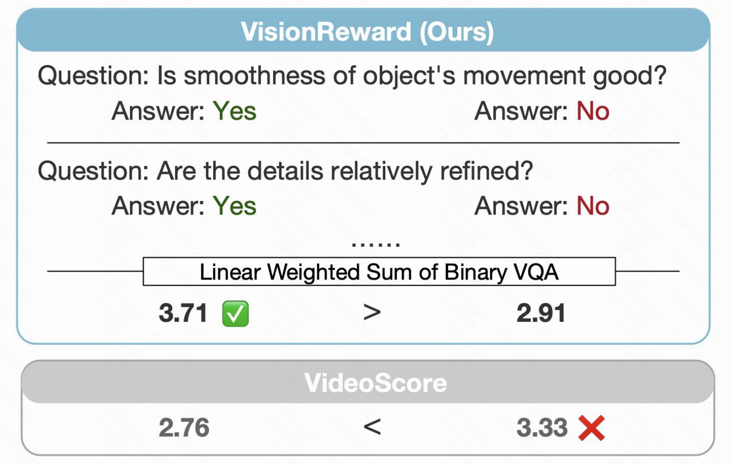
3. Policy Optimization (Deployment)

- ✓ Distributional Shift
- ✓ Entropy Collapse

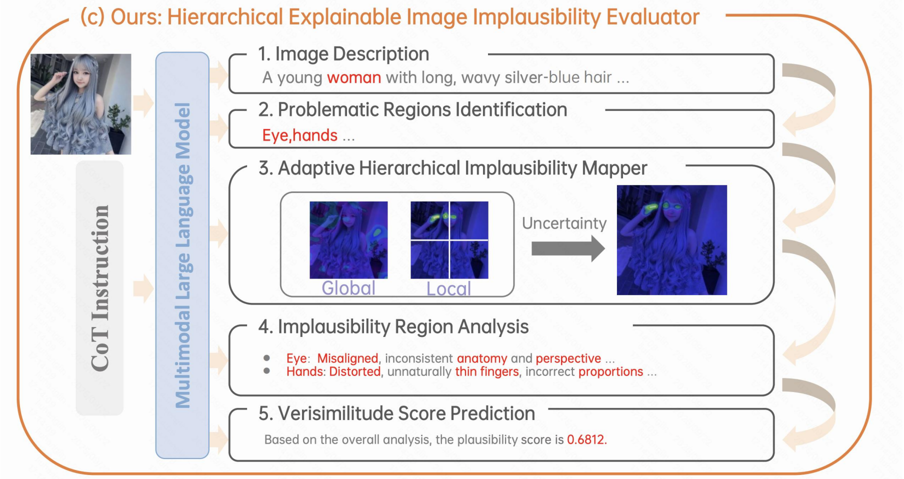
1. Human Feedback



(1) Point-wise/Pair-wise Score


VisionReward (Ours)
Question: Is smoothness of object's movement good?
Answer: Yes
Answer: No
Question: Are the details relatively refined?
Answer: Yes
Answer: No
......
Linear Weighted Sum of Binary VQA
3.71 ✅
>
2.91
VideoScore
2.76
<
3.33 ❌

(2) Bool question answer [1]


(c) Ours: Hierarchical Explainable Image Implausibility Evaluator
CoT Instruction
Multimodal Large Language Model
1. Image Description
A young woman with long, wavy silver-blue hair ...
2. Problematic Regions Identification
Eye,hands ...
3. Adaptive Hierarchical Implausibility Mapper
Global
Local
Uncertainty
4. Implausibility Region Analysis
Eye: Misaligned, inconsistent anatomy and perspective ...
Hands: Distorted, unnaturally thin fingers, incorrect proportions ...
5. Verisimilitude Score Prediction
Based on the overall analysis, the plausibility score is 0.6812.

(3) Text / Bounding box [2]

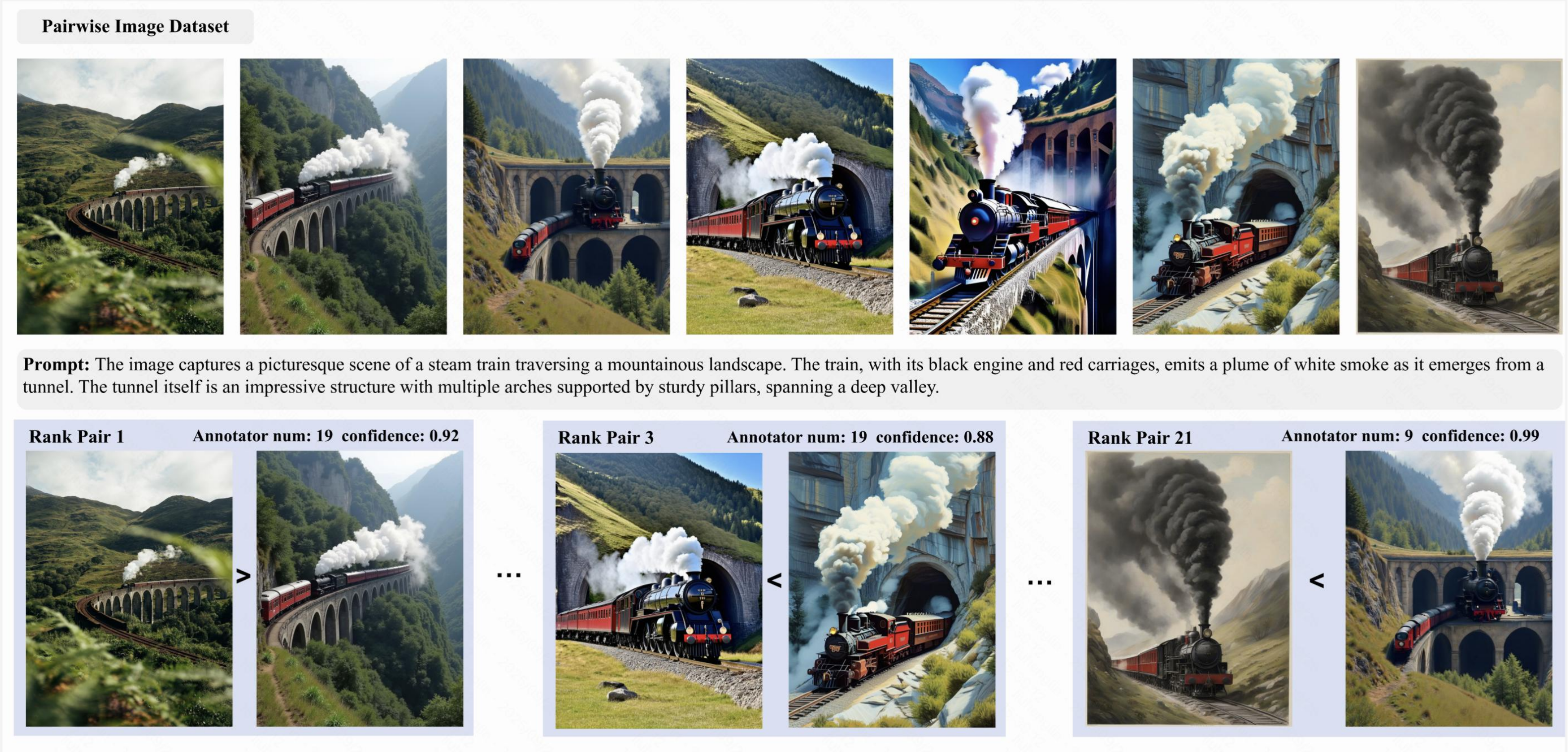
- 标量评分（如打分1-10）
  - ➢ 优点：表达强度
  - ➢ 缺点：评分标准更主观且人类难以把握，一致性更差
- 二元比较（如A回答优于B）
  - ➢ 优点：标注简单，一致性较高
  - ➢ 缺点：应用复杂
  - ➢ *例子：RewardDance[1] / Pref-GRPO[2]*

- ➢ 优点：反馈信息丰富
- ➢ 缺点：难以被直接被策略模型量化利用，标注成本高。
- ➢ 例子：*VisionReward[3] / LIFT[4]/ HEIE[5]*

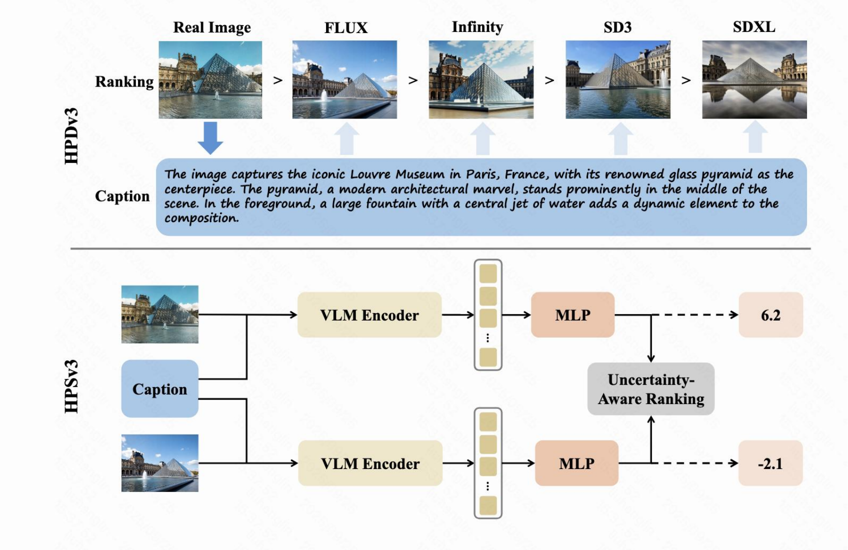
1. RewardDance: Reward Scaling in Visual Generation
2. Pref-GRPO: Pairwise Preference Reward-based GRPO for Stable Text-to-Image Reinforcement Learning
3. VisionReward: Fine-Grained Multi-Dimensional Human Preference Learning for Image and Video Generation
4. LiFT: Leveraging Human Feedback for Text-to-Video Model Alignment
5. HEIE: MLLM-Based Hierarchical Explainable AIGC Image Implausibility Evaluator

1. Human Feedback

不同的评估者可能对同一模型输出给出截然不同的评价，例如，LLM相关研究显示标注者之间的一致性率通常在63%至77%之间。在数据层面细粒度的收集人类的投票结果，作为标注的置信度，能够真实建模这种不一致性 [1]，有望能够缓解传统奖励模型在面对不一致的标签出现的过优化问题（即策略模型盲目地优化代理奖励）[2]


Pairwise Image Dataset
Rank Pair 1
Annotator num: 19 confidence: 0.92
>
...
Rank Pair 3
Annotator num: 19 confidence: 0.88
<
...
Rank Pair 21
Annotator num: 9 confidence: 0.99
<

Prompt: The image captures a picturesque scene of a steam train traversing a mountainous landscape. The train, with its black engine and red carriages, emits a plume of white smoke as it emerges from a tunnel. The tunnel itself is an impressive structure with multiple arches supported by sturdy pillars, spanning a deep valley.



The image captures the iconic Louvre Museum in Paris, France, with its renowned glass pyramid as the centerpiece. The pyramid, a modern architectural marvel, stands prominently in the middle of the scene. In the foreground, a large fountain with a central jet of water adds a dynamic element to the composition.

Figure 2. **An overview of HPDv3 and HPSv3.** HPDv3 integrates both real-world collected and generated images. HPSv3 employs a VLM backbone to extract rich semantic representations from images and captions, then utilizes uncertainty-aware ranking to effectively learn human preferences from paired comparison data.

[1] *HPSv3: Towards Wide-Spectrum Human Preference Score (ICCV 2025)*
[2] *Probabilistic Uncertain Reward Model*

2.Reward Modeling

动机：基于 CLIP、BLIP 架构的现有人类偏好奖励模型存在固有缺陷，会给仅涵盖文本描述的基础图像通常比包含更丰富细节和超出提示字面描述的视觉元素的高质量图像获得更高的分数，与人类真实审美偏好存在显著偏差 (包括基于人类偏好数据的微调变体，如Pickscore,ImageReward等经典模型)。

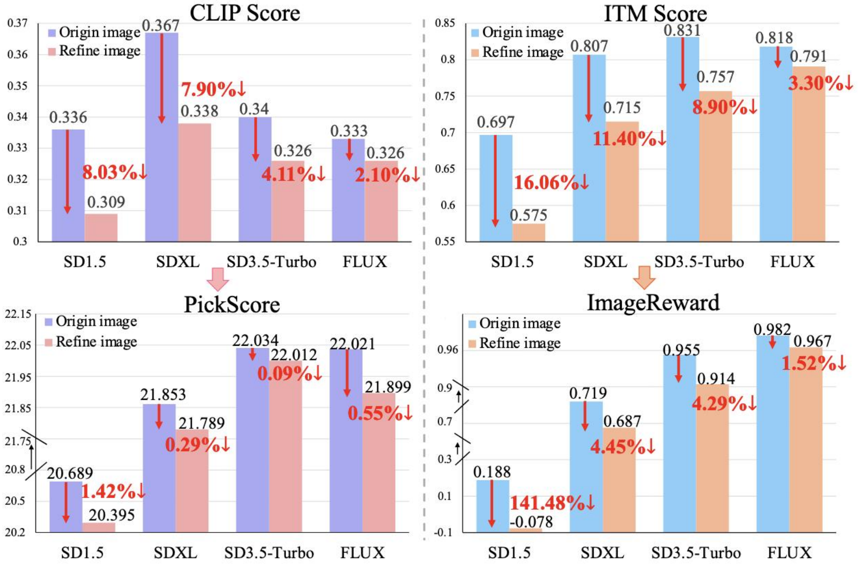


Figure 1. Reward Model Scoring Paradox Across Multiple Models Based on CLIP [26] (CLIP Score, PickScore) and BLIP [15] (ITM Score, ImageReward). Refine images are generated with refined prompts by LLMs.

现有奖励模型与人类真实偏好的偏差

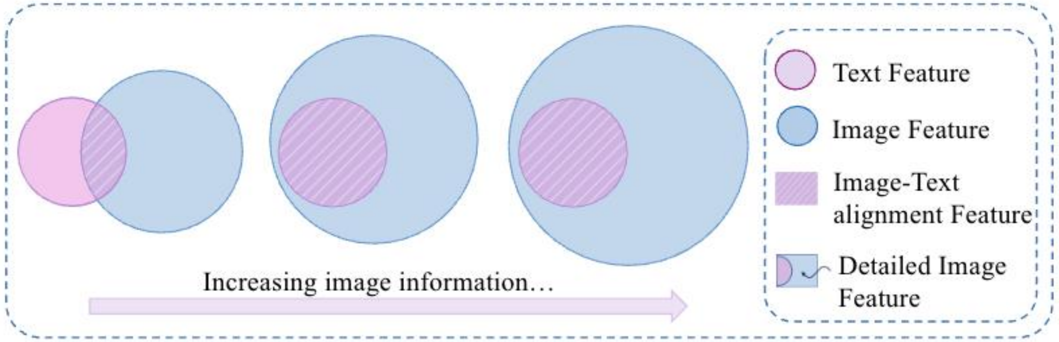


Figure 2. Text-Image Feature Interaction Venn Diagram: Increasing Image Information Under Fixed Text Prompts.

$$\mathrm{CLIP}(v,t) \approx \frac{I(v;t)}{\sqrt{I(t)\cdot I(v)}} = \frac{I(v;t)}{\sqrt{I(t)\cdot(I(v;t)+I(v|t))}}.$$

偏差机制分析

文本 - 图像
对齐相关

图像的额外
视觉信息

Enhancing Reward Models for High-quality Image Generation: Beyond Text-Image Alignment （ICCV 2025）

2.Reward Modeling

当图片和文本的一致程度达到阈值后，截断，从而利用 CLIP 模型在评估低质量和中等质量图像方面的有效性，同时避免了 CLIP 模型在高质量图像评分中的偏差。

做法：

1. 重构数据集，作为步骤2的标注：

高质量图片(I3)+复杂描述(P_{ref})：

高质量图片(I2)+简单描述(P_{easy})

高质量图片(I1)+简单描述(P_{easy})

2. 模型微调：

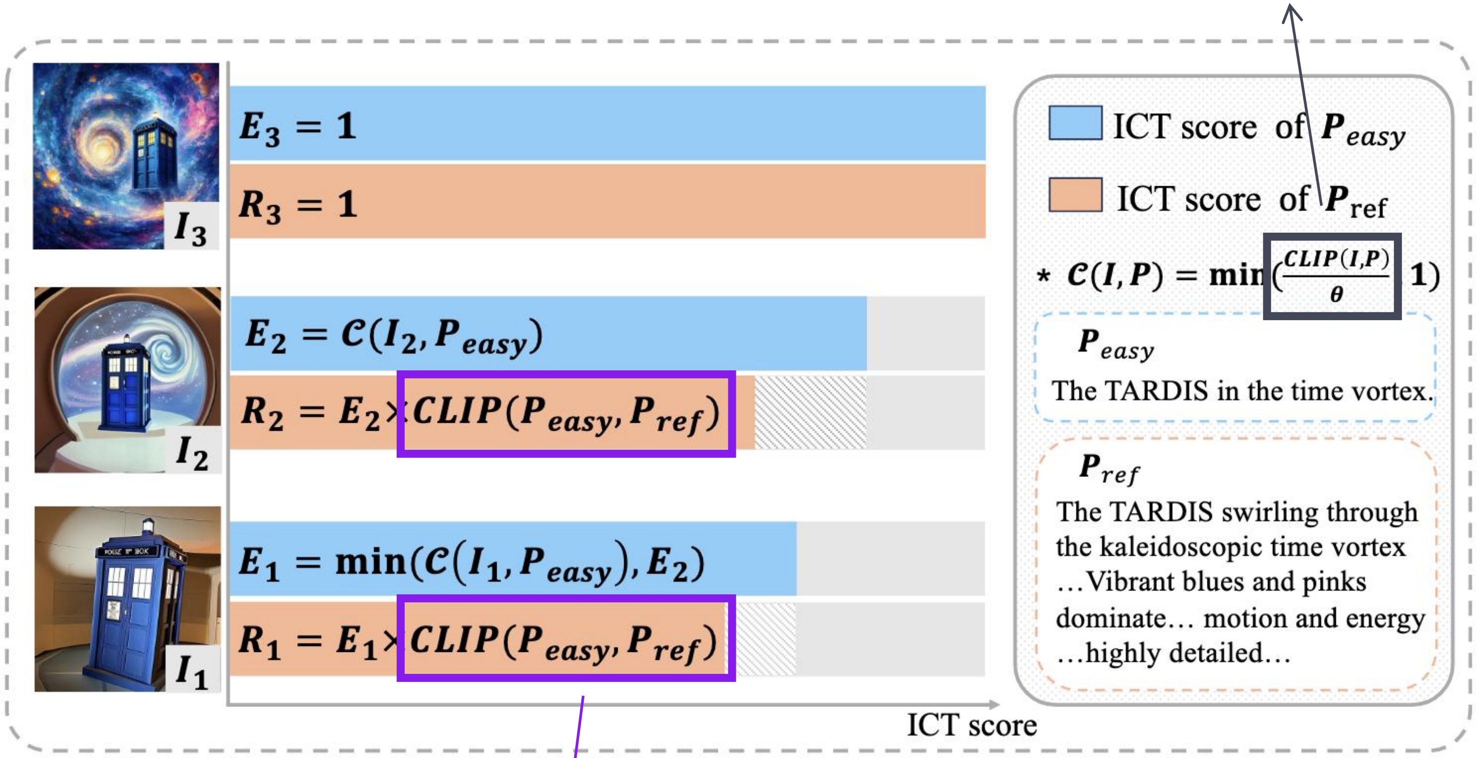
Image-Contained-Text Model

High·

$$\mathcal{L}_{\text{ICT}} = \sum_{i=1}^{3}(E_i - y_{i,e})^2 + \sum_{i=1}^{3}(R_i - y_{i,r})^2.$$

完全脱离文本模态，仅通过图像自身的视觉特征学习人类对 “高质量图像” 的偏好，成为 ICT 模型的 “美学补充评估工具”

$$\mathcal{L}_{\text{margin}} = \sum[\max(0, -\Delta(I_2, I_1) + m) + \max(0, -\Delta(I_3, I_2) + m)],$$



I_3
I_2
I_1
$E_3 = 1$
$R_3 = 1$
$E_2 = \mathcal{C}(I_2, P_{easy})$
$R_2 = E_2 \times CLIP(P_{easy}, P_{ref})$
$E_1 = \min(\mathcal{C}(I_1, P_{easy}), E_2)$
$R_1 = E_1 \times CLIP(P_{easy}, P_{ref})$
ICT score
ICT score of P_{easy}
ICT score of P_{ref}
$* \ \mathcal{C}(I, P) = \min(\frac{CLIP(I,P)}{\theta}, 1)$
P_{easy}
The TARDIS in the time vortex.
P_{ref}
The TARDIS swirling through the kaleidoscopic time vortex …Vibrant blues and pinks dominate… motion and energy …highly detailed…

Figure 3. The Image-Contained-Text (ICT) Scoring Framework.

通过 CLIP 模型计算的 “基础提示词与精细化提示词的语义相似度”（反映P_{ref} 在多大程度上是P_{easy} 的 “合理扩展”）

Enhancing Reward Models for High-quality Image Generation: Beyond Text-Image Alignment

2.Reward Modeling

动机：

- 传统回归模型：泛化性差、多数据集训练需感知尺度调整；
- 基于视觉语言模型的SFT方法标注成本高、易过拟合且输出僵化；
- 现有强化学习（RL）方法依赖数据集特定奖励设计，将质量视为绝对量，简化为回归任务。

做法：

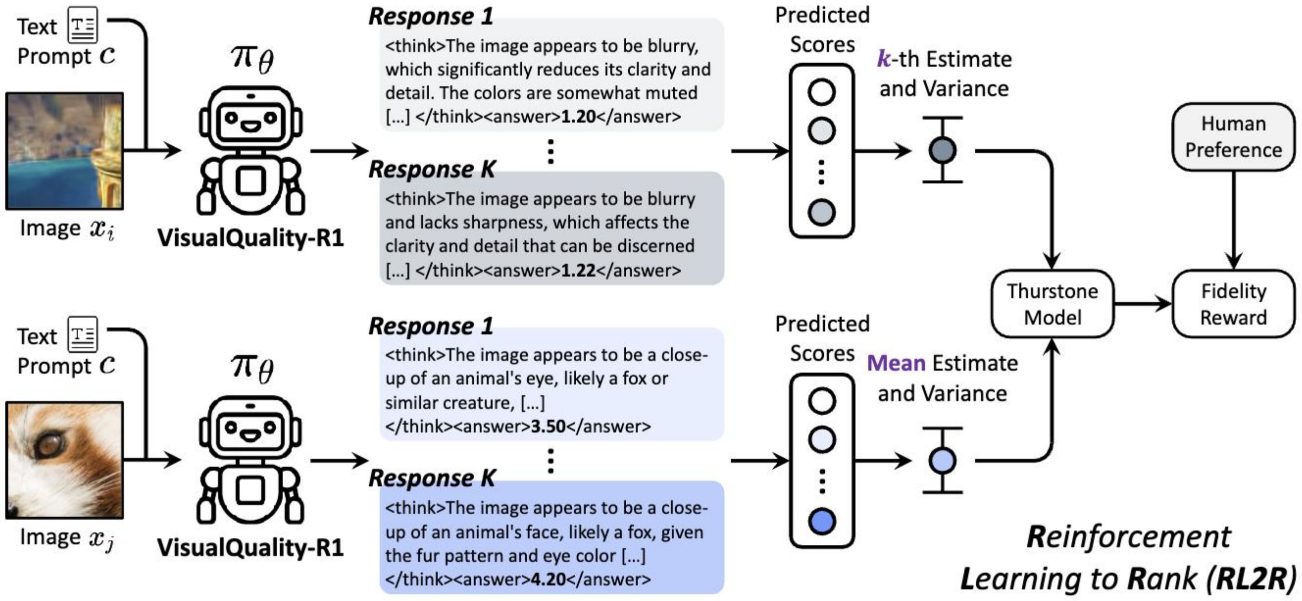
$$q(x_i) = [q_1(x_i), q_2(x_i), \ldots, q_K(x_i)]^\intercal$$

$$\boxed{p_k(x_i, x_j) = \Phi\left(\frac{q_k(x_i) - \mu(q(x_j))}{\sqrt{\sigma^2(q(x_i)) + \sigma^2(q(x_j)) + \gamma}}\right), \quad \text{for } i \neq j,}$$

相对性建模（Thurstone）：
x_i质量高于x_j的非对称比较概率

$$p(x, y) = \begin{cases} 1 & \text{if } \mathrm{MOS}(x) > \mathrm{MOS}(y) \\ 0.5 & \text{if } \mathrm{MOS}(x) = \mathrm{MOS}(y) \\ 0 & \text{otherwise} \end{cases}.$$

$$r_k(x_i) = \frac{1}{B-1} \sum_{j \neq i} \left(\sqrt{p(x_i, x_j) p_k(x_i, x_j)} + \sqrt{(1 - p(x_i, x_j))(1 - p_k(x_i, x_j))} \right).$$



<think>The image appears to be blurry, which significantly reduces its clarity and detail. The colors are somewhat muted [...] </think><answer>1.20</answer>

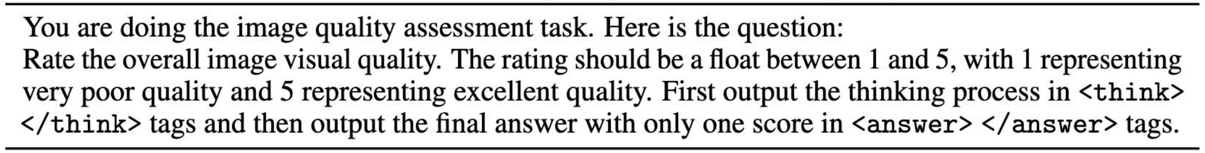
<think>The image appears to be blurry and lacks sharpness, which affects the clarity and detail that can be discerned [...] </think><answer>1.22</answer>

<think>The image appears to be a close-up of an animal's eye, likely a fox or similar creature, [...] </think><answer>3.50</answer>

<think>The image appears to be a close-up of an animal's face, likely a fox, given the fur pattern and eye color [...] </think><answer>4.20</answer>

Figure 2: System diagram of the proposed VisualQuality-R1 trained via RL2R. Given an image pair (x_i, x_j) with a shared text prompt c, VisualQuality-R1 generates K responses. Following GRPO [35], each response includes a detailed reasoning process and a predicted quality score. To assess relative visual quality, we calculate the asymmetric comparative probability that image x_i is perceived better than x_j under the Thurstone model [40]. This involves subtracting the mean predicted score of x_j from the k-th score of x_i, standardized by their added sample variance. A fidelity reward is derived from human preference, providing continuous supervisory signals for policy optimization.

Table 1: Structured text prompt used in VisualQuality-R1.

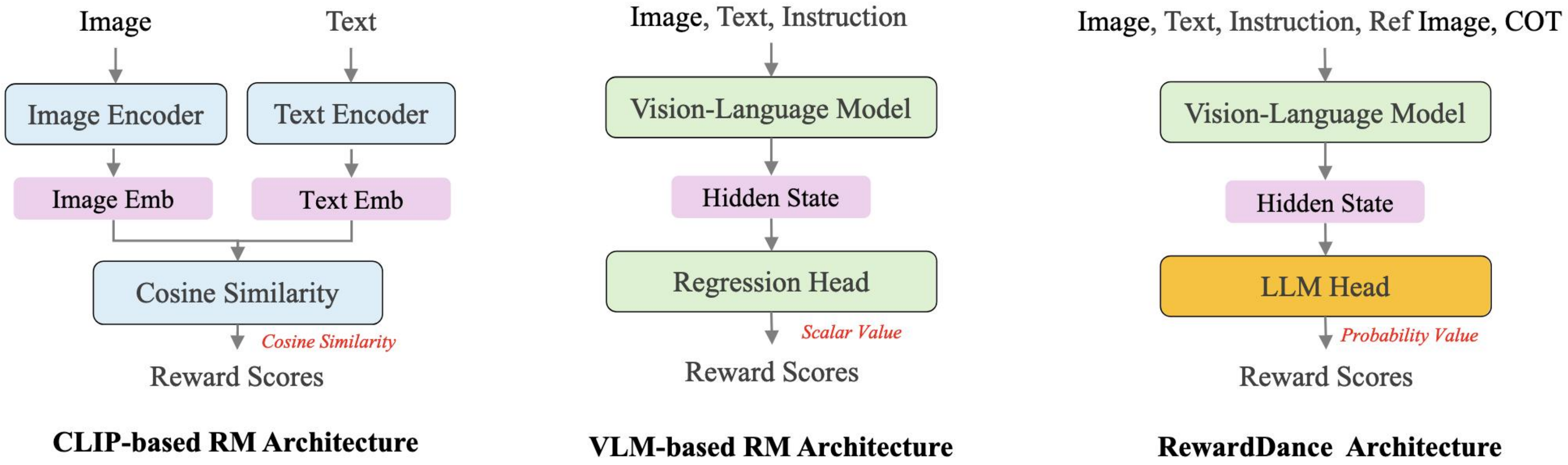


You are doing the image quality assessment task. Here is the question:
Rate the overall image visual quality. The rating should be a float between 1 and 5, with 1 representing very poor quality and 5 representing excellent quality. First output the thinking process in <think> </think> tags and then output the final answer with only one score in <answer> </answer> tags.

VisualQuality-R1: Reasoning-Induced Image Quality Assessment via Reinforcement Learning to Rank（NeurIPS 2025 Spotlight）

2.Reward Modeling

动机：视觉生成奖励模型存在 **扩展能力有限** 的问题：
1.架构限制：基于 CLIP 的奖励模型受双编码器架构与单模态设计制约，难以扩展且泛化性差；
2.范式错配：主流基于 Bradley-Terry 损失的回归型奖励模型，与视觉语言模型（VLM）的 next-token 预测机制不兼容，阻碍有效缩放；



CLIP-based RM Architecture **VLM-based RM Architecture** **RewardDance Architecture**

RewardDance: Reward Scaling in Visual Generation

2.Reward Modeling

动机：视觉生成奖励模型存在 **扩展能力有限**的问题：
1.架构限制：基于 CLIP 的奖励模型受双编码器架构与单模态设计制约，难以扩展且泛化性差；
2.范式错配：主流基于 Bradley-Terry 损失的回归型奖励模型，与视觉语言模型（VLM）的 next-token 预测机制不兼容，阻碍有效缩放；

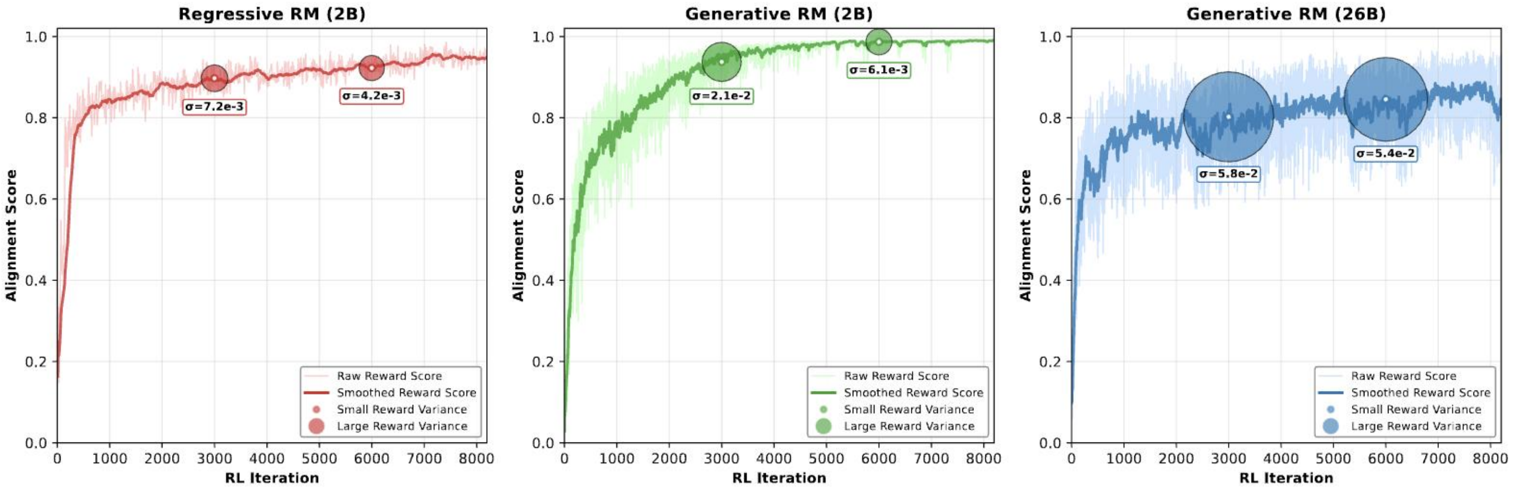


Figure 2 Comparison of training dynamics for Regressive vs. Generative reward models during diffusion RL fine-tuning. At the same 2B model scale (Left vs. Middle panel), the generative reward model exhibits significantly superior training dynamics compared to the regression-based one: it facilitates higher exploration magnitude, manifested as greater reward variance, and a more favorable reward growth trend. This higher diversity in reward signals indicates that the generative RM exhibits stronger robustness against reward hacking. Under a regression-based RM, the diffusion model risks learning to exploit reward loopholes to achieve high scores without making substantive progress. This inherent robustness is key to the generative RM's successful scaling to 26B parameters (Right panel).

2.Reward Modeling

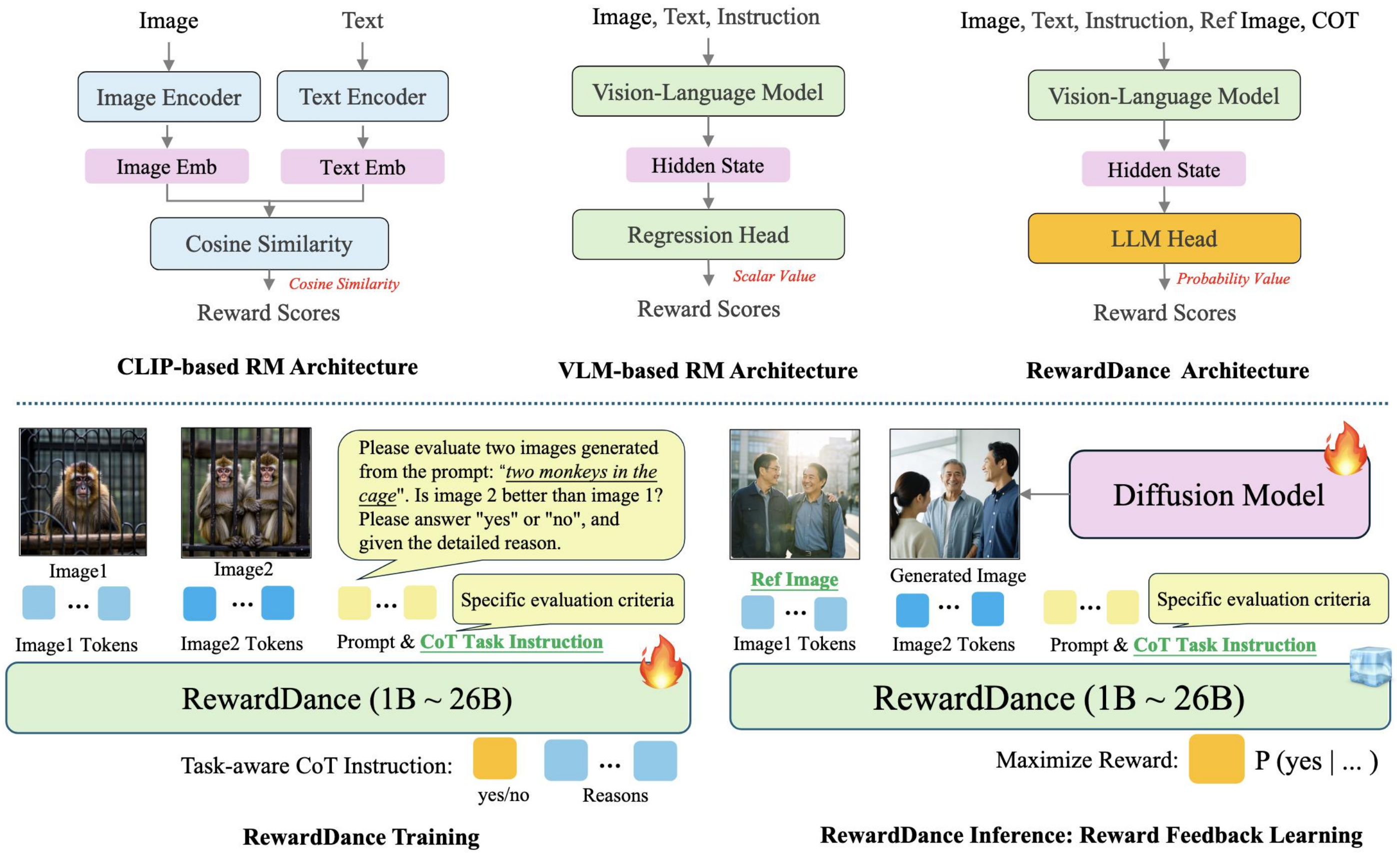
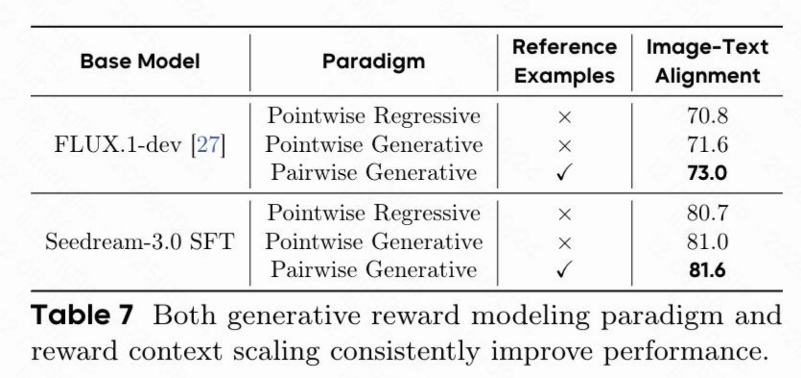


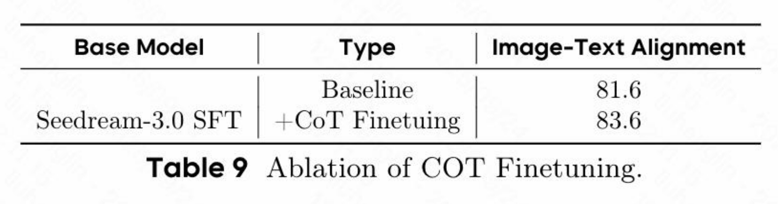
Figure 3 Overview of RewardDance framework compared with existing reward model Architecture. (Top): Previous works use CLIP-based or VLM-based reward models to provide scalar reward scores for diffusion model training. (Bottom): Our RewardDance approach (from 1B to 26B parameters) uses task-aware CoT instructions for reward modeling with reasoning. Red flames indicate trainable components; blue ice cubes indicate frozen parameters.



Base Model	Paradigm	Reference Examples	Image-Text Alignment
FLUX.1-dev [27]	Pointwise Regressive	×	70.8
	Pointwise Generative	×	71.6
	Pairwise Generative	✓	**73.0**
Seedream-3.0 SFT	Pointwise Regressive	×	80.7
	Pointwise Generative	×	81.0
	Pairwise Generative	✓	**81.6**

Table 7 Both generative reward modeling paradigm and reward context scaling consistently improve performance.

生成式 vs. 回归式



Base Model	Type	Image-Text Alignment
Seedream-3.0 SFT	Baseline	81.6
	+CoT Finetuning	83.6

Table 9 Ablation of COT Finetuning.

上下文缩放
（思维链）

1. 生成式奖励建模
将奖励分数重构为 VLM 预测 “yes” token 的概率 —— 即判断生成图像在特定标准下是否优于参考图像。

2. 双维度缩放机制
- 模型缩放：
首次将奖励模型参数从 10 亿系统性扩展至 260 亿，证明参数规模与奖励评估性能、最终生成质量呈强正相关；
- 上下文缩放：输入包含
    - 任务专属指令（如生成质量评估标准）
    - 参考图片
    - 思维链（CoT）推理数据

提升奖励判断的鲁棒性与准确性。

2.Reward Modeling

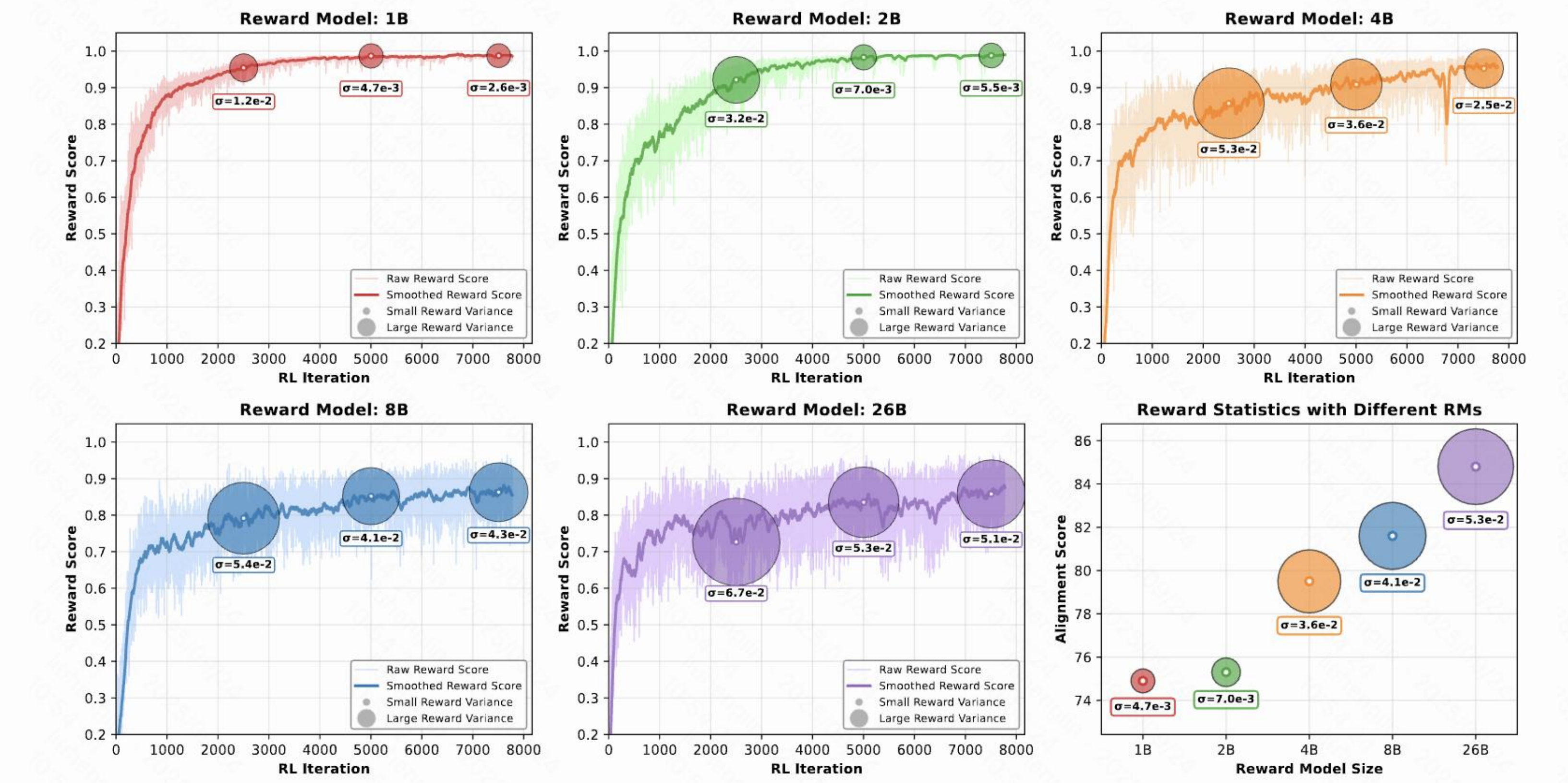


Figure 5 This figure shows the reward curves for Seedream during the RL stage, experiments with a separate reward model of varying sizes (1B to 26B). While reward scores consistently improve with more RL iterations across all models, a key trade-off emerges with RM size: larger RMs tend to exhibit a higher standard deviation, suggesting stronger robustness and less susceptibility to reward hacking.

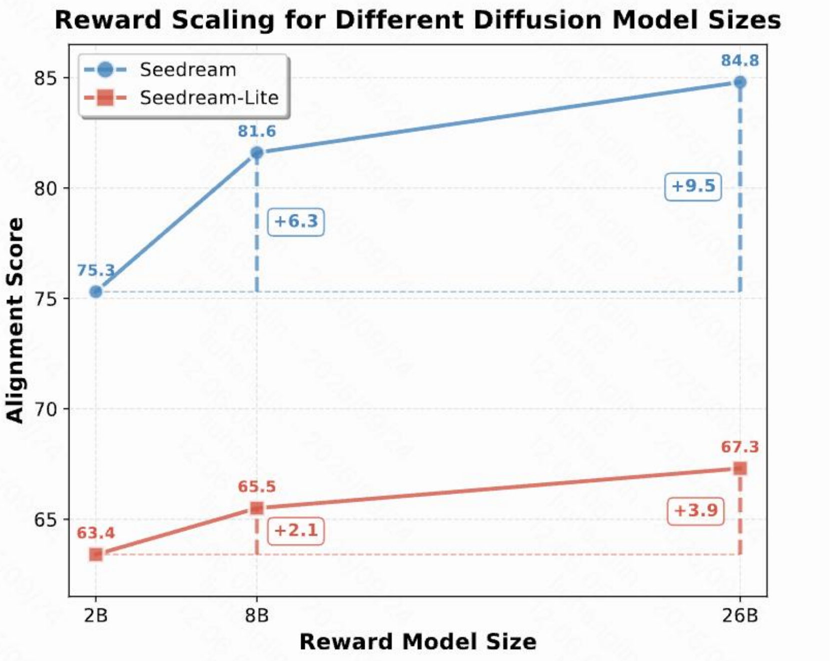


Figure 7 The performance of Seedream and Seedream-Lite models with different RM model sizes. While both models benefit from larger RMs, the performance improvement is substantially more pronounced for the larger model. This suggests that larger generative models require commensurately larger reward models to realize their full potential.

模型规模更大，整个训练过程（尤其是后期）有着更高的奖励方差, 表明该模型在训练后期仍能维持较强的探索能力。

更大规模的 DiT 架构能从奖励模型的缩放中获得更多收益，从而实现更显著的性能提升。

RewardDance: Reward Scaling in Visual Generation

2.Reward Modeling

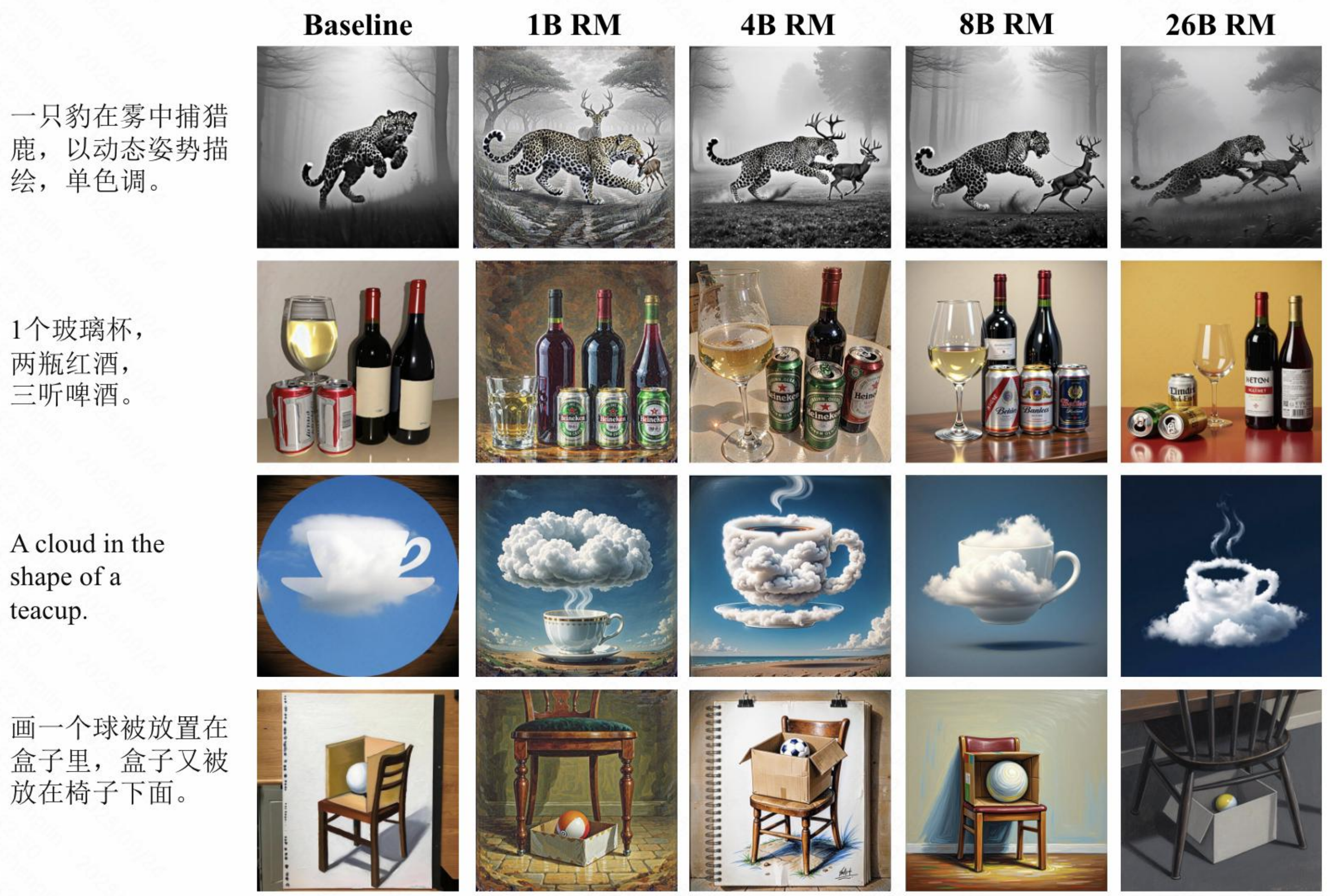

Baseline
1B RM
4B RM
8B RM
26B RM
一只豹在雾中捕猎鹿，以动态姿势描绘，单色调。
1个玻璃杯，两瓶红酒，三听啤酒。
A cloud in the shape of a teacup.
画一个球被放置在盒子里，盒子又被放在椅子下面。

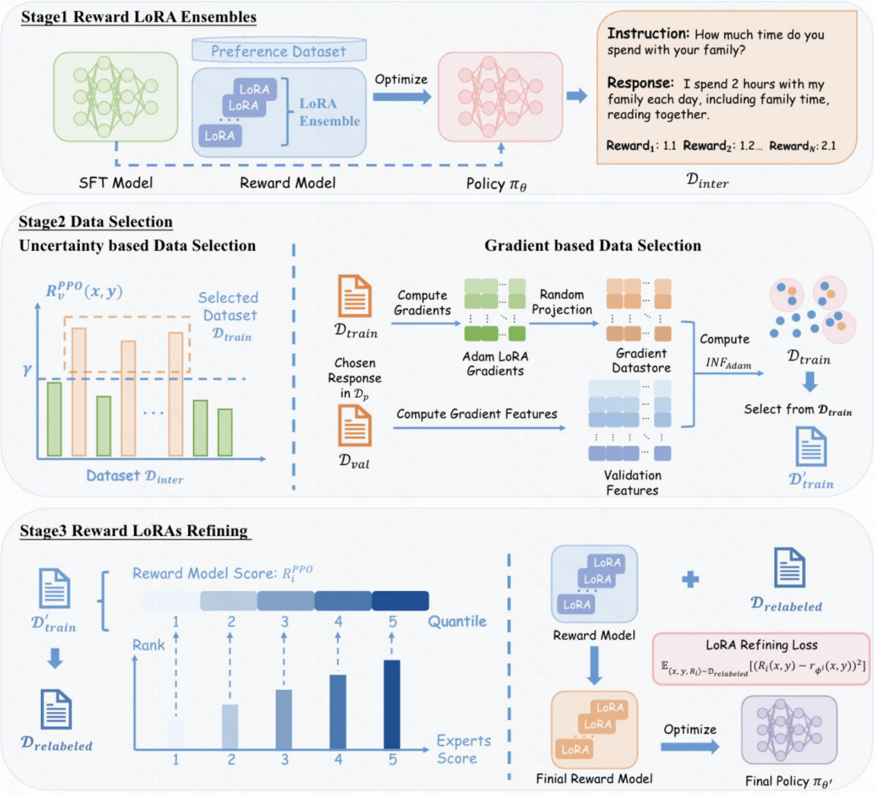
Figure 8 Text-to-Image generation comparison across reward models of increasing size (Baseline, 1B, 4B, 8B, 26B). Larger reward models demonstrate progressively better prompt adherence, visual quality, and semantic understanding.

RewardDance: Reward Scaling in Visual Generation

3.1 Policy Optimization 分布偏移

动机：

1. 奖励模型作为 “真实” 奖励函数的代理并不完美；
2. 策略优化会持续改变奖励模型训练数据的分布，导致固定奖励模型出现分布外问题，尤其在 on-policy 方法中；
3. 收集新的人类偏好数据可缓解该问题，但成本高、优化难度大。



Stage1 Reward LoRA Ensembles
Preference Dataset
LoRA
LoRA
LoRA
LoRA Ensemble
Optimize
SFT Model
Reward Model
Policy π_θ
$\mathcal{D}_{inter}$
Stage2 Data Selection
Uncertainty based Data Selection
$R_v^{PPO}(x, y)$
Selected Dataset $\mathcal{D}_{train}$
γ
Dataset $\mathcal{D}_{inter}$
Gradient based Data Selection
$\mathcal{D}_{train}$
Chosen Response in $\mathcal{D}_p$
$\mathcal{D}_{val}$
Compute Gradients
Adam LoRA Gradients
Random Projection
Gradient Datastore
Compute Gradient Features
Validation Features
Compute INF_{Adam}
$\mathcal{D}_{train}$
Select from $\mathcal{D}_{train}$
$\mathcal{D}'_{train}$
Stage3 Reward LoRAs Refining
$\mathcal{D}'_{train}$
$\mathcal{D}_{relabeled}$
Reward Model Score: R_i^{PPO}
1
2
3
4
5
Quantile
Rank
Experts Score
Reward Model
LoRA Refining Loss
$\mathbb{E}_{(x,y,R_i)\sim\mathcal{D}_{relabeled}}[(R_i(x,y) - r_{\phi^i}(x,y))^2]$
$\mathcal{D}_{relabeled}$
Finial Reward Model
Optimize
Final Policy $\pi_{\theta'}$

Instruction: How much time do you spend with your family?

Response: I spend 2 hours with my family each day, including family time, reading together.

Reward$_1$: 1.1 **Reward$_2$**: 1.2... **Reward$_N$**: 2.1

Figure 1: Overall training pipeline of our UGDA. The optimization is based on the Low-Rank Adaptation (LoRA) (Hu et al., 2021). We divide our pipeline into three stages, which are Reward LoRA Ensembles, Data Selection, Reward LoRAs Refining, respectively.

筛选关键数据+重新标注+更新奖励模型：

1.初始奖励模型构建

1.1 构建奖励模型由多个lora模型集成而来（不同lora模型的差异来源于训练随机性），用集成奖励分数的均值训练策略，用方差量化奖励模型的不确定性；

1.2 收集策略训练过程中的所有交互样本，为后续数据筛选做准备。

2.数据筛选

2.1 基于不确定性筛选：

•原因：奖励方差（不确定性）高的样本更可能帮助提升奖励模型性能。

•做法：本文奖励模型有多个lora分支。按奖励方差（不确定性）对交互样本排序，选取前 50% 高不确定性样本组成。

2.2 基于梯度影响筛选：

•原因：找到对优化policy model影响大的数据，重新标注这部分样本。

•做法：在上一步筛选出的数据中，进一步筛选50%。

$$INF_{Adam}(z, z') \triangleq \sum_{i=1}^{N} \overline{\eta}_i \frac{\langle \nabla\mathcal{L}(z';\theta_i), \Gamma(z,\theta_i)\rangle}{\|\nabla\mathcal{L}(z';\theta_i)\| \cdot \|\Gamma(z,\theta_i)\|}$$

3. 基于b中的数据及标注，微调奖励模型，再训policy model

UNCERTAINTY AND INFLUENCE AWARE REWARD MODEL REFINEMENT FOR REINFORCEMENT LEARNING FROM HUMAN FEEDBACK（ICLR 2025）

3.1 Policy Optimization 分布偏移

动机：

1. 奖励模型作为 “真实” 奖励函数的代理并不完美；
2. 策略优化会持续改变奖励模型训练数据的分布，导致固定奖励模型出现分布外问题，尤其在 on-policy 方法中；
3. 收集新的人类偏好数据可缓解该问题，但成本高、优化难度大。

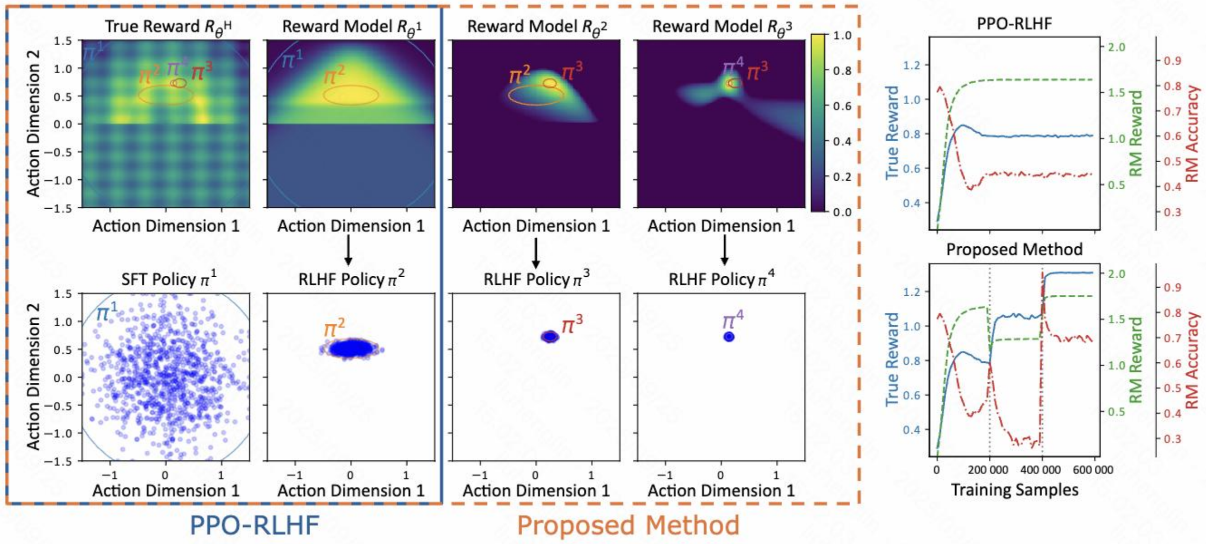


Figure 1: Visualization of our approach in a 2D stateless task. On the top-left side, the true reward function and learned RMs are visualized, on the lower-left side the SFT policy and learned policy distributions are shown. The colored circles contain 99% of each policy's action samples. The first RM R_{θ^1} is trained on data sampled from the SFT policy π^1 and thus performs well on pairs of actions sampled from π^1, but is inaccurate on samples generated by π^2. This leads the policy trained with standard PPO-RLHF to stagnate early (top-right). Using IW, after $k = 200,000$ samples, we train an off-policy corrected RM R_{θ^2} that is accurate on samples generated by π^2 and can thus be used to continue training and obtain a better policy π^3. Iterating this process, we obtain a better final policy (bottom-right).

重要性采样策略更新奖励模型（无需重新标注）：

做法：用重要性加权修正 奖励模型 的训练目标，将 RM 的训练目标从π_1分布修正为当前π_i分布，使每个阶段的 奖励模型（R_{θ_i}）都能适配当前策略（π_i）的分布，从而持续提供准确的奖励信号。

$$\mathcal{L}^{\pi^i}_{\mathrm{RM}}(\theta) = \mathbb{E}_{(s,\mathbf{a}_\mathrm{w},\mathbf{a}_\mathrm{l})\sim P_{\pi^i}}\left[l_{\mathrm{RM}}(s,\mathbf{a}_\mathrm{w},\mathbf{a}_\mathrm{l};\theta)\right] = \mathbb{E}_{(s,\mathbf{a}_\mathrm{w},\mathbf{a}_\mathrm{l})\sim P_{\pi^1}}\left[w(s,\mathbf{a}_\mathrm{w},\mathbf{a}_\mathrm{l})\,l_{\mathrm{RM}}(s,\mathbf{a}_\mathrm{w},\mathbf{a}_\mathrm{l};\theta)\right]$$

$$w(s,\mathbf{a}_\mathrm{w},\mathbf{a}_\mathrm{l}) = \frac{P(s)\pi^i(\mathbf{a}_\mathrm{w}\mid s)\pi^i(\mathbf{a}_\mathrm{l}\mid s)P(\mathbf{a}_\mathrm{w}>\mathbf{a}_\mathrm{l}\mid s)}{P(s)\pi^1(\mathbf{a}_\mathrm{w}\mid s)\pi^1(\mathbf{a}_\mathrm{l}\mid s)P(\mathbf{a}_\mathrm{w}>\mathbf{a}_\mathrm{l}\mid s)} = \frac{\pi^i(\mathbf{a}_\mathrm{w}\mid s)\pi^i(\mathbf{a}_\mathrm{l}\mid s)}{\pi^1(\mathbf{a}_\mathrm{w}\mid s)\pi^1(\mathbf{a}_\mathrm{l}\mid s)}$$

依据：若已知两个分布的概率比（即重要性权重），可将 “目标分布 P_{π_i}上的期望” 转换为 “原始分布 P_{π_1}上的加权期望”

Off-Policy Corrected Reward Modeling for Reinforcement Learning from Human Feedback（COLM 2025）

3.2 Policy Optimization 熵坍塌

挑战：强化学习，在训练时间扩展规模有限。

原因分析：

策略熵（所选动作中的可预测性或随机性）在早期训练阶段急剧下降->探索能力的减弱->性能饱和。

无kl约束限制下，策略熵与性能有定量关系，即受到熵耗尽的限制，且上限是完全可预测的。

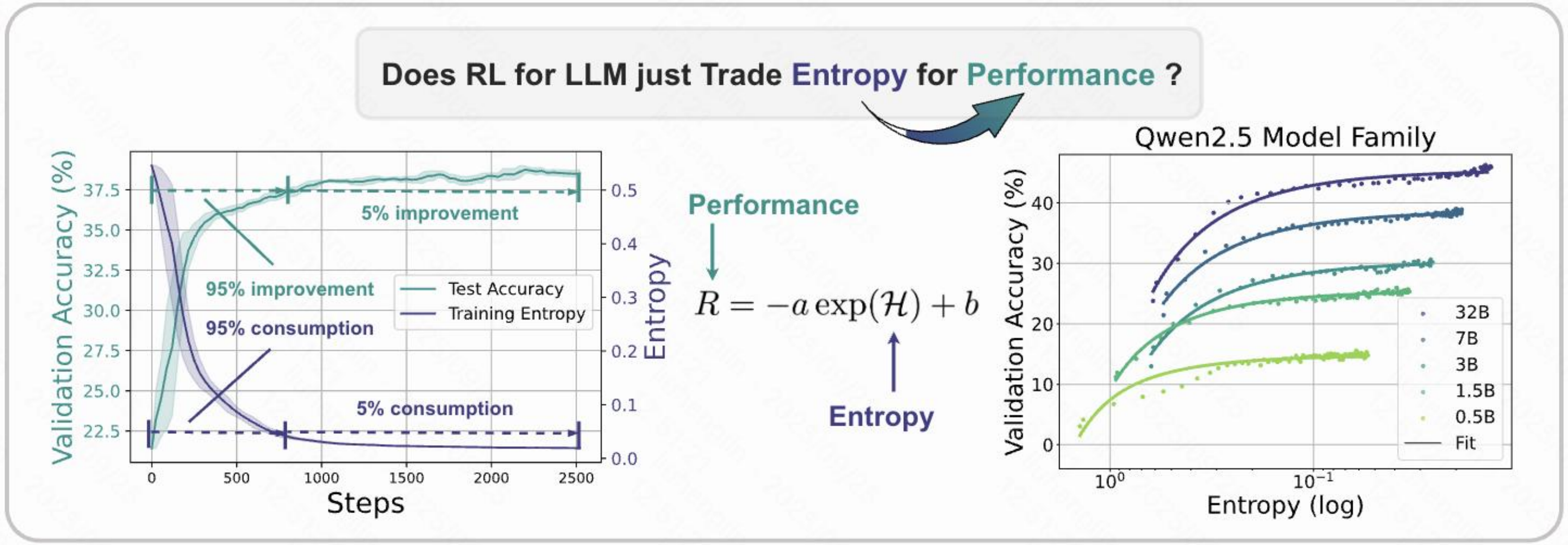


Figure 1: *Left:* Entropy collapse and performance saturation. Over 95% entropy drop/performance gains take place at the early stage of RL training. The model then reaches a plateau with little improvement. *Right:* The predictable relationship between validation performance and policy entropy. Without intervention, the policy “trades” entropy for performance exponentially, showing clear ceilings that hinder further policy enhancement.

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

3.2 Policy Optimization 熵坍塌

挑战：强化学习，在训练时间扩展规模有限。

原因分析：

策略熵（所选动作中的可预测性或随机性）在早期训练阶段急剧下降->探索能力的减弱->性能饱和。

无kl约束限制下，策略熵与性能有定量关系，即受到熵耗尽的限制，且上限是完全可预测的。

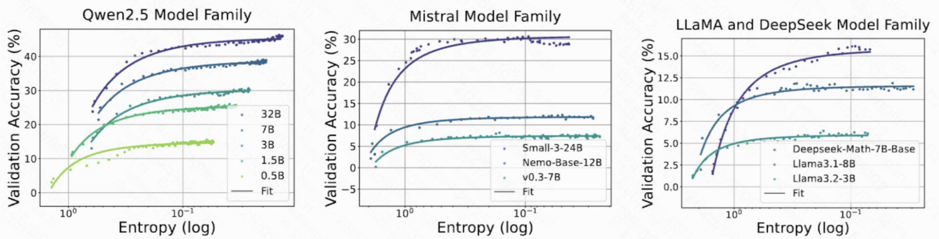


Figure 3: Fitting curves between policy entropy and validation performance on math task. We conduct validation every 4 rollout steps until convergence.

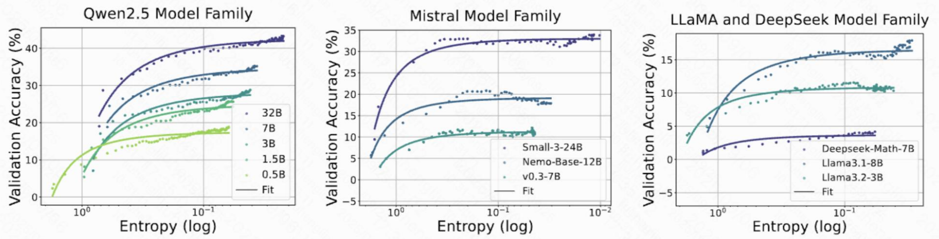


Figure 4: Fitting curves between policy entropy and validation performance in coding task. We conduct validation every 4 rollout steps until convergence.

不同模型架构&规模

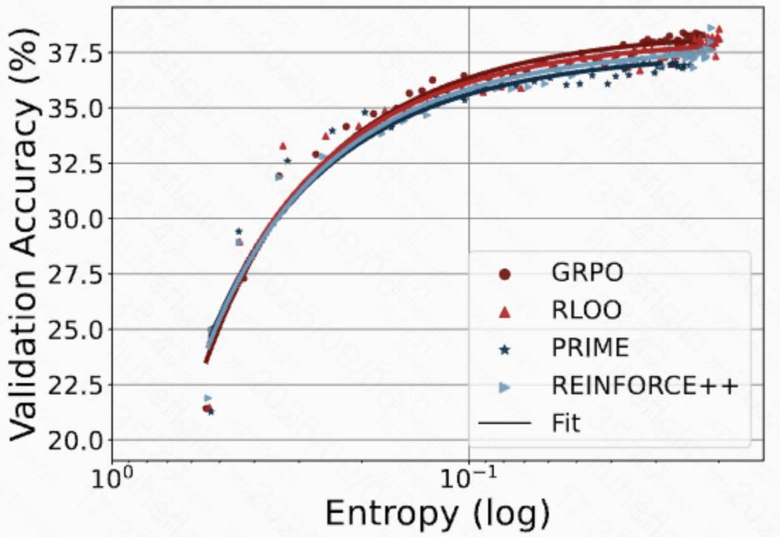


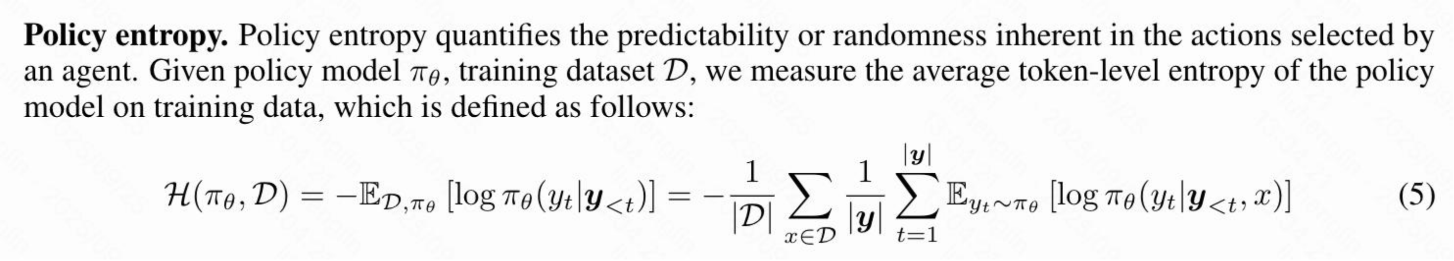
Figure 6: Training Qwen2.5-7B with different RL algorithms.

不同强化学习算法

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

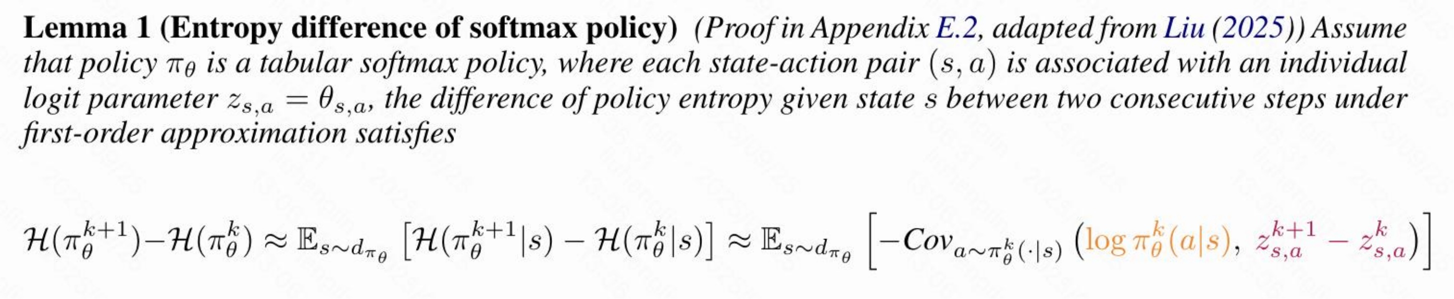
3.2 Policy Optimization 熵坍塌

策略熵为何会单调下降？理论分析



Policy entropy. Policy entropy quantifies the predictability or randomness inherent in the actions selected by an agent. Given policy model π_θ, training dataset $\mathcal{D}$, we measure the average token-level entropy of the policy model on training data, which is defined as follows:

$$\mathcal{H}(\pi_\theta,\mathcal{D}) = -\mathbb{E}_{\mathcal{D},\pi_\theta}\left[\log \pi_\theta(y_t|\boldsymbol{y}_{<t})\right] = -\frac{1}{|\mathcal{D}|}\sum_{x\in\mathcal{D}}\frac{1}{|\boldsymbol{y}|}\sum_{t=1}^{|\boldsymbol{y}|}\mathbb{E}_{y_t\sim\pi_\theta}\left[\log\pi_\theta(y_t|\boldsymbol{y}_{<t},x)\right] \qquad (5)$$

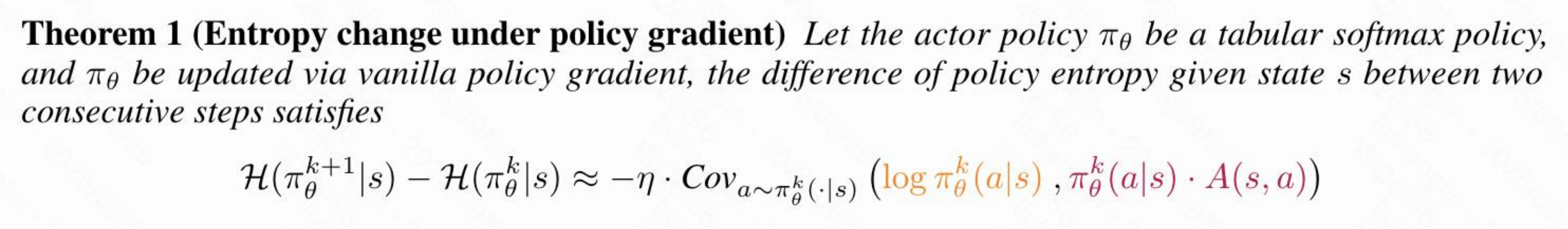


Lemma 1 (Entropy difference of softmax policy) *(Proof in Appendix E.2, adapted from Liu (2025)) Assume that policy π_θ is a tabular softmax policy, where each state-action pair (s,a) is associated with an individual logit parameter $z_{s,a}=\theta_{s,a}$, the difference of policy entropy given state s between two consecutive steps under first-order approximation satisfies*

$$\mathcal{H}(\pi_\theta^{k+1})-\mathcal{H}(\pi_\theta^{k}) \approx \mathbb{E}_{s\sim d_{\pi_\theta}}\left[\mathcal{H}(\pi_\theta^{k+1}|s)-\mathcal{H}(\pi_\theta^{k}|s)\right] \approx \mathbb{E}_{s\sim d_{\pi_\theta}}\left[-Cov_{a\sim\pi_\theta^k(\cdot|s)}\left(\log\pi_\theta^k(a|s),\ z_{s,a}^{k+1}-z_{s,a}^{k}\right)\right]$$

动作的对数概率 与 动作 logit 的变化量 的协方差

- 若高概率动作的 logit 持续增大，协方差为正，熵会随之下降；
- 若低概率动作的 logit 持续增大，协方差为负，熵会随之上升。



Theorem 1 (Entropy change under policy gradient) *Let the actor policy π_θ be a tabular softmax policy, and π_θ be updated via vanilla policy gradient, the difference of policy entropy given state s between two consecutive steps satisfies*

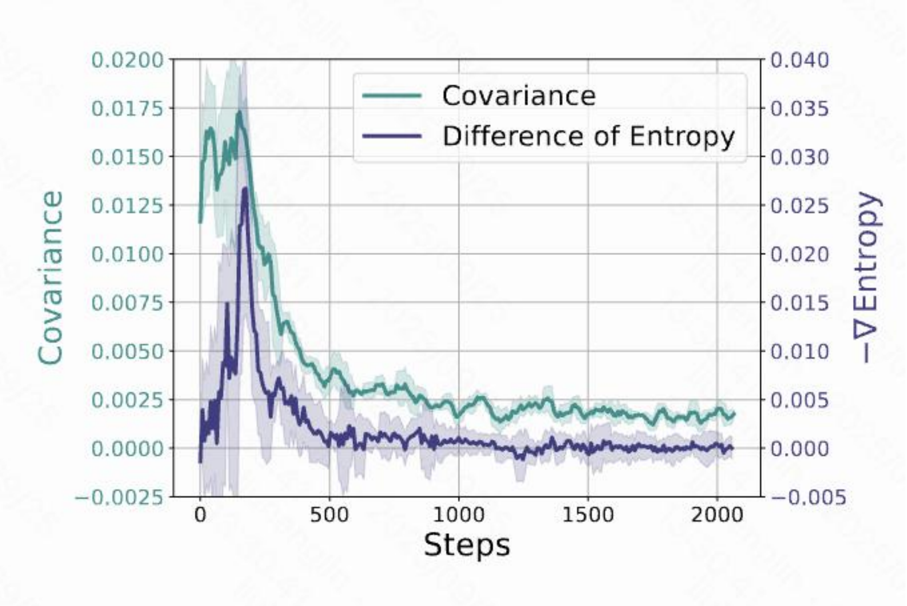
$$\mathcal{H}(\pi_\theta^{k+1}|s)-\mathcal{H}(\pi_\theta^{k}|s)\approx -\eta\cdot Cov_{a\sim\pi_\theta^k(\cdot|s)}\left(\log\pi_\theta^k(a|s)\ ,\pi_\theta^k(a|s)\cdot A(s,a)\right)$$

在policy gradient的算法下，
logit的变化量与概率*奖励成正比

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

3.2 Policy Optimization 熵坍塌

策略熵为何会单调下降？实验验证



-d (H) 和 Cov (・) 的实证曲线呈现出高度相似的动态变化

- 在训练初期，熵 H 快速下降，同时伴随着相对较大的正值 Cov (・);
- 随着强化学习训练的推进，熵的衰减速度减慢，Cov (・) 稳定在较低水平，这反映了策略的逐渐收敛。

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

3.2 Policy Optimization 熵坍塌

传统方法的局限性：

1. 直接添加熵损失
做法：在优化奖励的同时，保留一定的熵。

$$L_{total} = L_{policy} - \beta\ H(\pi_{\theta})$$

缺点：超参数敏感，没有一个超参数能同时实现“熵稳定” 和 “性能提升”

2. 模型分布约束
做法：策略模型和参考模型的KL惩罚

$$L_{total} = L_{policy} + \gamma D_{KL}(\pi_{ref}||\pi_{\theta})$$

缺点：稳定策略熵，但是损失性能。

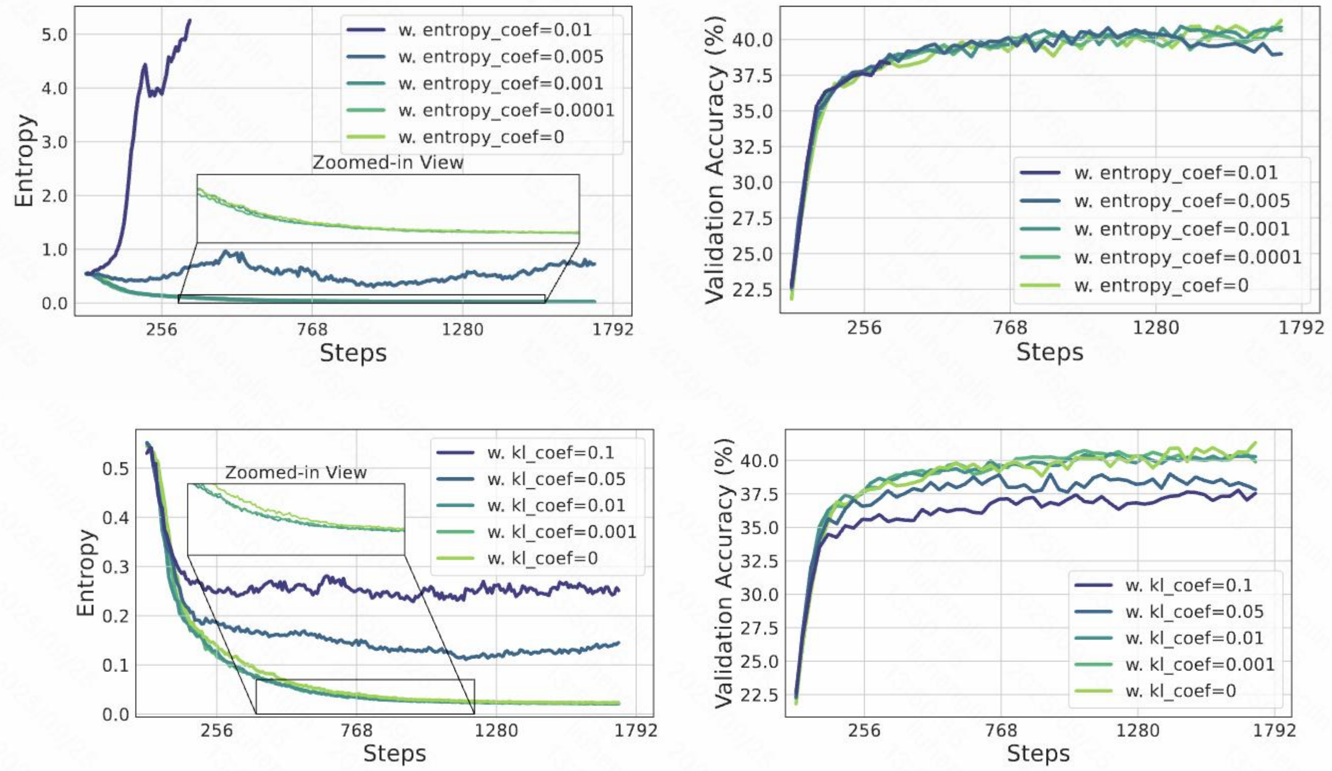
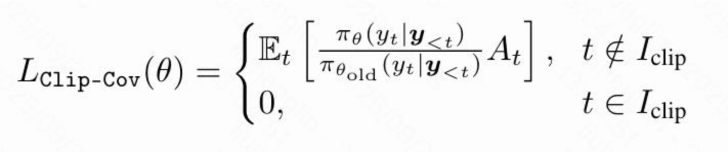


Figure 10: The policy entropy and validation accuracy of adding KL penalty between policy and reference model where $L_{\text{KL}} = L + \beta \mathbb{D}_{\text{KL}}(\pi_\theta||\pi_{\text{ref}})$. L is the original loss and β is the coefficient of KL loss.

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

3.2 Policy Optimization 熵坍塌

本文解决方案：抑制高协方差token的优化。
观察：“极少数高协方差 token 主导了熵坍缩”。
因此，只需针对性抑制这部分 token 的更新，即可有效控制熵，无需对所有 token 施加约束。



$$L_{\texttt{Clip-Cov}}(\theta) = \begin{cases} \mathbb{E}_t\left[\frac{\pi_\theta(y_t|\boldsymbol{y}_{<t})}{\pi_{\theta_{\text{old}}}(y_t|\boldsymbol{y}_{<t})}A_t\right], & t \notin I_{\text{clip}} \\ 0, & t \in I_{\text{clip}} \end{cases}$$

方案一：clip ratio

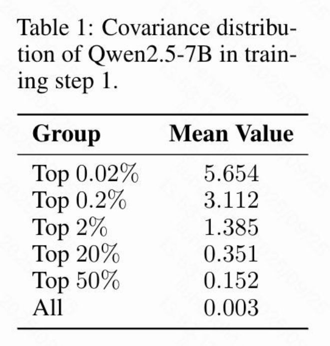
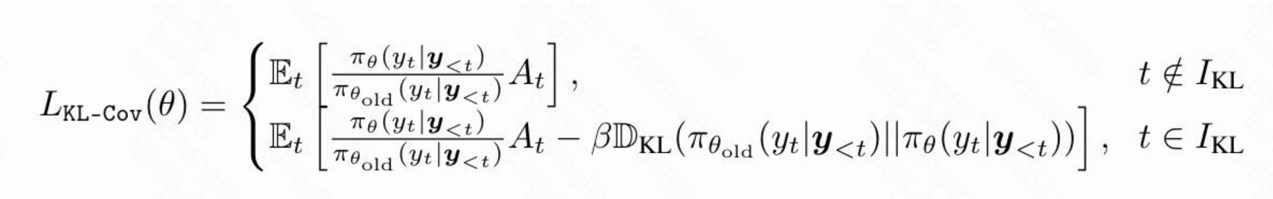


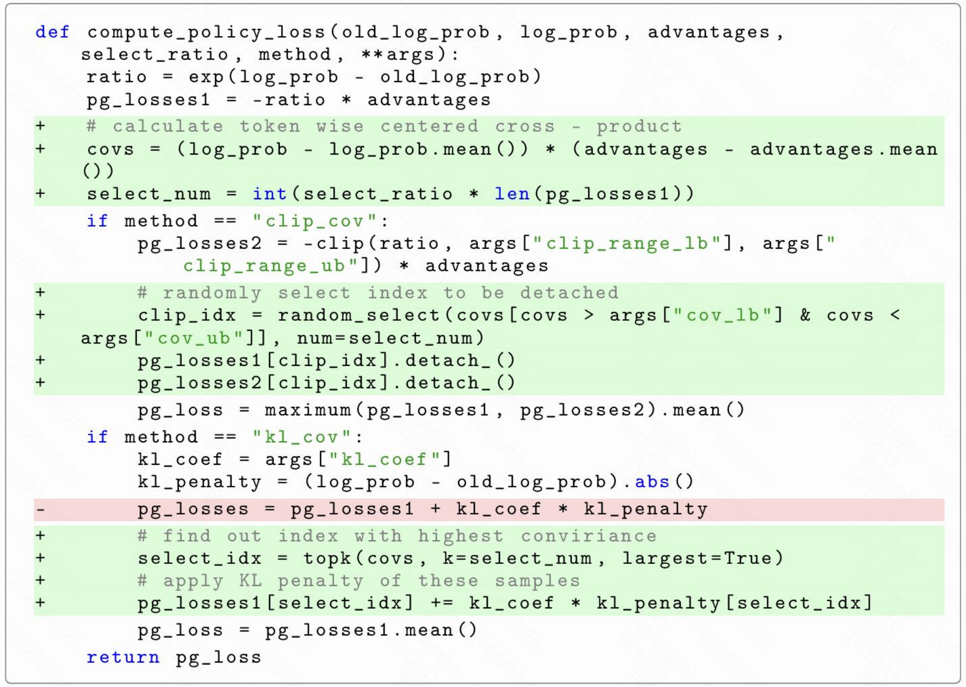
Table 1: Covariance distribution of Qwen2.5-7B in training step 1.

Group	Mean Value
Top 0.02%	5.654
Top 0.2%	3.112
Top 2%	1.385
Top 20%	0.351
Top 50%	0.152
All	0.003



$$L_{\texttt{KL-Cov}}(\theta) = \begin{cases} \mathbb{E}_t\left[\frac{\pi_\theta(y_t|\boldsymbol{y}_{<t})}{\pi_{\theta_{\text{old}}}(y_t|\boldsymbol{y}_{<t})}A_t\right], & t \notin I_{\text{KL}} \\ \mathbb{E}_t\left[\frac{\pi_\theta(y_t|\boldsymbol{y}_{<t})}{\pi_{\theta_{\text{old}}}(y_t|\boldsymbol{y}_{<t})}A_t - \beta\mathbb{D}_{\text{KL}}(\pi_{\theta_{\text{old}}}(y_t|\boldsymbol{y}_{<t})||\pi_\theta(y_t|\boldsymbol{y}_{<t}))\right], & t \in I_{\text{KL}} \end{cases}$$

方案二：+KL



```
def compute_policy_loss(old_log_prob, log_prob, advantages,
    select_ratio, method, **args):
    ratio = exp(log_prob - old_log_prob)
    pg_losses1 = -ratio * advantages
+   # calculate token wise centered cross - product
+   covs = (log_prob - log_prob.mean()) * (advantages - advantages.mean
    ())
+   select_num = int(select_ratio * len(pg_losses1))
    if method == "clip_cov":
        pg_losses2 = -clip(ratio, args["clip_range_lb"], args["
            clip_range_ub"]) * advantages
+       # randomly select index to be detached
+       clip_idx = random_select(covs[covs > args["cov_lb"] & covs <
    args["cov_ub"]], num=select_num)
+       pg_losses1[clip_idx].detach_()
+       pg_losses2[clip_idx].detach_()
        pg_loss = maximum(pg_losses1, pg_losses2).mean()
    if method == "kl_cov":
        kl_coef = args["kl_coef"]
        kl_penalty = (log_prob - old_log_prob).abs()
-       pg_losses = pg_losses1 + kl_coef * kl_penalty
+       # find out index with highest conviriance
+       select_idx = topk(covs, k=select_num, largest=True)
+       # apply KL penalty of these samples
+       pg_losses1[select_idx] += kl_coef * kl_penalty[select_idx]
        pg_loss = pg_losses1.mean()
    return pg_loss
```

Listing 1: The pseudo-code of the policy loss computation with Clip-Cov and KL-Cov. The implementation only need to modify several lines of code.

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

3.2 Policy Optimization 熵坍塌

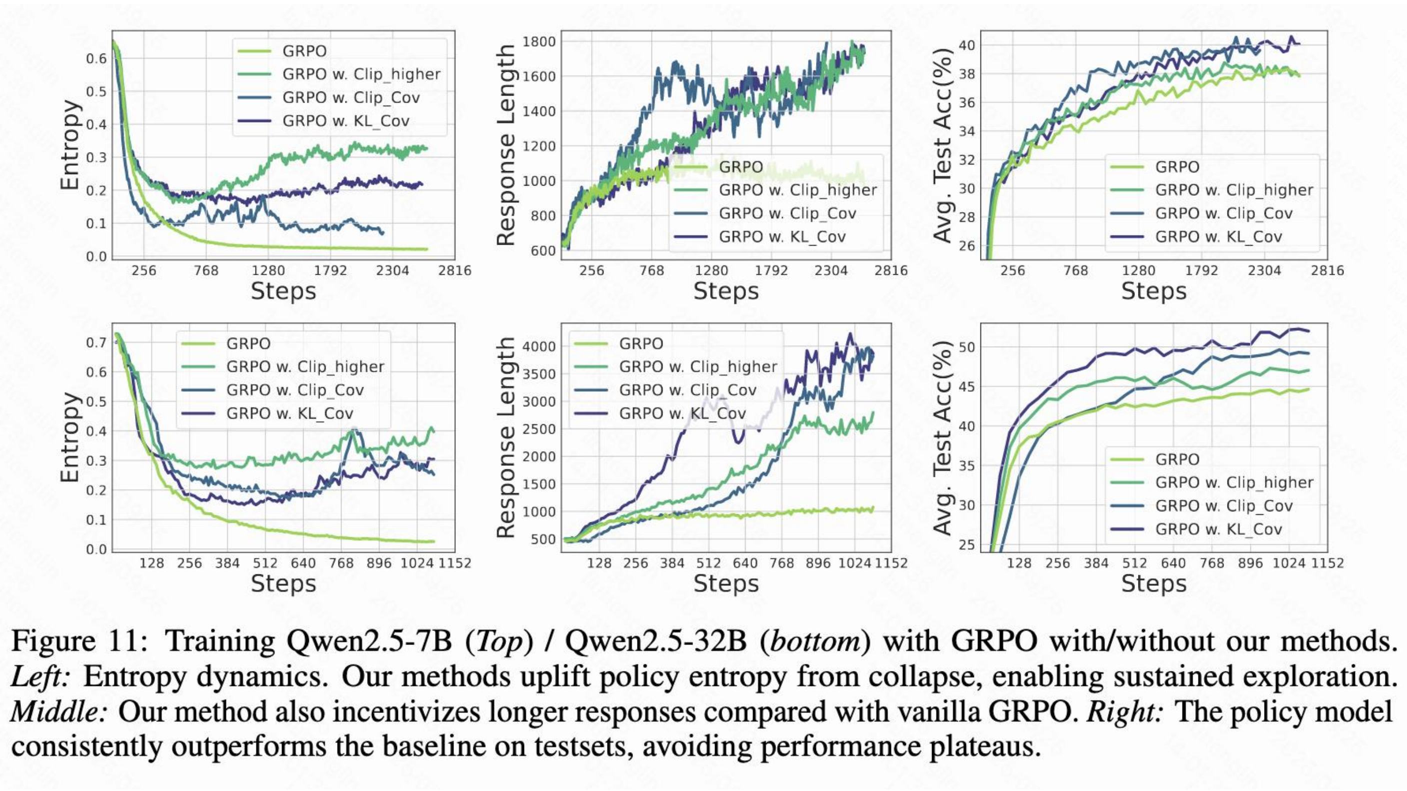


Figure 11: Training Qwen2.5-7B (*Top*) / Qwen2.5-32B (*bottom*) with GRPO with/without our methods. *Left:* Entropy dynamics. Our methods uplift policy entropy from collapse, enabling sustained exploration. *Middle:* Our method also incentivizes longer responses compared with vanilla GRPO. *Right:* The policy model consistently outperforms the baseline on testsets, avoiding performance plateaus.

熵曲线更稳定，性能持续提升

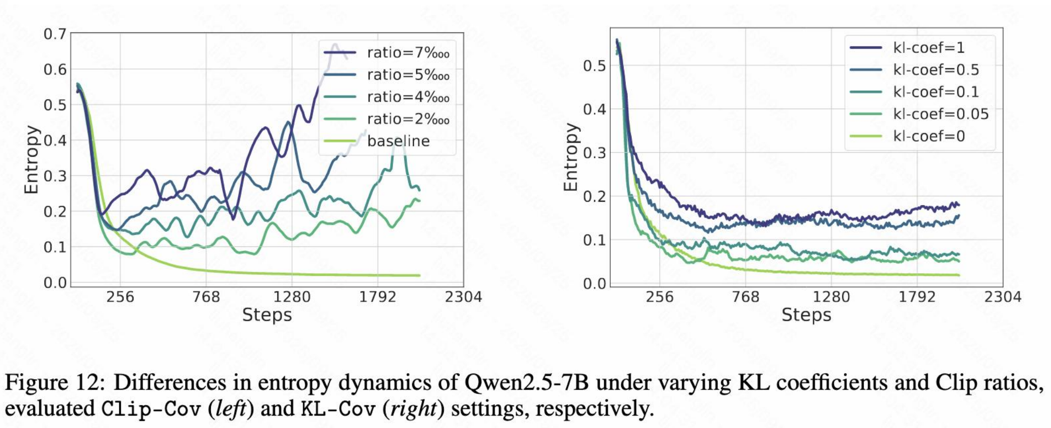
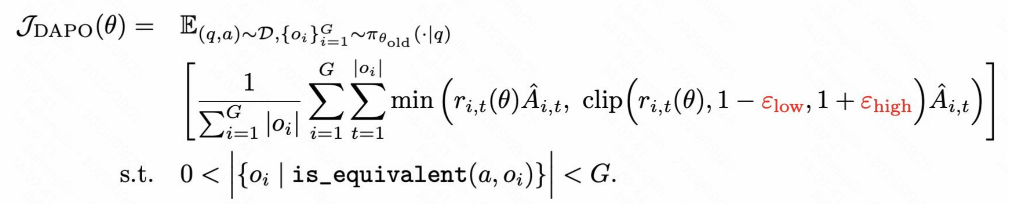


Figure 12: Differences in entropy dynamics of Qwen2.5-7B under varying KL coefficients and Clip ratios, evaluated Clip-Cov (*left*) and KL-Cov (*right*) settings, respectively.

KL方法参数鲁棒：只需调整超参数即可实现对策略熵的控制，进而能够引导熵的变化。



$$\mathcal{J}_{\text{DAPO}}(\theta) = \mathbb{E}_{(q,a)\sim\mathcal{D},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{\sum_{i=1}^{G}|o_i|}\sum_{i=1}^{G}\sum_{t=1}^{|o_i|}\min\Big(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\Big(r_{i,t}(\theta),1-\varepsilon_{\text{low}},1+\varepsilon_{\text{high}}\Big)\hat{A}_{i,t}\Big)\right]$$

$$\text{s.t.}\quad 0 < \Big|\{o_i \mid \texttt{is\_equivalent}(a,o_i)\}\Big| < G.$$

Clip-higher*

The Entropy Mechanism of Reinforcement Learning for Reasoning Language Models

* *DAPO: An Open-Source LLM Reinforcement Learning System at Scale*

3.2 Policy Optimization 熵坍塌

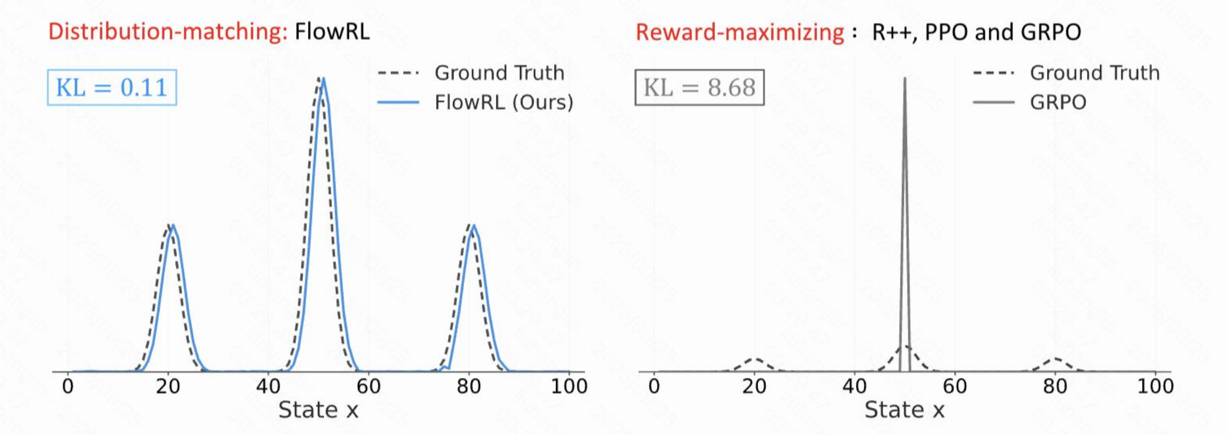
1. 核心问题：
奖励最大化类强化学习方法（如PPO、GRPO）存在“模式坍缩”问题。它们会过度拟合奖励分布中出现频率最高的“主导模式”，而忽略其他出现频率较低但同样有效的模式。

2. 导致的后果：
这引发了两个关键缺陷：
- 推理路径多样性不足：生成的解决方案同质化严重，缺乏多样的解题思路。

- 泛化能力弱：模型难以适应和处理那些小众但正确的逻辑，在思维链等需要多样化推理的场景中表现不佳。

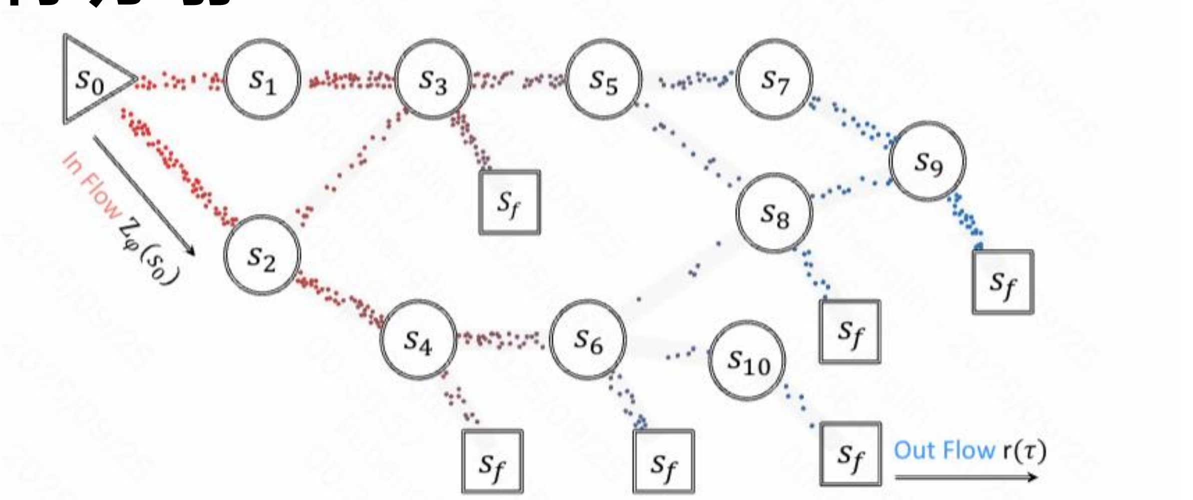
3. 现有解决方案的局限与核心挑战：
- 现有方法：如调整裁剪比例、增加熵奖励等，只是通过间接调整数据分布来提升多样性，属于治标不治本的优化。

- 根本挑战：如何从算法目标层面进行创新，在RL训练中从根本上促进多样化探索，避免策略过早收敛到单一的解决方案模式。如何在 RL 训练中促进多样化探索，避免策略收敛到单一的主导解决方案模式？



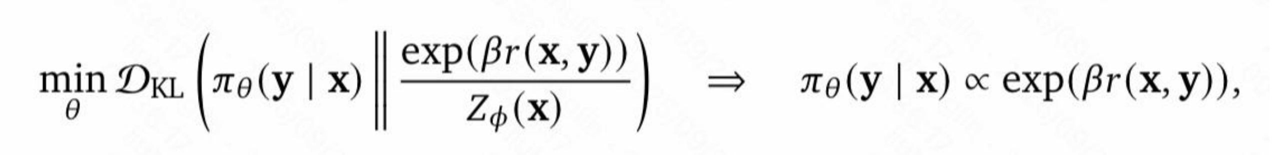
FlowRL: Matching Reward Distributions for LLM Reasoning

3.2 Policy Optimization 熵坍塌



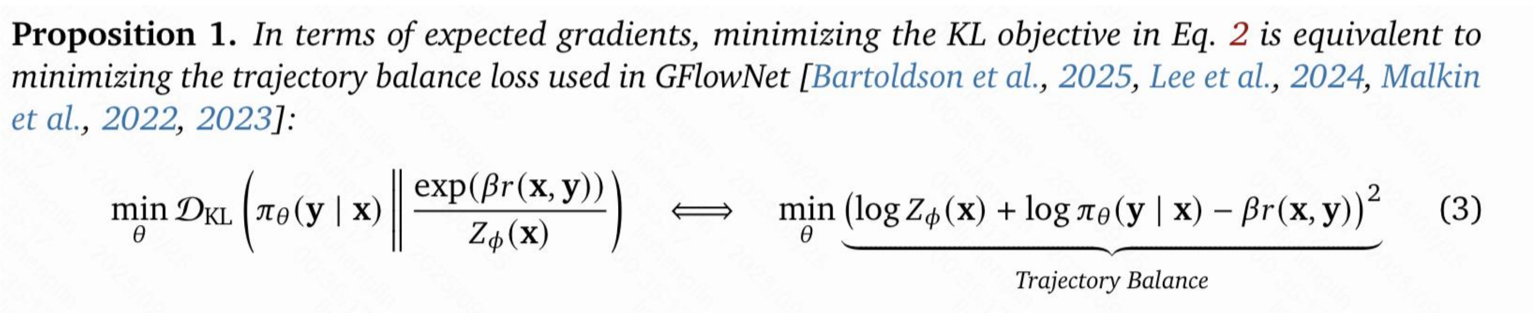
思想：借鉴GFlowNets，目标不是简单地学习数据的分布，而是确保policy model生成的概率与给定的奖励函数成正比。

做法：将目标从 “奖励最大化” 转向 “奖励分布匹配”，通过最小化策略与该目标分布的反向 KL 散度，实现对完整奖励分布的匹配。



$$\min_\theta \mathcal{D}_{\mathrm{KL}}\left(\pi_\theta(\mathbf{y} \mid \mathbf{x}) \,\middle\|\, \frac{\exp(\beta r(\mathbf{x},\mathbf{y}))}{Z_\phi(\mathbf{x})}\right) \quad \Rightarrow \quad \pi_\theta(\mathbf{y} \mid \mathbf{x}) \propto \exp(\beta r(\mathbf{x},\mathbf{y})),$$

优化目标等价于 “让策略生成路径的概率，与目标分布中路径的权重尽可能一致”，既避免策略只聚焦高奖励路径，又确保低奖励但合理的路径被保留。



Proposition 1. *In terms of expected gradients, minimizing the KL objective in Eq. 2 is equivalent to minimizing the trajectory balance loss used in GFlowNet [Bartoldson et al., 2025, Lee et al., 2024, Malkin et al., 2022, 2023]:*

$$\min_\theta \mathcal{D}_{\mathrm{KL}}\left(\pi_\theta(\mathbf{y} \mid \mathbf{x}) \,\middle\|\, \frac{\exp(\beta r(\mathbf{x},\mathbf{y}))}{Z_\phi(\mathbf{x})}\right) \iff \min_\theta \underbrace{\left(\log Z_\phi(\mathbf{x}) + \log \pi_\theta(\mathbf{y} \mid \mathbf{x}) - \beta r(\mathbf{x},\mathbf{y})\right)^2}_{\textit{Trajectory Balance}} \qquad (3)$$

FlowRL: Matching Reward Distributions for LLM Reasoning

3.2 Policy Optimization 熵坍塌

问题一：仅依赖奖励可能导致策略生成 “奖励高但逻辑异常”的路径。

解决一：引入参考策略 π_{ref} （固定的预训练 LLM 生成分布），作为 “先验约束”—— 预训练模型已学习到语言逻辑和基础推理规则，其生成的路径天然具备较高合法性。

$$\exp\left(\beta\, r(\mathbf{x}, \mathbf{y})\right) \cdot \pi_{\text{ref}}(\mathbf{y} \mid \mathbf{x}),$$

问题二：对于长度可变的轨迹（例如长链推理的输出结果），对数概率项随长度增大而增大。

解决二：归一化，平衡长序列与短序列的贡献。

$$\log \pi_\theta(\mathbf{y} \mid \mathbf{x}) = \textstyle\sum_{t=1}^{|\mathbf{y}|} \log \pi_\theta(y_t \mid y_{<t}, \mathbf{x})$$

问题三：训练效率

解决三：离线数据 + 重要性采样。由于优化目标聚焦于轨迹平衡而非期望回报，因此会阻断当前策略的梯度回传，以避免策略过度偏移。

$$\min_\theta \left(\log Z_\phi(\mathbf{x}) + \log \pi_\theta(\mathbf{y} \mid \mathbf{x}) - \beta r(\mathbf{x}, \mathbf{y})\right)^2$$

$$\Downarrow$$

FlowRL

$$\mathcal{L}_{\text{FlowRL}} = w \cdot \left(\log Z_\phi(\mathbf{x}) + \boxed{\frac{1}{|\mathbf{y}|} \log \pi_\theta(\mathbf{y} \mid \mathbf{x})} - \beta \hat{r}(\mathbf{x}, \mathbf{y}) - \boxed{\frac{1}{|\mathbf{y}|} \log \pi_{\text{ref}}(\mathbf{y} \mid \mathbf{x})}\right)^2 \qquad (6)$$

$$\boxed{w = \text{clip}\left(\frac{\pi_\theta(\mathbf{y} \mid \mathbf{x})}{\pi_{\text{old}}(\mathbf{y} \mid \mathbf{x})}, 1 - \epsilon, 1 + \epsilon\right)^{\text{detach}}} \quad \hat{r}_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}.$$

FlowRL: Matching Reward Distributions for LLM Reasoning

Take Away

- 奖励信号的反馈形式（精细化/置信程度）
- 奖励信号与人类偏好的一致性（准确性/泛化性）
  - 奖励建模过程
  - 策略优化过程