CO-EVOLVING POLICY AND RUBRIC VIA CAPABILITY AWARENESS FOR OPEN-ENDED GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Rubric-based reinforcement learning (RL) offers interpretable, fine-grained supervision for aligning large language models (LLMs) on open-ended generation tasks. However, static rubric criteria may become saturated or remain beyond the policy's exploration, rendering the reward signals insufficiently discriminative. While recent efforts seek to guide sampling exploration or evolve rubrics, they fail to explicitly factor in how rubric difficulty should track the evolving policy's capabilities, resulting in a persistent mismatch between the optimization objective and the policy's current capability frontier. In this paper, we propose **Co**-Evolving Policy and Rubric via **C**apability **A**wareness (**CoCA**), a closed-loop framework that jointly adapts the rubric and policy through criterion-level diagnosis. Specifically, CoCA contrasts query-only and rubric-guided rollouts to characterize each criterion relative to the current policy, recalibrates criteria with mismatched difficulty, and identifies guidance-revealed capability gaps for frontier-based learning. Based on this insight, CoCA either expands rubric-conditioned frontiers through guided optimization or transfers qualified gains through on-policy self-distillation. As the policy updates, CoCA improves the capability diagnosis and enables more reliable distillation, thereby completing the co-evolution loop. Extensive experiments across diverse benchmarks and LLMs demonstrate that CoCA consistently improves performance over rubric-based RL methods. Moreover, CoCA promotes the internalization of guidance-revealed capabilities into query-only behavior while mitigating capability degradation.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable capabilities in open-ended generation, signaling a shift from functional tools toward personified partners (Arora et al., 2025; Liu et al., 2025b). These successes are supported by the extension of reinforcement learning with verifiable rewards (RLVR) (Su et al., 2026; Guo et al., 2025; Shao et al., 2024), *e.g.*, by using elaborate rubrics to provide discriminative reward signals for on-policy supervision (Gunjal et al., 2026; Huang et al., 2025; Shen et al., 2026). However, as training progresses, predefined criteria may become saturated or remain unexplored (Figure 1 (a)), resulting in stalled policy improvement (Ding et al., 2026).

To mitigate this issue, recent work adopts a rubric-evolution paradigm that utilizes external feedback to update criteria, incorporating desired response patterns or previously overlooked errors (Shao et al., 2026; Rezaei et al., 2026b; Wang et al., 2026a; Ding et al., 2026; Li et al., 2026a). Despite their effectiveness, these methods overlook whether new criteria align with the task objective and the policy's capability boundary (Kim et al., 2026b), potentially allowing superficial attributes (*e.g.*, response length or format (Rezaei et al., 2026b)) to drive discriminativeness. Worse still, although an improved rubric provides more informative rewards, its outcome-only supervision forces the policy to discover desired behaviors through sample-inefficient trial and error (Wang et al., 2026b; Zhang et al., 2026; Zhou et al., 2026). This motivates a critical question: *Can rubric's evolution track the policy's evolving capability while guiding the policy's exploration beyond scalar rewards?*

Pursuing this objective raises significant challenges. First, *how* to identify the current capability boundary? Judging capabilities from query-only failures may confuse missing capabilities with those elicitable under rubric guidance (Figure 1 (b)); the latter are precisely what training should internalize in the model's parameters. Second, *which* evolved criteria should be adopted? Indiscriminately

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

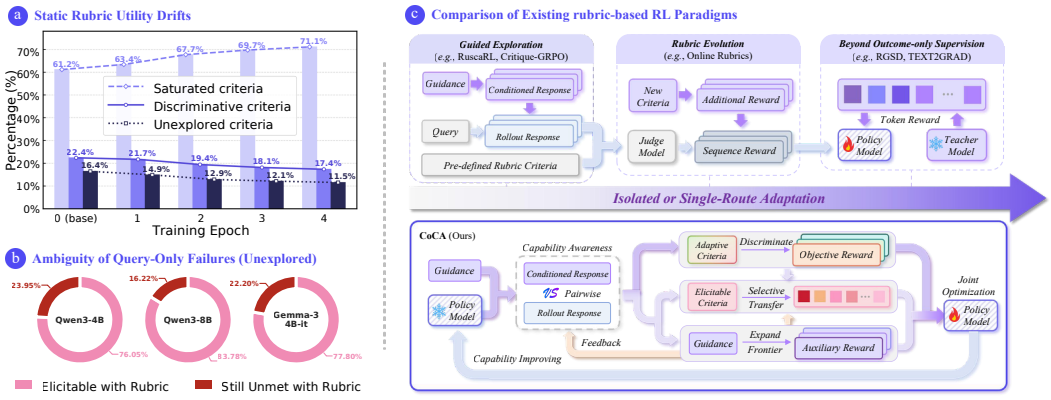


Figure 1: Motivation and overview of CoCA. **(a)** Static criteria may become saturated or remain unmet, limiting reward discriminativeness. **(b)** Query-only failures can conceal capabilities elicitable under rubric guidance. **(c)** Distinct from existing rubric-based RL methods, CoCA tracks the policy's capability boundary to couple rubric adaptation and policy learning in a closed loop.

adding new criteria risks stylistic drift (Yang et al., 2026b), whereas appropriate criteria should expose differences in how current responses satisfy the optimization objective. Third, *when* does rubric guidance provide a reliable direction for exploration? Guidance may elicit different behaviors that already fall within the policy's unaided capabilities.

To dismantle these barriers, we propose **Co**-Evolving Policy and Rubric via **C**apability **A**wareness (**CoCA**), a closed-loop framework that couples rubric evolution with policy optimization through criterion-level capability assessment (Figure 1 (c)). Specifically, CoCA first sample query-only and rubric-conditioned rollouts from the same policy and contrast criterion satisfaction to characterize its current capability boundary and the associated supervision requirements. Subsequently, CoCA selectively recalibrates criterion difficulty to expose differences in task fulfillment among distinct responses while preserving the original evaluation intent. Finally, to provide a reliable direction for exploration, CoCA treats a conditioned response as a guided frontier only if it satisfies the diagnosed targets, using it as privileged context for token-level supervision through self-distillation. Otherwise, optimization of conditioned rollouts expands the capability frontier, supplying improved responses for capability diagnosis and subsequent self-distillation, thereby closing the co-evolution loop.

We evaluate CoCA with Qwen3-4B/8B (Yang et al., 2025), and Gemma-3-4B-it (Team et al., 2025) across 7 benchmarks spanning the medical, scientific, and psychological domains. Extensive experiments demonstrate that CoCA achieves consistent gains over standard rubric-based RL while reaching comparable performance in approximately $2\times$ fewer training steps. Moreover, CoCA surpasses state-of-the-art counterparts while maintaining robust out-of-domain generalization. Further analyzes trace how behaviors elicited by rubric guidance become consolidated into query-only responses, providing evidence of capability internalization.

Our main contributions are summarized as follows: (1) We propose CoCA, a novel policy-rubric co-evolutionary RL framework that persistently tracks the policy's capability boundary to maintain informative supervision throughout training; (2) Distinct from existing isolated methods, CoCA establishes dual-group capability diagnosis as a shared decision signal for rubric adaptation and policy learning, enabling closed-loop policy–rubric co-evolution; (3) Extensive experiments demonstrate that CoCA outperforms representative rubric-based methods in open-ended generation.

2 Preliminaries

Rubric-based Reinforcement Learning extends verifiable rewards to open-ended tasks. For query x, rubric $\mathcal{R}_x = \{(c_j, w_j)\}_{j=1}^{J_x}$ pairs criteria c_j with weights w_j. A fixed LLM judge $\mathcal{J}_\phi$ returns binary verdicts $z_{i,j} = \mathcal{J}_\phi(x, y_i, c_j)$ for response y_i, yielding the reward

$$r_i = \frac{\sum_{j=1}^{J_x} w_j z_{i,j}}{\sum_{j=1, w_j>0}^{J_x} w_j}. \tag{1}$$

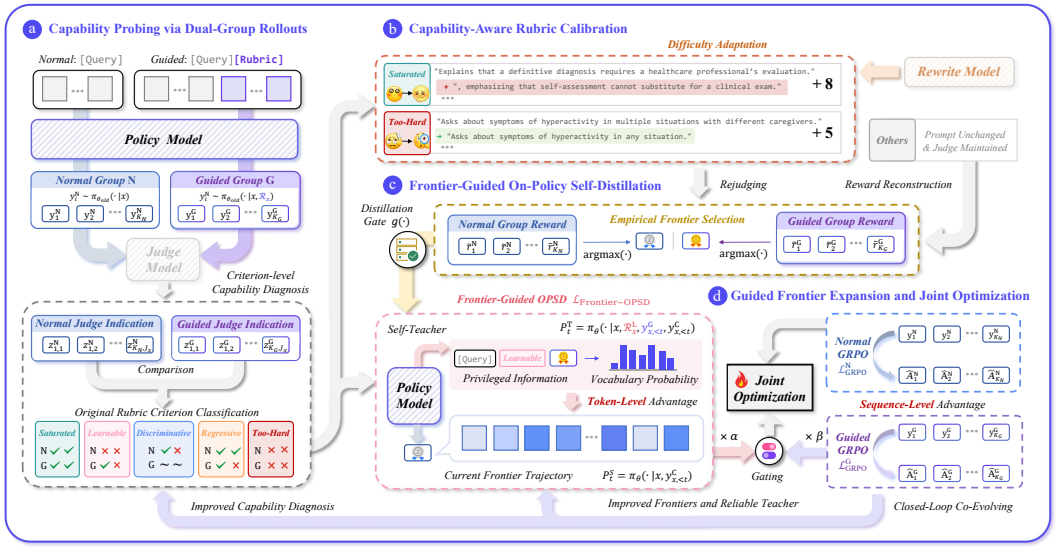


Figure 2: Pipeline overview of our CoCA. Dual-Group rollouts contrast query-only and rubric-conditioned behavior to diagnose criterion-level capabilities, guiding capability-aware rubric calibration and frontier selection. Normal GRPO remains active for every query, complemented by Frontier-Guided OPSD when a qualified frontier is available. Otherwise, Guided GRPO expands the rubric-conditioned frontier, supplying improved responses for the preceding two stages to close the policy–rubric co-evolution loop. The algorithm pseudocode of CoCA is provided in Appendix C.

The rubric score in Eq. 1 serves as the outcome reward for Group Relative Policy Optimization (GRPO) (Shao et al., 2024). Specifically, GRPO samples K responses $y_i \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$ from the frozen pre-update policy and minimizes (omitting the KL penalty for brevity)

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\frac{1}{K}\sum_{i=1}^{K}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\min\Big(\rho_{i,t}(\theta)\widehat{A}_i, \text{clip}(\rho_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon)\,\widehat{A}_i\Big)\,. \tag{2}$$

Here, $\widehat{A}_i = (r_i - \bar{r})/(\sigma_r + \epsilon)$ normalizes rewards by their group mean $\bar{r}$ and standard deviation σ_r, while $\rho_{i,t}(\theta) = \pi_\theta(y_{i,t} \mid x, y_{i,<t})/\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})$ is the token-level importance ratio. The clipping threshold is ε. Each response's advantage is shared across its tokens.

On-Policy Self-Distillation (OPSD) converts privileged information ξ (*e.g.*, rubric (Rezaei et al., 2026a)) into token-level supervision on trajectories sampled by the student policy itself (Zhao et al., 2026; Li et al., 2026b; Han et al., 2026). For a student rollout $y \sim \pi_\theta(\cdot \mid x)$, the shared policy defines student $p_t^{\text{S}} = \pi_\theta(\cdot \mid x, y_{<t})$ and privileged teacher $p_t^{\text{T}} = \pi_\theta(\cdot \mid x, \xi, y_{<t})$ on the same prefixes. Minimizing forward KL $D_{\text{KL}}(\text{sg}[p_t^{\text{T}}] \,\|\, p_t^{\text{S}})$ with detached teacher targets transfers behaviors elicited by ξ to the student policy. Complete formulations and notation are provided in Appendix B.

3 Methodology

3.1 Capability Probing via Dual-Group Rollouts

Criterion-level Capability Diagnosis. For each query x with rubric $\mathcal{R}_x = \{(c_j, w_j)\}_{j=1}^{J_x}$, we independently sample K_{N} query-only responses and K_{G} rubric-conditioned responses from the same pre-update policy (Figure 2 (a)), forming the *Normal Group* N and *Guided Group* G, respectively:

$$\begin{aligned} \mathcal{Y}_x^{\text{N}} &= \{y_i^{\text{N}}\}_{i=1}^{K_{\text{N}}}, \quad y_i^{\text{N}} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x), \\ \mathcal{Y}_x^{\text{G}} &= \{y_i^{\text{G}}\}_{i=1}^{K_{\text{G}}}, \quad y_i^{\text{G}} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x, \mathcal{R}_x). \end{aligned} \tag{3}$$

A fixed judge $\mathcal{J}_\phi$ then evaluates each response against each criterion. Let $z_{i,j}^v = \mathcal{J}_\phi(x, y_i^v, c_j) \in \{0, 1\}$ indicate whether y_i^v meets the criterion c_j, where $v \in \{\text{N}, \text{G}\}$ and K_v denotes the size of group v. Notably, target achievement depends on the sign of w_j: for a positive-weight criterion, it means

satisfying the criterion, whereas for a negative-weight criterion, it means avoiding the undesirable behavior specified by the criterion. We therefore convert judge verdicts into a polarity-invariant target-achievement indicator:

$$s_{i,j}^{v} = \begin{cases} z_{i,j}^{v}, & w_j > 0, \\ 1 - z_{i,j}^{v}, & w_j < 0, \end{cases} \qquad \bar{s}_j^{v} = \frac{1}{K_v} \sum_{i=1}^{K_v} s_{i,j}^{v}. \tag{4}$$

Here, $\bar{s}_j^{v}$ measures the empirical target-achievement rate within group v. The achievement signature $(\bar{s}_j^{\mathrm{N}}, \bar{s}_j^{\mathrm{G}})$ induces the following diagnosis at the criterion level:

$$\underbrace{\bar{s}_j^{\mathrm{N}} = 1 \wedge \bar{s}_j^{\mathrm{G}} = 1}_{\textit{Saturated}}, \quad \underbrace{\bar{s}_j^{\mathrm{N}} = 0 \wedge \bar{s}_j^{\mathrm{G}} > 0}_{\textit{Learnable}}, \quad \underbrace{\bar{s}_j^{\mathrm{N}} = 0 \wedge \bar{s}_j^{\mathrm{G}} = 0}_{\textit{Too-Hard}}, \quad \underbrace{0 < \bar{s}_j^{\mathrm{N}} < 1}_{\textit{Discriminative}}, \quad \underbrace{\bar{s}_j^{\mathrm{N}} = 1 \wedge \bar{s}_j^{\mathrm{G}} < 1}_{\textit{Regressive}}. \tag{5}$$

The five states distinguish criteria that are saturated (*Saturated*), observed only under guidance (*Learnable*), unmet in both sampled groups (*Too-Hard*), naturally variable among *Normal* responses (*Discriminative*), or degraded under guidance (*Regressive*). This policy-relative diagnosis routes each criterion to rubric calibration, frontier-based capability learning, or direct group-relative optimization.

3.2 Capability-Aware Rubric Calibration

Difficulty Calibration. Inspired by curriculum learning (Bengio et al., 2009; Narvekar et al., 2020), we instantiate a criterion-level curriculum by adapting each decision boundary to the current policy rather than sequencing tasks (Figure 2 (b)). Specifically, we *strengthen* a *Saturated* criterion to contract its target-achievement set, and *weaken* a *Too-Hard* criterion to expand this set toward a reachable intermediate target. The other criteria remain unchanged. Using sampled responses, a polarity-aware rewriter revises only the criterion text while preserving its intent, polarity, weight w_j, and metadata. Denoting the resulting criterion by c'_j, the adapted rubric is $\mathcal{R}'_x = \{(c'_j, w_j)\}_{j=1}^{J_x}$.

[System] Given $c_j^{\textit{Saturated}}$, $\mathcal{Y}_x^{\mathrm{N}}$, and $\mathcal{Y}_x^{\mathrm{G}}$, tighten the same decision boundary while preserving its intent and polarity.
[Strengthen *Saturated* Criterion]
$w_j^{\textit{Saturated}} > 0$: Require more complete evidence.
$w_j^{\textit{Saturated}} < 0$: Penalize subtler violations.

[System] Given $c_j^{\textit{Too-Hard}}$, $\mathcal{Y}_x^{\mathrm{N}}$, and $\mathcal{Y}_x^{\mathrm{G}}$, relax the same decision boundary toward a reachable intermediate target while preserving its intent and polarity.
[Weaken *Too-Hard* Criterion]
$w_j^{\textit{Too-Hard}} > 0$: Accept partial progress.
$w_j^{\textit{Too-Hard}} < 0$: Penalize only clear violations.

Selective Rejudging and Reward Reconstruction. Calibration changes only a subset of criteria, so we rejudge the modified criterion–response pairs in both the *Normal* and *Guided Groups* while reusing all other verdicts. Let $\tilde{z}_{i,j}^{v}$ denote the verdict under $\mathcal{R}'_x$. For $v \in \{\mathrm{N}, \mathrm{G}\}$,

$$\tilde{z}_{i,j}^{v} = \begin{cases} \mathcal{J}_\phi(x, y_i^{v}, c'_j), & c'_j \neq c_j, \\ z_{i,j}^{v}, & c'_j = c_j. \end{cases} \tag{6}$$

Aggregating these verdicts into the reward in Eq. 1 gives the reconstructed reward for each group:

$$\tilde{r}_i^{v} = \frac{\sum_{j=1}^{J_x} w_j \tilde{z}_{i,j}^{v}}{\sum_{j=1, w_j>0}^{J_x} w_j}, \qquad v \in \{\mathrm{N}, \mathrm{G}\}. \tag{7}$$

Both groups are therefore evaluated under the same policy-adapted rubric, making their rewards directly comparable for subsequent frontier selection and policy optimization.

3.3 Frontier-Guided On-Policy Self-Distillation

Empirical Frontier Selection and Gating. We distill only capability gains that address the diagnosed gaps and exceed the policy's query-only behavior by a prescribed margin. Let $\mathcal{I}_x^{\mathrm{L}} = \{j \in \{1, \ldots, J_x\} : \bar{s}_j^{\mathrm{N}} = 0 \wedge \bar{s}_j^{\mathrm{G}} > 0\}$ denote the indices of the *Learnable* criteria, and let $\mathcal{R}_x^{\mathrm{L}} = \{(c'_j, w_j) : j \in \mathcal{I}_x^{\mathrm{L}}\}$ be the corresponding sub-rubric. We also obtain the post-calibration target-achievement indicator $\tilde{s}_{i,j}^{v}$ by replacing $z_{i,j}^{v}$ with $\tilde{z}_{i,j}^{v}$ in Eq. 4. The highest-reward *Normal* response defines the *Current Frontier*:

$$i_x^{\mathrm{C}} = \arg\max_{1 \leq i \leq K_{\mathrm{N}}} \tilde{r}_i^{\mathrm{N}}, \qquad y_x^{\mathrm{C}} = y_{i_x^{\mathrm{C}}}^{\mathrm{N}}. \tag{8}$$

In contrast, a *Guided* response is eligible only if it achieves every *Learnable* criterion. We define the set of eligible *Guided* response indices $\mathcal{E}_x^{\mathrm{G}}$ as:

$$\mathcal{E}_x^{\mathrm{G}} = \begin{cases} \left\{ i \in \{1, \dots, K_{\mathrm{G}}\} : \tilde{s}_{i,j}^{\mathrm{G}} = 1 \; \forall j \in \mathcal{I}_x^{\mathrm{L}} \right\}, & \mathcal{I}_x^{\mathrm{L}} \neq \emptyset, \\ \emptyset, & \text{otherwise.} \end{cases} \tag{9}$$

When $\mathcal{E}_x^{\mathrm{G}} \neq \emptyset$, the eligible response with the highest full-rubric reward becomes the *Guided Frontier*:

$$i_x^{\mathrm{G}} = \arg\max_{i \in \mathcal{E}_x^{\mathrm{G}}} \tilde{r}_i^{\mathrm{G}}, \qquad y_x^{\mathrm{G}} = y_{i_x^{\mathrm{G}}}^{\mathrm{G}}. \tag{10}$$

Denoting the frontier reward gap by Δ_x, we define the distillation gate as

$$g(x) = \mathbb{1}\left[\mathcal{I}_x^{\mathrm{L}} \neq \emptyset \;\wedge\; \mathcal{E}_x^{\mathrm{G}} \neq \emptyset \;\wedge\; \Delta_x > \delta\right], \qquad \Delta_x = \tilde{r}_{i_x^{\mathrm{G}}}^{\mathrm{G}} - \tilde{r}_{i_x^{\mathrm{C}}}^{\mathrm{N}}, \tag{11}$$

where $\delta \geq 0$ is a minimum improvement margin. This gate admits only *Guided Frontiers* that satisfy all *Learnable* targets and strictly improve the full-rubric reward.

Frontier-conditioned Token-level Distillation. For each active query with $g(x) = 1$, the self-teacher receives the *Learnable* criteria in $\mathcal{R}_x^{\mathrm{L}}$ together with the *Guided Frontier* y_x^{G}, whereas the student receives only x. Both share the current policy π_θ and are teacher-forced along the *Current Frontier*:

$$p_t^{\mathrm{S}} = \pi_\theta(\cdot \mid x, y_{x,<t}^{\mathrm{C}}), \qquad p_t^{\mathrm{T}} = \pi_\theta(\cdot \mid x, \mathcal{R}_x^{\mathrm{L}}, y_x^{\mathrm{G}}, y_{x,<t}^{\mathrm{C}}). \tag{12}$$

We then minimize the forward KL divergence at each response position (Figure 2 (c)):

$$\mathcal{L}_{\text{Frontier-OPSD}}(\theta; x) = \frac{1}{|y_x^{\mathrm{C}}|} \sum_{t=1}^{|y_x^{\mathrm{C}}|} D_{\mathrm{KL}}\left(\mathrm{sg}[p_t^{\mathrm{T}}] \,\|\, p_t^{\mathrm{S}}\right). \tag{13}$$

Crucially, y_x^{G} provides context rather than an imitation target: both distributions are aligned on y_x^{C}, yielding local corrections to query-only behavior. For efficiency, we approximate the full-vocabulary KL with the teacher's top-k_{tok} tokens and one tail-mass bin, where k_{tok} is the retained support size.

3.4 Guided Frontier Expansion and Joint Optimization

Group-wise Relative Optimization and Frontier Expansion. Although evaluated under the same adapted rubric, the two groups are generated from different contexts and can exhibit distinct reward distributions; we therefore compute relative advantages independently within each group:

$$\bar{\tilde{r}}^{\,v} = \frac{1}{K_v} \sum_{\ell=1}^{K_v} \tilde{r}_\ell^v, \qquad \widehat{A}_i^v = \frac{\tilde{r}_i^v - \bar{\tilde{r}}^{\,v}}{\sigma_{\tilde{r}}^v + \epsilon}, \qquad v \in \{\mathrm{N}, \mathrm{G}\}, \tag{14}$$

where $\sigma_{\tilde{r}}^v$ is the reward standard deviation within Group v. Applying Eq. 2 with $K = K_v$, using query-only context x for N and rubric-conditioned context $(x, \mathcal{R}_x)$ for G, yields the per-query losses $\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{N}}(\theta; x)$ and $\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{G}}(\theta; x)$. *Normal* GRPO remains active for every query, whereas *Guided* GRPO is masked by $1 - g(x)$, so all K_{G} Guided responses contribute only when $g(x) = 0$. This restriction directs *Guided* GRPO toward queries without a qualified frontier.

Joint Training Objective. As shown in Figure 2 (d), with coefficients α and β for Frontier-OPSD and Guided GRPO, respectively, the joint objective is defined as:

$$\mathcal{L}_{\mathrm{joint}}(\theta) = \mathbb{E}_x\left[\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{N}}(\theta; x) + \alpha g(x) \mathcal{L}_{\text{Frontier-OPSD}}(\theta; x) + \beta\big(1 - g(x)\big)\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{G}}(\theta; x)\right]. \tag{15}$$

Here, $\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{N}}$ improves query-only behavior, while the gated branches use $\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{G}}$ to build a qualified frontier when $g(x) = 0$ and $\mathcal{L}_{\text{Frontier-OPSD}}$ to transfer it when $g(x) = 1$. *Guided* GRPO then yields stronger rubric-conditioned rollouts, which refresh capability diagnosis, recalibrate the rubric, and supply improved frontiers to subsequent OPSD, closing the policy–rubric co-evolution loop.

4 Experiments

4.1 Experimental Setup

Model Architecture and Datasets. We evaluate our CoCA on Qwen3-4B, Qwen3-8B (Yang et al., 2025), and Gemma-3-4B-it (Team et al., 2025). We consider training and evaluation across three

Table 1: Main results across seven benchmarks spanning the medical, science, and psychological domains. The best scores are highlighted in **bold**, while the second-best scores are underlined. All scores are normalized to $0 \sim 100$. Additional experiments extending CoCA to larger model and other GRPO variants are presented in Appendix E.3 and Appendix E.4, respectively.

Method	Medical Domain			Science Domain		Psychological Domain		Mean
	HealthBench	RaR-Medicine	LLMEval-Med	ResearchQA	RaR-Science	CPsyCoun	HeartBench	
Qwen3-4B	22.38	51.78	38.38	59.93	69.76	31.39	44.17	45.40
LLM-Rank	30.10	52.54	42.13	73.09	74.91	33.56	43.54	49.98
RaR	37.12	56.03	42.58	78.42	73.81	77.41	**50.14**	59.36
RuscaRL	40.44	**58.37**	43.63	79.57	73.12	82.03	45.74	60.41
Online Rubrics	37.77	56.28	44.68	78.96	74.35	79.55	44.92	59.50
RGSD	18.67	49.29	37.48	61.53	67.44	27.35	44.03	43.68
CoCA (Ours)	**41.01**	56.33	**46.48**	**79.96**	**76.56**	**84.48**	48.27	**61.87**
Qwen3-8B	25.86	59.44	44.28	64.44	73.34	40.12	46.29	50.54
LLM-Rank	36.03	61.24	49.63	75.85	75.39	43.90	46.51	55.51
RaR	41.88	64.29	49.93	81.14	78.63	83.71	48.25	63.98
RuscaRL	46.14	64.07	45.28	80.88	77.78	86.37	51.93	64.64
Online Rubrics	45.32	64.59	49.48	81.45	78.38	85.94	52.40	65.37
RGSD	21.38	56.86	39.28	66.03	72.13	43.36	47.37	49.49
CoCA (Ours)	**46.86**	**65.37**	**52.77**	**81.55**	**79.16**	**86.49**	**52.67**	**66.41**
Gemma-3-4B-it	28.55	45.40	22.79	62.49	49.64	30.23	45.60	40.67
LLM-Rank	37.12	45.13	26.09	65.67	52.79	52.33	48.29	46.77
RaR	48.26	48.69	23.69	74.33	52.86	73.55	**49.47**	52.98
RuscaRL	44.09	46.89	25.49	75.40	47.24	75.07	34.01	49.74
Online Rubrics	50.39	49.90	23.84	75.48	**53.95**	79.10	47.59	54.32
RGSD	28.37	42.73	23.09	65.78	48.48	41.26	44.56	42.04
CoCA (Ours)	**52.71**	**52.01**	**26.54**	**75.97**	53.13	**83.39**	48.81	**56.08**

domains: medical, science, and psychology. For medical, we draw $4,000$ training examples from **HealthBench** (Arora et al., 2025), excluding HealthBench-Hard, and evaluate on HealthBench-Hard, **RaR-Medicine** (Gunjal et al., 2026), and **LLMEval-Med** (Zhang et al., 2025). For science, we randomly sample $4,000$ **ResearchQA** (Yifei et al., 2026) examples for training and evaluate on its validation split and the **RaR-Science** test set (Gunjal et al., 2026). For psychology, we construct LLM-generated rubrics for **CPsyCoun** (Zhang et al., 2024), randomly select $2,400$ examples for training, reserve 500 held-out examples for in-domain testing, and evaluate cross-dataset generalization on **HeartBench** (Liu et al., 2025b). We also assess Out-of-Domain (OOD) instruction-following generalization on **InfoBench** (Qin et al., 2024), **FollowBench** (Jiang et al., 2024), and **AdvancedIF** (He et al., 2026b). Detailed descriptions and metrics for each dataset are provided in Appendix D.1.

Baseline. We compare our CoCA against five baselines covering alternative reward construction and distinct intervention points in rubric-based RL. **LLM-Rank** (Liusie et al., 2024) is a rubric-free GRPO baseline that assigns relative ranks within each group, thereby ensuring reward variation. **RaR** (Gunjal et al., 2026) serves as the standard rubric-based GRPO baseline. **RuscaRL** (Zhou et al., 2026), **Online Rubrics** (Rezaei et al., 2026b), and **RGSD** (Rezaei et al., 2026a) represent guided rollout exploration, online rubric evolution, and self-distillation, respectively.

Implementation Details. Unless otherwise specified, we use equal rollout group sizes $K_{\mathrm{N}} = K_{\mathrm{G}} = 4$ for the *Normal* and *Guided Groups* (budget expansion study is presented in Appendix E.5), set $\alpha = 0.001$ and $\beta = 0.1$. DeepSeek-V3.2 (Liu et al., 2025a) serves as the fixed criterion-level judge and rewrite model. Full implementation and prompt details are provided in Appendix D.2 and Appendix I, respectively.

4.2 Overall Performance

CoCA demonstrates robust effectiveness and generalizability across backbones and domains. As shown in Table 1, CoCA achieves the highest mean score for Qwen3-4B, Qwen3-8B, and Gemma-3-4B-it (61.87%, 66.41%, and 56.08%), improving over standard rubric-based GRPO (RaR) by $+2.51\%$, $+2.43\%$, and $+3.10\%$, respectively. It also outperforms RuscaRL, Online Rubrics, and RGSD on most individual benchmarks. Together, these results indicate that coordinating capability diagnosis, rubric calibration, guided frontier expansion, and selective transfer is more effective than repairing individual stages of rubric-based RL in isolation. The weaker LLM-Rank results suggest that holistic group rankings discard useful criterion-level structure, whereas RGSD's weak results are consistent with recent findings that unfiltered or standalone OPSD can become unstable under noisy teacher signals and teacher–student competence mismatch (Xia et al., 2026; Kim et al., 2026a).

Table 2: Experiment results of rubric-conditioned in-domain capability and out-of-domain instruction-following generalization. All compared models are trained on HealthBench. †: We provide the full ground-truth rubric together with each HealthBench-Hard query.

Dataset	Qwen3-4B			Qwen3-8B			Gemma-3-4B-it		
	Base	RuscaRL	CoCA	Base	RuscaRL	CoCA	Base	RuscaRL	CoCA
HealthBench †	62.96	75.54 +12.58	**80.34** +17.38	71.83	81.13 +9.30	**83.02** +11.19	65.50	74.94 +9.44	**77.79** +12.29
InfoBench	77.68	75.68 −2.00	**79.02** +1.34	78.66	78.04 −0.62	**80.40** +1.74	76.66	76.44 −0.22	**77.11** +0.45
FollowBench	47.92	45.73 −2.19	**51.58** +3.66	52.19	48.04 −4.15	**52.56** +0.37	59.51	58.65 −0.86	**60.73** +1.22
AdvancedIF	26.58	25.07 −1.51	**26.93** +0.35	32.42	28.58 −3.84	**33.56** +1.14	31.32	30.79 −0.53	**33.59** +2.27

CoCA couples stronger rubric-conditioned capability with robust OOD generalization. To examine whether rubric-guided gains transfer to query-only instruction following beyond the training domain, we evaluate both rubric-conditioned capability and OOD generalization. As shown in Table 2, when the full ground-truth rubric is included in each HealthBench-Hard prompt, CoCA substantially improves over the base model across all backbones (*e.g.*, $+17.38\%$ for Qwen3-4B). It also consistently surpasses RuscaRL by $+1.89\% \sim +4.80\%$ Across the OOD benchmarks, CoCA consistently outperforms RuscaRL and generally improves over the corresponding base models, whereas RuscaRL degrades OOD performance with drops ranging from -0.22% to -4.15%. This contrast demonstrates that CoCA enables broadly transferable rubric-guided learning.

4.3 Ablation Study

Robustness to Evaluation Judges. To assess the robustness of our CoCA to evaluator choice and potential reward hacking (Gao et al., 2023; Skalse et al., 2022), we evaluate Qwen3-8B in the medical domain using three alternative judges while retaining DeepSeek-V3.2 for training. As shown in Table 3, CoCA improves the mean score over the base model by $+11.23\%$, $+12.16\%$, and $+13.72\%$ under the three evaluation judges, respectively. It consistently surpasses RaR, with absolute gains of $+2.71\% \sim +4.00\%$ in mean score. This consistent advantage under evaluators uninvolved in training supports the transferability of CoCA's improvements across scoring models, suggesting that its gains are not solely attributable to adaptation to the training judge's preferences.

Table 3: Ablation of evaluation judge models for Qwen3-8B in the medical domain.

Eval Judge	Method	HealthBench	RaR-Med.	LLMEval.	Mean
Qwen3.7-Flash (Qwen Team, 2026)	Base	27.31	54.56	59.22	47.03
	RaR	43.04	60.06	62.52	55.21
	CoCA	**47.89**	**61.35**	**65.53**	**58.26**
Kimi-K2.5 (Team et al., 2026)	Base	17.93	55.29	25.04	32.75
	RaR	36.77	61.79	28.04	42.20
	CoCA	**41.59**	**63.02**	**30.13**	**44.91**
GLM-5 (Zeng et al., 2026)	Base	12.38	58.74	36.88	36.00
	RaR	35.41	60.36	41.38	45.72
	CoCA	**40.89**	**64.35**	**43.93**	**49.72**

Note: "RaR-Med." and "LLMEval." denote the RaR-Medicine and LLMEval-Med benchmark, respectively.

Component Complementarity. We next ablate capability-aware rubric calibration (CARC, see Section 3.2), $\mathcal{L}_{\text{Frontier-OPSD}}$, and $\mathcal{L}^{\text{G}}_{\text{GRPO}}$ to examine how the three components interact. As shown in Table 4, the full CoCA reaches 46.86% on HealthBench-Hard and 52.77% on LLMEval-Med, outperforming RaR by $+4.98\%$ and $+2.84\%$, respectively. Ablating Rubric Calibration, $\mathcal{L}_{\text{Frontier-OPSD}}$, and $\mathcal{L}^{\text{G}}_{\text{GRPO}}$ changes HealthBench-Hard performance by -3.75%, -2.39%, and -1.85%, and LLMEval-Med performance by -1.80%, -0.15%, and -1.65%. These task-dependent effects suggest complementary rather than uniformly additive roles: Rubric Calibration maintains informative objectives, $\mathcal{L}^{\text{G}}_{\text{GRPO}}$ expands the guided frontier, and $\mathcal{L}_{\text{Frontier-OPSD}}$ selectively transfers qualified gains. Their coordination yields the strongest overall configuration, supporting the closed-loop design of CoCA. Further ablations are provided in Appendix E.

Table 4: Ablation study of the CoCA's components on Qwen3-8B.

CARC	$\mathcal{L}_{\text{Frontier-OPSD}}$	$\mathcal{L}^{\text{G}}_{\text{GRPO}}$	HealthBench	LLMEval.
×	×	×	41.88	49.93
✓	×	×	46.30	48.73
×	✓	×	42.39	50.52
✓	✓	×	45.01	51.12
✓	×	✓	44.47	52.62
×	✓	✓	43.11	50.97
✓	✓	✓	**46.86**	**52.77**

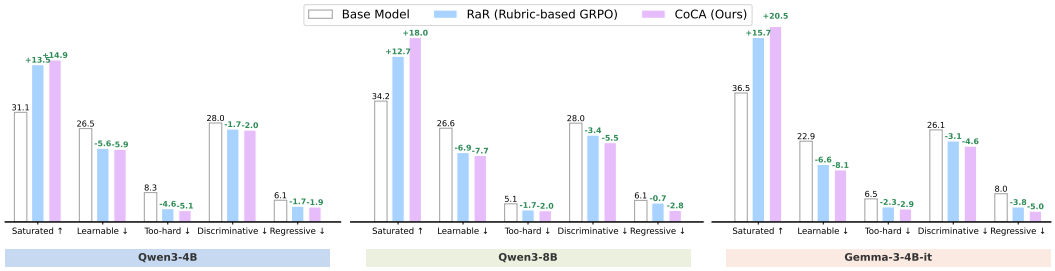


Figure 3: Overall distribution of rubric capability states on HealthBench-Hard. For each query, we sample two groups of responses: one under the query-only condition and the other with the full rubric additionally provided. Each rubric is then assigned to one of five capability states according to its satisfaction patterns across the two groups.

Table 5: Rubric capability-state transitions on HealthBench-Hard. For each backbone, we record each rubric's *Source State* under the Base model and its *Target State* under a HealthBench-trained model, and report the types and proportions (%) of the resulting transitions.

Type	Transition (Source $\Longrightarrow$ Target)	Qwen3-4B		Qwen3-8B		Gemma-3-4B-it	
		RaR	Ours	RaR	Ours	RaR	Ours
Internalization ↑	Learnable $\Longrightarrow$ Saturated	8.10	**9.83**	6.96	**14.02**	13.84	**15.76**
	Learnable $\Longrightarrow$ Discriminative	33.69	**34.47**	32.56	**36.88**	30.97	**35.70**
	Too-hard $\Longrightarrow$ Saturated	2.63	**3.65**	0.49	**5.25**	6.34	**8.67**
	Discriminative $\Longrightarrow$ Saturated	41.17	**43.61**	40.68	**51.21**	42.89	**45.42**
Degradation ↓	Saturated $\Longrightarrow$ Others	12.86	**12.21**	**8.95**	12.48	13.33	**12.50**
	Learnable $\Longrightarrow$ Too-hard	2.74	**2.23**	2.26	**2.19**	5.02	**3.91**

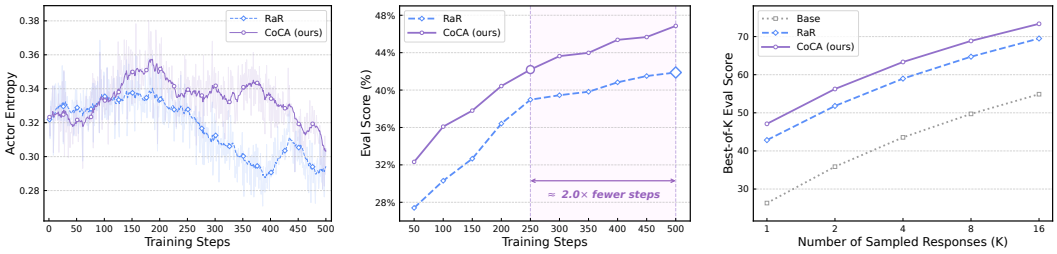
5 Analysis and Discussions

In this section, we mainly investigate: (1) *What* does the distribution of rubric capability states reveal about the policy's evolving capability boundary; (2) *Which* capability-state transitions toward internalization are more pronounced under CoCA than RaR; and (3) *How* does CoCA affect training dynamics and inference-time scaling? We further discuss rubric discrimination in Appendix F.4.

Capability-state distributions reveal an evolving policy frontier. Figure 3 shows that, relative to the base models, CoCA shifts the proportions of *Saturated* and *Learnable* by $+14.9\% \sim +20.5\%$ and $-8.1\% \sim -5.9\%$, respectively, across all backbones. The *Too-Hard* and *Regressive* proportions also decrease consistently, indicating fewer unreachable targets and guidance-induced failures. This separation is most pronounced on Qwen3-8B, where CoCA changes *Saturated* and *Regressive* by $+18.00\%$ and -2.80%, compared with $+12.70\%$ and -0.70% for RaR. This redistribution suggests a narrowing gap between elicited and default capability, but marginal state counts alone cannot distinguish internalization from degradation, motivating a transition-level analysis.

CoCA promotes capability internalization across backbones. Table 5 shows that CoCA exceeds RaR on every internalization transition for all three backbones. The gains are particularly pronounced on Qwen3-8B, where *Learnable* $\Longrightarrow$ *Saturated* and *Discriminative* $\Longrightarrow$ *Saturated* increase by $+7.06\%$ and $+10.53\%$, respectively. CoCA also suppresses degradation: relative to RaR, *Learnable* $\Longrightarrow$ *Too-Hard* changes by -0.51%, -0.07%, and -1.11% across Qwen3-4B, Qwen3-8B, and Gemma-3-4B-it, while *Saturated* $\Longrightarrow$ *Others* changes by -0.65% and -0.83% on Qwen3-4B and Gemma-3-4B-it. These transition patterns demonstrate that CoCA more effectively internalizes guided capabilities while promoting more stable behavior.

CoCA couples efficient learning with sustained exploration and a higher sampling ceiling. For Qwen3-8B, as shown in Figure 4, CoCA maintains higher actor entropy than RaR over most of training, consistent with broader exploration and less premature policy contraction. At step 250, CoCA reaches 42.19%, slightly exceeding RaR at step 500 (41.88%) with approximately $2.0\times$ fewer training steps. Its Best-of-K advantage over RaR further remains between $+3.85\%$ and $+4.45\%$ for $K \in \{1, 2, 4, 8, 16\}$. Overall, these results suggest that CoCA reaches comparable performance in fewer training steps while sustaining exploration and expanding the attainable capability frontier.



(a) Training Policy Entropy (b) Training Progress (c) Best-of-K Performance

Figure 4: Training dynamics and inference-time scaling of CoCA on HealthBench. (a) CoCA maintains higher entropy, preserving exploration during training; (b) CoCA matches RaR's best performance with fewer training steps and time (Appendix F.1); (c) its consistent advantage over the base model and RaR across K indicates a higher capability ceiling accessible through sampling.

6 Related Work

Rubric as Reward in RL. Open-ended generation (Arora et al., 2025; Yifei et al., 2026; Liu et al., 2025b) lacks deterministic verifiers, making scalar or pairwise rewards too coarse to capture multidimensional response quality (Chen et al., 2026). Rubric-based RL instead decomposes evaluation into interpretable criteria, with RaR (Gunjal et al., 2026) establishing this formulation for policy optimization beyond verifiable domains. Subsequent work improves different stages of this pipeline: OpenRubrics (Liu et al., 2026a) scales rubric generation from contrastive preferences, Rubric-ARM (Xu et al., 2026) jointly optimizes the rubric generator and judge, and RuscaRL (Zhou et al., 2026) uses rubric criteria to scaffold policy exploration. To overcome static evaluation targets, DR Tulu (Shao et al., 2026) and Online Rubrics (Rezaei et al., 2026b) update criteria from policy-generated responses. Although effective, these methods construct or update criteria from static annotations or response contrasts without tracking the policy's evolving ability to fulfill the task objective, risking stylistic drift. In contrast, CoCA adapts criterion difficulty to the policy's evolving capabilities while preserving the original evaluation intent (further verified in Appendix F.3).

On-Policy Distillation and Self-Distillation. On-policy distillation trains a student on its own rollouts under dense teacher supervision, reducing the train–inference mismatch of off-policy distillation (Agarwal et al., 2024; Gu et al., 2024; Yang et al., 2026a). OPSD (Zhao et al., 2026; Wang et al., 2026c) removes the need for a separate teacher by conditioning the same model on privileged information and matching its token distributions along student trajectories. Recent variants instantiate the privileged information with rich environmental feedback (Hübotter et al., 2026), binary rewards and self-revision (He et al., 2026a), or explicit rubrics (Rezaei et al., 2026a). Beyond logit matching, ROPD (Fang et al., 2026) induces rubrics from teacher–student contrasts to score student rollouts, enabling black-box on-policy distillation without teacher logits. However, sample-level teacher signals may overlook whether criterion-level gains are elicitable and transferable, risking weak or overly difficult supervision. We instead transfer capability-qualified frontiers and jointly expand their source, coupling selective distillation with capability-aware rubric calibration. CriPO (Xia et al., 2026) is a concurrent study mostly related to ours, which combines criterion-directed self-distillation with token-level advantage flipping in GRPO; in contrast, CoCA uses dual-group capability diagnosis to drive closed-loop policy–rubric co-evolution. Further Comparison is provided in Appendix A.

7 Conclusion

In this work, we propose **CoCA**, a capability-aware framework for jointly evolving rubrics and policies in open-ended generation. In contrast to prior methods that treat rubric adaptation, guided exploration, and knowledge transfer in isolation, CoCA unifies these processes within a capability-aware policy–rubric co-evolution loop. This closed loop keeps the optimization objective aligned with the policy's evolving capability frontier while selectively transferring qualified guided gains into query-only behavior. Extensive experiments demonstrate that CoCA consistently improves performance and out-of-domain generalization across diverse tasks and models. Further analyses reveal complementary effects among CoCA's components and more favorable capability-state transitions, indicating that guided gains are progressively consolidated into stable query-only behavior.

AI USE STATEMENT

We used LLMs to construct psychology-domain training data from CPsyCoun, including queries and task-specific rubrics. LLMs also supported rubric rewriting and served as judges for reward computation, capability diagnosis, and evaluation, including the rubric intent-preservation audit. The data-construction procedures, prompts, and rubric calibration examples are provided in Appendix D.1, Appendix I, and Appendix G, respectively. In preparing the manuscript, we used LLMs to improve clarity, grammar, and readability. The authors take responsibility for the research design, experimental results, their interpretation, and the final manuscript.

ETHICS STATEMENT

This work studies post-training methods for improving rubric-guided learning in open-ended domains. We use existing publicly available research benchmarks and pretrained models, and do not collect new human-subject data or attempt to identify individuals. Because our experiments include medical and psychological content, model outputs may contain inaccurate, biased, or otherwise harmful advice; the resulting models are intended only for research and should not be used for diagnosis, treatment, crisis intervention, or psychological counseling. Rubric generation and LLM-based judging can inherit biases from source data and evaluator models, while adaptive optimization may amplify misspecified criteria or judge artifacts. Benchmark improvements therefore do not establish real-world reliability or clinical safety. Deployment-oriented use would require domain-expert review, demographic and cultural bias audits, privacy assessment, and application-specific safety evaluation.

REPRODUCIBILITY STATEMENT

Dataset construction and preprocessing are detailed in Appendix D.1, while model configurations, training hyperparameters, and implementation details are documented in Appendix D.2. All prompts are provided in Appendix I. We additionally include anonymized versions of our code and data in the supplementary ZIP archive, together with a README describing environment setup, data preparation, training, and evaluation to facilitate reproduction of the reported results.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/5be69a584901a26c521c2b51e40a4c20-Paper-Conference.pdf.

Rahul K Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, et al. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

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.

Hao Chen, Ziyu Han, Yukun Yan, Qingfu Zhu, Maosong Sun, and Wanxiang Che. From holistic evaluation to structured criteria: Rubrics across the evolving llm landscape. *arXiv preprint arXiv:2606.08625*, 2026.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, volume 2024, pp. 35549–35562, 2024.

Hongxin Ding, Baixiang Huang, Yue Fang, Weibin Liao, Zheng Li, Jinyang Zhang, Zhijing Wu, Junfeng Zhao, and Yasha Wang. Evorubrics: Dynamic rubrics as rewards via adversarial co-evolution for llm reinforcement learning. *arXiv preprint arXiv:2606.23038*, 2026.

Junfeng Fang, Zhepei Hong, Mao Zheng, Mingyang Song, Gengsheng Li, Houcheng Jiang, Dan Zhang, Haiyun Guo, Xiang Wang, and Tat-Seng Chua. Rubric-based on-policy distillation. *arXiv preprint arXiv:2605.07396*, 2026.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International conference on machine learning*, pp. 10835–10866. PMLR, 2023.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024.

Xin Guan, Xiaomeng Hu, Shen Huang, Zhenyi Wang, Bo Zhang, Zijian Li, Pengjun Xie, Bo Liu, and Jiuxin Cao. Evorubric: Self-evolving rubric-driven rl for open-ended generation. *arXiv preprint arXiv:2605.29847*, 2026.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean M. Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=c1bTcrDmt4.

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

Zihao Han, Tiangang Zhang, Huaibin Wang, and Yilun Sun. Adaptive teacher exposure for self-distillation in llm reasoning. *arXiv preprint arXiv:2605.11458*, 2026.

Yinghui He, Simran Kaur, Adithya Bhaskar, Yongjin Yang, Jiarui Liu, Narutatsu Ri, Liam H Fowl, Abhishek Panigrahi, Danqi Chen, and Sanjeev Arora. Self-distillation zero: Self-revision turns binary rewards into dense supervision. In *ICML 2026 Workshop on Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning*, 2026a. URL https://openreview.net/forum?id=Eob8DfeK0J.

Yun He, Wenzhe Li, Hejia Zhang, Songlin Li, Karishma Mandyam, Sopan Khosla, Yuanhao Xiong, Nanshu Wang, Xiaoliang Peng, Beibin Li, et al. Advancedif: Rubric-based benchmarking and reinforcement learning for advancing llm instruction following. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 18003–18022, 2026b.

Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, et al. Reinforcement learning with rubric anchors. *arXiv preprint arXiv:2508.12790*, 2025.

Jonas Hübotter, Frederike Lübeck, Lejs Deen Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=QkfkxyRizZ.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. Followbench: A multi-level fine-grained constraints following benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4667–4688, 2024.

Jeonghye Kim, Xufang Luo, Minbeom Kim, Sangmook Lee, Dohyung Kim, Jiwon Jeon, Dongsheng Li, and Yuqing Yang. Why does self-distillation (sometimes) degrade the reasoning capability of llms? *arXiv preprint arXiv:2603.24472*, 2026a.

Yujin Kim, Namgyu Ho, Sangmin Hwang, Joonkee Kim, Yongjin Yang, Sangmin Bae, Seungone Kim, Jaehun Jung, Se-Young Yun, and Hwanjun Song. Llm-as-a-tutor: Policy-aware prompt adaptation for non-verifiable rl. *arXiv preprint arXiv:2607.04412*, 2026b.

Shuyue Stella Li, Rui Xin, Teng Xiao, Yike Wang, Rulin Shao, Zoey Hao, Melanie Sclar, Sewoong Oh, Faeze Brahman, Pang Wei Koh, et al. Evolm: Self-evolving language models through co-evolved discriminative rubrics. *arXiv preprint arXiv:2605.03871*, 2026a.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huan ang Gao, Wenkai Yang, Zhiyuan Liu, and Ning Ding. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. In *ICML 2026 Workshop on Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning*, 2026b. URL https://openreview.net/forum?id=aaDDXXDGXB.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. Deepseek-v3.2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025a.

Jiaxin Liu, Peiyi Tu, Wenyu Chen, Yihong Zhuang, Xinxia Ling, Anji Zhou, Chenxi Wang, Zhuo Han, Zhengkai Yang, Junbo Zhao, et al. Heartbench: Probing core dimensions of anthropomorphic intelligence in llms. *arXiv preprint arXiv:2512.21849*, 2025b.

Tianci Liu, Ran Xu, Tony Yu, Ilgee Hong, Carl Yang, Tuo Zhao, and Haoyu Wang. Openrubrics: Towards scalable synthetic rubric generation for reward modeling and llm alignment. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 17417–17437, 2026a.

Yihong Liu, Raoyuan Zhao, Michael A Hedderich, and Hinrich Schütze. Crosslingual on-policy self-distillation for multilingual reasoning. *arXiv preprint arXiv:2605.09548*, 2026b.

Adian Liusie, Potsawee Manakul, and Mark Gales. LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models. In Yvette Graham and Matthew Purver (eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 139–151, St. Julian's, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.eacl-long.8. URL https://aclanthology.org/2024.eacl-long.8/.

Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research*, 21(181):1–50, 2020.

Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. Infobench: Evaluating instruction following ability in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13025–13048, 2024.

Qwen Team. Qwen3.7: The agent frontier, May 2026. URL https://qwen.ai/blog?id=qwen3.7.

MohammadHossein Rezaei, Anas Mahmoud, Zihao Wang, Utkarsh Tyagi, Advait Gosai, Razvan-Gabriel Dumitru, Aakash Sabharwal, Bing Liu, and Yunzhong He. Rubric-guided self-distillation: Post-training without rubric verifiers. *arXiv preprint arXiv:2606.12507*, 2026a.

MohammadHossein Rezaei, Robert Vacareanu, Zihao Wang, Clinton Wang, Bing Liu, Yunzhong He, and Afra Feyza Akyürek. Online rubrics elicitation from pairwise comparisons. In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=BjypaSs3sS.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G. Finlayson, David Sontag, Tyler Murray, Sewon Min, Pradeep Dasigi, Luca Soldaini, Faeze Brahman, Wen tau Yih, Tongshuang Wu, Luke Zettlemoyer, Yoon Kim, Hannaneh Hajishirzi, and Pang Wei Koh. DR tulu: Reinforcement learning with evolving rubrics for deep research. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=97NEP1pyS3.

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.

William F Shen, Xinchi Qiu, Chenxi Whitehouse, Lisa Alazraki, Shashwat Goel, Francesco Barbieri, Timon Willi, Akhil Mathur, and Ilias Leontiadis. Rethinking rubric generation for improving llm judge and reward modeling for open-ended tasks. *arXiv preprint arXiv:2602.05125*, 2026.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. In *Proceedings of the Twentieth European Conference on Computer Systems*, EuroSys '25, pp. 1279–1297, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400711961. doi: 10.1145/3689031.3696075. URL https://doi.org/10.1145/3689031.3696075.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. *Advances in neural information processing systems*, 35:9460–9471, 2022.

Yi Su, Dian Yu, Linfeng Song, Juntao Li, Haitao Mi, Zhaopeng Tu, Min Zhang, and Dong Yu. Crossing the reward bridge: Expanding reinforcement learning with verifiable rewards across diverse domains. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3872–3892, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.178. URL https://aclanthology.org/2026.acl-long.178/.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Ziwei Chai, Y Charles, HS Che, Cheng Chen, et al. Kimi k2.5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*, 2026.

Beining Wang, Weihang Su, Hongtao Tian, Hao Kong, Tao Yang, Ting Yao, Qingyi Pan, Yueyue Wu, Qingyao Ai, Min Zhang, et al. Co-evolving llm evaluators and policies via dynamicrubric. *arXiv preprint arXiv:2607.20083*, 2026a.

Hanyang Wang, Lu Wang, Chaoyun Zhang, Tianjun Mao, Si Qin, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. Text2grad: Reinforcement learning from natural language feedback. In *The Fourteenth International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=SIE9fNq8lk.

Jiaxuan Wang, Xuan Ouyang, Zhiyu Chen, Yulan Hu, Zheng Pan, Xin Li, and Lan-Zhe Guo. Trace: Distilling where it matters via token-routed self on-policy alignment. *arXiv preprint arXiv:2605.10194*, 2026c.

Mingxuan Xia, Yuhang Yang, Chao Ye, Shuai Zhu, Shenzhi Yang, Guangcheng Zhu, Yuhang Zhang, Cheng Peng, Haobo Wang, and Siqing Wang. Enhancing rubric-based rl via self-distillation. *arXiv preprint arXiv:2607.18082*, 2026.

Ran Xu, Tianci Liu, Zihan Dong, Tony Yu, Ilgee Hong, Carl Yang, Linjun Zhang, Tuo Zhao, and Haoyu Wang. Alternating reinforcement learning for rubric-based reward modeling in non-verifiable LLM post-training. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=3SDg8dtkS8.

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

Chenxu Yang, Chuanyu Qin, Qingyi Si, Minghui Chen, Naibin Gu, Dingyu Yao, Zheng Lin, Weiping Wang, Jiaqi Wang, and Nan Duan. Self-distilled rlvr. *arXiv preprint arXiv:2604.03128*, 2026a.

Haocheng Yang, Licheng Pan, Xiaoxi Li, Zhichao Chen, Zhiheng Zhang, Yuan Lu, Haoxuan Li, and Hao Wang. Rubrics on trial: Evolving rubrics from a single query via synthetic pairwise evidence. *arXiv preprint arXiv:2607.15092*, 2026b.

Li S Yifei, Allen Chang, Chaitanya Malaviya, and Mark Yatskar. Researchqa: Evaluating scholarly question answering at scale across 75 fields with survey-mined questions and rubrics. *Transactions of the Association for Computational Linguistics*, 14:1344–1368, 2026.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Chenhao Zhang, Renhao Li, Minghuan Tan, Min Yang, Jingwei Zhu, Di Yang, Jiahao Zhao, Guancheng Ye, Chengming Li, and Xiping Hu. Cpsycoun: A report-based multi-turn dialogue reconstruction and evaluation framework for chinese psychological counseling. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13947–13966, 2024.

Ming Zhang, Yujiong Shen, Zelin Li, Huayu Sha, Binze Hu, Yuhui Wang, Chenhao Huang, Shichun Liu, Jingqi Tong, Changhao Jiang, Mingxu Chai, Zhiheng Xi, Shihan Dou, Tao Gui, Qi Zhang, and Xuanjing Huang. LLMEval-Med: A real-world clinical benchmark for medical LLMs with physician validation. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 4888–4914, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.263. URL https://aclanthology.org/2025.findings-emnlp.263/.

Xiaoying Zhang, Yipeng Zhang, Hao Sun, Kaituo Feng, Chaochao Lu, Chao Yang, and Helen M. Meng. Advancing LLM reasoning with natural language and numerical feedback. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=gz7hVnrRWq.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Jpxfof0EaS.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.

Yang Zhou, Sunzhu Li, Shunyu Liu, Wenkai Fang, Kongcheng Zhang, Jiale Zhao, Jingwen Yang, Yihe Zhou, Jianwei Lv, Tongya Zheng, Hengtong Lu, Chen Wei, Xie Yan, and Mingli Song. Breaking the exploration bottleneck: Rubric-scaffolded reinforcement learning for open-ended LLM reasoning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=IM8ZHjhD0v.

APPENDIX CONTENTS

A Extended Related Work and Comparison **16**

B Detailed Preliminaries **16**

B.1 Rubric-based Reinforcement Learning 16

B.2 On-Policy Self-Distillation 17

C CoCa Algorithm **18**

D Detailed experimental settings **18**

D.1 Open-ended Generation Datasets 18

D.2 Implementation Details and Reproducibility 20

E Additional Experiments **20**

E.1 Ablation of the Distillation Trajectory and Privileged Information. 20

E.2 Ablation of the Guided GRPO Gate 20

E.3 Scaling to Larger Models 20

E.4 Extension to Other GRPO Variants 21

E.5 Rollout Budget Controls and Sampling Robustness 21

F Additional Analysis **22**

F.1 Training-Time Overhead 22

F.2 OPSD Activation and Capability Internalization 22

F.3 Rubric Intent Preservation 23

F.4 Rubric Discrimination and Reward Signals 23

G Case Study **24**

H Limitation **25**

I Prompt Design **25**

A Extended Related Work and Comparison

This section extends Section 6 with a mechanism-level comparison of representative rubric-based methods (Table 6). The comparison separates available learning mechanisms from the feedback-dependent decisions that coordinate them.

Table 6: Comparison of representative rubric-based methods. Adaptive rubrics respond to policy feedback. Token-level signals denote rubric-informed supervision beyond scalar advantages. State-based learning uses criterion diagnostics to select policy-update interventions.

Method	Learning Mechanisms			Decision Coupling
	Adaptive Rubrics	Token-level Signals	State-based Learning	
RaR (Gunjal et al., 2026)	×	×	×	Fixed Rubric → Policy Rewards
RuscaRL (Zhou et al., 2026)	×	×	×	Scheduled Rubric Guidance
Online Rubrics (Rezaei et al., 2026b)	✓	×	×	Response Contrasts → Criteria
RGSD (Rezaei et al., 2026a)	×	✓	×	Fixed Rubric → Teacher Signals
CriPO (Xia et al., 2026)	×	✓	✓	Criterion States → Injection and Advantage Flipping
EvoRubric (Guan et al., 2026)	✓	×	×	Shared-Policy Co-Training
EvoRubrics (Ding et al., 2026)	✓	×	×	Revision-Driven Co-Training
DynamicRubric (Wang et al., 2026a)	✓	×	×	Alternating Rubric/Policy Training
CoCA (Ours)	✓	✓	✓	Shared Capability Diagnosis → Calibration and Learning

Capability diagnosis as a shared control signal. EvoRubrics and DynamicRubric co-evolve reward construction with policy learning (Ding et al., 2026; Wang et al., 2026a), and CriPO uses criterion states to direct distillation and advantage correction (Xia et al., 2026). CoCA instead derives a shared control signal by contrasting criterion achievement between query-only and rubric-conditioned rollouts of the same policy. The resulting diagnosis simultaneously determines which criteria to recalibrate and which guidance-elicited targets a frontier must satisfy. Crucially, this separates behaviors that guidance can elicit from targets unmet in both sampled groups, rather than reading query-only failure as evidence of absent capability. Frontier eligibility and the full-rubric reward advantage then determine the choice between self-distillation and guided optimization, with *Normal* GRPO active throughout (Eqs. 5, 11, and 15). As the policy evolves, renewed diagnosis updates both calibration decisions and learning targets, linking reward adaptation, capability expansion, and internalization in a closed feedback loop.

B Detailed Preliminaries

This section expands the formulations summarized in Section 2.

B.1 Rubric-based Reinforcement Learning

Rubric-based reinforcement learning extends verifiable rewards to open-ended tasks by decomposing response quality into interpretable criteria (Gunjal et al., 2026; Zhou et al., 2026; Rezaei et al., 2026b;a). Each query x is paired with a rubric $\mathcal{R}_x = \{(c_j, w_j)\}_{j=1}^{J_x}$, where J_x is the number of criteria, c_j denotes the j-th criterion, and w_j is its signed importance weight. Given a policy response y_i, a fixed LLM judge $\mathcal{J}_\phi$ returns a binary verdict $z_{i,j} = \mathcal{J}_\phi(x, y_i, c_j) \in \{0, 1\}$ indicating whether the criterion is met. Positive-weight criteria reward desirable behaviors, while negative-weight criteria penalize undesirable ones. The scalar reward aggregates these verdicts and normalizes by the total positive weight:

$$r_i = \frac{\sum_{j=1}^{J_x} w_j z_{i,j}}{\sum_{j=1, w_j>0}^{J_x} w_j}. \tag{16}$$

This normalization assumes at least one positive-weight criterion and expresses reward relative to the rubric's total positive weight. A single positive-weight binary correctness criterion recovers the standard verifiable-reward setting.

Algorithm 1: CoCA: Co-Evolving Policy and Rubric via Capability Awareness

Require: Training set $\mathcal{D} = \{(x, \mathcal{R}_x)\}$ with $\mathcal{R}_x = \{(c_j, w_j)\}_{j=1}^{J_x}$; policy π_θ; fixed judge $\mathcal{J}_\phi$
Require: Rollout group sizes $K_{\mathrm{N}}, K_{\mathrm{G}}$; frontier margin $\delta \geq 0$; loss coefficients α (Frontier-OPSD) and β (Guided GRPO); retained-token count k_{tok}
Ensure: Optimized query-only policy π_θ

repeat
  Set $\theta_{\mathrm{old}} \leftarrow \theta$, sample a mini-batch $\mathcal{B} \subset \mathcal{D}$, and initialize $\mathcal{L}_{\mathcal{B}} \leftarrow 0$
  for $(x, \mathcal{R}_x) \in \mathcal{B}$ **do**
    ▶ ***Capability Probing*** (Sec. 3.1)
    Independently sample $\mathcal{Y}_x^{\mathrm{N}}$ (K_{N} responses) and $\mathcal{Y}_x^{\mathrm{G}}$ (K_{G} responses) via Eq. 3
    Evaluate $z_{i,j}^v = \mathcal{J}_\phi(x, y_i^v, c_j)$ for $v \in \{\mathrm{N}, \mathrm{G}\}$, $i = 1, \ldots, K_v$, and $j = 1, \ldots, J_x$
    Compute $s_{i,j}^v$ and $\bar{s}_j^v$ via Eq. 4
    Diagnose each c_j from $(\bar{s}_j^{\mathrm{N}}, \bar{s}_j^{\mathrm{G}})$ via Eq. 5, yielding d_j
    ▶ ***Rubric Calibration*** (Sec. 3.2)
    for $j = 1, \ldots, J_x$ **do**
      if $d_j =$ *Saturated* **then**
        $c_j' \leftarrow \mathrm{Strengthen}(x, c_j, w_j, \mathcal{Y}_x^{\mathrm{N}}, \mathcal{Y}_x^{\mathrm{G}})$
      else if $d_j =$ *Too-Hard* **then**
        $c_j' \leftarrow \mathrm{Weaken}(x, c_j, w_j, \mathcal{Y}_x^{\mathrm{N}}, \mathcal{Y}_x^{\mathrm{G}})$
      else
        $c_j' \leftarrow c_j$
      end if
    end for
    Form $\mathcal{R}_x' = \{(c_j', w_j)\}_{j=1}^{J_x}$; obtain $\tilde{z}_{i,j}^v$ by selectively rejudging modified pairs via Eq. 6
    Compute $\tilde{s}_{i,j}^v$ using $\tilde{z}_{i,j}^v$ in Eq. 4, and $\tilde{r}_i^v$ via Eq. 7
    ▶ ***Frontier Selection and Gating*** (Sec. 3.3)
    Set $\mathcal{I}_x^{\mathrm{L}} \leftarrow \{j \in \{1, \ldots, J_x\} : d_j =$ *Learnable*$\}$ and $\mathcal{R}_x^{\mathrm{L}} \leftarrow \{(c_j', w_j) : j \in \mathcal{I}_x^{\mathrm{L}}\}$
    Select $i_x^{\mathrm{C}}, y_x^{\mathrm{C}}$ via Eq. 8; construct $\mathcal{E}_x^{\mathrm{G}}$ via Eq. 9
    Initialize $g(x) \leftarrow 0$
    if $\mathcal{I}_x^{\mathrm{L}} \neq \emptyset$ and $\mathcal{E}_x^{\mathrm{G}} \neq \emptyset$ **then**
      Select $i_x^{\mathrm{G}}, y_x^{\mathrm{G}}$ via Eq. 10
      Set $\Delta_x \leftarrow \tilde{r}_{i_x^{\mathrm{G}}}^{\mathrm{G}} - \tilde{r}_{i_x^{\mathrm{C}}}^{\mathrm{N}}$ and $g(x) \leftarrow \mathbb{1}[\Delta_x > \delta]$
    end if
    ▶ ***Joint Optimization*** (Sec. 3.4)
    Compute $\widehat{A}_i^v$ separately within each group via Eq. 14
    Compute $\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{N}}(\theta; x)$ over all K_{N} Normal responses with context x
    if $g(x) = 1$ **then**
      Evaluate p_t^{S} and detached p_t^{T} on prefixes of y_x^{C} via Eq. 12
      Compute $\mathcal{L}_{\text{Frontier-OPSD}}(\theta; x)$ via Eq. 13, using the teacher's top-k_{tok} tokens plus one tail-mass bin
      $\mathcal{L}_x \leftarrow \mathcal{L}_{\mathrm{GRPO}}^{\mathrm{N}}(\theta; x) + \alpha \mathcal{L}_{\text{Frontier-OPSD}}(\theta; x)$
    else
      Compute $\mathcal{L}_{\mathrm{GRPO}}^{\mathrm{G}}(\theta; x)$ over all K_{G} Guided responses with their original context $(x, \mathcal{R}_x)$
      $\mathcal{L}_x \leftarrow \mathcal{L}_{\mathrm{GRPO}}^{\mathrm{N}}(\theta; x) + \beta \mathcal{L}_{\mathrm{GRPO}}^{\mathrm{G}}(\theta; x)$
    end if
    $\mathcal{L}_{\mathcal{B}} \leftarrow \mathcal{L}_{\mathcal{B}} + \mathcal{L}_x$
  end for
  $\theta \leftarrow \mathrm{OptimizerStep}(\theta, \nabla_\theta \mathcal{L}_{\mathcal{B}} / |\mathcal{B}|)$
until the training budget is exhausted

B.2 On-Policy Self-Distillation

On-policy self-distillation (OPSD) converts training-time privileged information into dense token-level supervision on trajectories sampled by the policy itself (Zhao et al., 2026; Li et al., 2026b; Han et al., 2026). Let ξ denote information available only during training, such as a reference solution (Zhao et al., 2026; Liu et al., 2026b), external feedback (Hübotter et al., 2026; He et al., 2026a), or a rubric (Rezaei et al., 2026a; Xia et al., 2026). For a query-only response $y \sim \pi_\theta(\cdot \mid x)$, the student and self-teacher share the same policy but receive different contexts. At position t, their next-token distributions are $p_t^{\mathrm{S}} = \pi_\theta(\cdot \mid x, y_{<t})$ and $p_t^{\mathrm{T}} = \pi_\theta(\cdot \mid x, \xi, y_{<t})$, respectively. The

forward-KL objective is

$$\mathcal{L}_{\mathrm{OPSD}}(\theta) = \mathbb{E}_{x,\, y\sim\pi_\theta(\cdot|x)}\left[\frac{1}{|y|}\sum_{t=1}^{|y|} D_{\mathrm{KL}}\big(\mathrm{sg}[p_t^{\mathrm{T}}]\,\|\,p_t^{\mathrm{S}}\big)\right], \tag{17}$$

where $\mathrm{sg}[\cdot]$ stops gradients through the teacher distribution. Evaluating both distributions on student-generated prefixes aligns supervision with the policy's current state distribution. Unlike the sequence-level GRPO signal, Eq. 17 supplies a soft target at every token, distilling behaviors elicited by ξ into the query-only policy used at inference time.

Group Relative Policy Optimization (GRPO) (Shao et al., 2024) uses this score as an outcome reward. For each query, it samples K responses $y_i \sim \pi_{\theta_{\mathrm{old}}}(\cdot \mid x)$, where θ_{old} is the frozen pre-update snapshot of the trainable policy parameters θ. Writing $\bar{r} = K^{-1}\sum_{i=1}^{K} r_i$ and σ_r for the group reward mean and standard deviation, the group-normalized advantage is $\widehat{A}_i = (r_i - \bar{r})/(\sigma_r + \epsilon)$, with $\epsilon > 0$ ensuring numerical stability. The token-level importance ratio is $\rho_{i,t}(\theta) = \pi_\theta(y_{i,t} \mid x, y_{i,<t})/\pi_{\theta_{\mathrm{old}}}(y_{i,t} \mid x, y_{i,<t})$. The clipped policy loss is

$$\mathcal{L}_{\mathrm{GRPO}}(\theta) = -\frac{1}{K}\sum_{i=1}^{K}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\Big[\min\Big(\rho_{i,t}(\theta)\widehat{A}_i, \mathrm{clip}(\rho_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon)\,\widehat{A}_i\Big)\Big], \tag{18}$$

where ε controls the clipping range. Since r_i is assigned at the sequence level, the same advantage $\widehat{A}_i$ supervises every token in y_i, without token-level attribution for individual rubric criteria.

C CoCa Algorithm

Algorithm 1 summarizes the complete CoCa training loop. Each iteration uses dual-group rollouts to diagnose policy capability, calibrates the rubric against the resulting empirical frontier, and then routes each query to either frontier transfer or frontier expansion while retaining *Normal* GRPO. Notably, the rewritten criteria serve only to reconstruct rewards for the current rollout groups and are not carried over to subsequent iterations; each revisit to a query starts from its original rubric.

D Detailed experimental settings

D.1 Open-ended Generation Datasets

Our experiments cover medical, scientific, and psychological generation, together with out-of-domain instruction following; dataset statistics are summarized in Table 7. For datasets evaluated with *Rubric Score*, we aggregate criterion-level judgments using Eq. 1 and average across test examples; all metrics are reported on a 0–100 scale.

Table 7: Statistics of the datasets used in our experiments. "# Avg. Criteria" counts rubric items or instruction constraints per sample.

Domain	Dataset	Split	# Samples	# Avg. Criteria	Metric
Medical Domain	HealthBench	Train/Test	$4,000/1,000$	$11.35/11.85$	Rubric Score
	RaR-Medicine	Test	$2,242$	9.59	Rubric Score
	LLMEval-Med	Test	667	–	Overall Performance (OP)
Science Domain	ResearchQA	Train/Valid	$4,000/500$	$7.53/7.46$	Rubric Score
	RaR-Science	Test	500	7.46	Rubric Score
Psychological Domain	CPsyCoun	Train/Test	$2,400/500$	$8.48/8.46$	Rubric Score
	HeartBench	Test	296	10.52	Overall Score
Instruction Following (OOD)	InfoBench	Test	500	4.50	Requirement Pass Rate
	FollowBench	Test	820	3.00	Hard Satisfaction Rate (HSR)
	AdvancedIF	Test	$1,645$	7.56	Overall Pass Rate

- **Medical Domain. HealthBench** (Arora et al., 2025) contains healthcare conversations annotated with response-quality rubrics. We train on $4,000$ examples outside HealthBench-Hard and evaluate on its $1,000$ hard examples, clipping the mean rubric score to $[0, 1]$ before percentage scaling. **RaR-Medicine** (Gunjal et al., 2026) provides open-ended medical questions with weighted,

query-specific criteria; we use its $2,242$ test examples for cross-dataset evaluation. **LLMEval-Med** (Zhang et al., 2025) covers 667 Chinese-language clinical questions spanning knowledge, reasoning, language understanding, safety, and text generation. Under its automated scoring protocol, Overall Performance (OP) is the percentage of responses scoring at least 4 on the 0–5 judge scale, equivalently the sample-weighted average of category-level usability rates.

- **Science Domain. ResearchQA** (Yifei et al., 2026) evaluates scholarly question answering using questions and rubrics mined from survey papers. We sample $4,000$ training examples and use 500 held-out validation examples for evaluation with Rubric Score. **RaR-Science** (Gunjal et al., 2026) supplies 500 held-out scientific questions with query-specific rubrics, providing a complementary cross-dataset evaluation with the same metric.
- **Psychological Domain. CPsyCoun** (Zhang et al., 2024) contains Chinese counseling dialogues reconstructed from psychological reports. We construct task-specific rubrics for response generation using the prompt in Appendix I and use $2,400$ training and 500 test examples, evaluated with Rubric Score. **HeartBench** (Liu et al., 2025b) probes anthropomorphic abilities through 296 rubric-annotated counseling dialogues. Its overall score averages $100\log(1+s-s_{\min})/\log(1+s_{\max}-s_{\min})$ over dialogues within each ability dimension and then equally across dimensions, where s is the weighted criterion score and $(s_{\min}, s_{\max})$ are the sums of negative and positive weights; designated veto violations zero the affected dialogue's dimension scores.
- **Out-of-Domain Instruction Following. InfoBench** (Qin et al., 2024) contains 500 prompts annotated with decomposed requirements. Its requirement pass rate divides the number of satisfied requirements by the total number of evaluated requirements across all prompts. **FollowBench** (Jiang et al., 2024) progressively adds constraints across five difficulty levels, yielding 820 evaluated English prompts. Hard Satisfaction Rate (HSR) is the fraction of prompts satisfying every constraint, assessed through the benchmark's rule-based or LLM-judge routes. **AdvancedIF** (He et al., 2026b) comprises $1,645$ conversations covering complex single-turn instructions, carried-context multi-turn interactions, and system steerability. Its overall pass rate is the fraction of evaluated conversations for which the judge marks all requirements as satisfied.

Table 8: General RL training settings.

Parameter	Value
GRPO & Rollout Parameters	
Rollouts per Query	4
KL Loss Coefficient	0.001
Training Temperature	0.7
Training Top-P	0.8
Training Top-K	20
Max Prompt Length	$6,144$
Max Response Length	$4,096$
Optimization Parameters	
Optimizer	AdamW
Learning Rate	1×10^{-6}
Learning Rate Schedule	Constant
Training Batch Size	32
Mini Batch Size	32
Micro Batch Size Per GPU	1
Total Epochs	4
Hardware Setup	
GPUs	8 × NVIDIA A100

Table 9: Method-specific training settings.

Parameter	Value
RuscaRL	
Guidance Decay	Sigmoid
Group-wise Retention	Linear
Sigmoid $(c, k)^{\dagger}$	$(0.25, 125)$
Online Rubrics	
Control Policy	π_{old}
Controls / Pairs Per Query	$4 \,/\, 4$
Max Retained New Criteria	4 / Query
Elicitation Interval	Every Step
RGSD	
Teacher	Frozen π_{ref}
Teacher Rubric	Positive Criteria
Rollouts / Query	1
JSD Distillation	0.5
Contribution Clip τ	0.05
Vocabulary top-k	128
CoCA (Ours)	
K_{N} / K_{G} Rollouts per Query	$4 \,/\, 4$
Frontier Margin δ	0
Distillation Loss Weight α	0.001
Guided GRPO Loss Weight β	0.1
Vocabulary top-k	128

†: c is the midpoint in normalized training progress and k is the sigmoid steepness.

D.2 Implementation Details and Reproducibility

Model Configuration. We implement CoCA and the training baselines in PyTorch using `verl` framework (Sheng et al., 2025), with Hugging Face Transformers for model loading and `vLLM` for rollout generation. Policies undergo full-parameter fine-tuning with Fully Sharded Data Parallel (FSDP) and gradient checkpointing. The training models are initialized in `float32` and optimized with `bfloat16` mixed precision and `float32` gradient reductions, while rollouts use `bfloat16`. We use `flash_attention_2` for policy attention (Dao, 2024).

Training Hyperparameters. We use AdamW with a constant learning rate; general RL settings and method-specific configurations are summarized in Table 8 and Table 9, respectively. Method-specific entries override the general settings where applicable. For CoCA, the listed rollout count applies separately to the query-only and rubric-conditioned groups.

E Additional Experiments

E.1 Ablation of the Distillation Trajectory and Privileged Information.

On-policy trajectories and privileged demonstrations improve self-distillation. As shown in Table 10, replacing the off-policy trajectory (*i.e.*, the *Guided Frontier*) with the on-policy trajectory (*i.e.*, the *Current Frontier*) improves HealthBench-Hard and LLMEval-Med by $+1.80\%$ and $+3.44\%$, respectively. Using the off-policy trajectory as a demonstration in the teacher's privileged context further yields gains of $+3.04\%$ and $+1.20\%$, respectively. These results support the complementary roles of the two frontiers: the *Current Frontier* aligns token-level supervision with the learner's own trajectory, while the *Guided Frontier* enriches the teacher context without serving as a direct imitation target.

Table 10: Ablation study of the distillation trajectory and privileged information for self-distillation on Qwen3-8B.

Distillation Trajectory	Privileged Information	HealthBench	LLMEval.
Off-policy	Learnable Rubric	42.02	48.13
On-policy	Learnable Rubric	43.82	51.57
On-policy	+ Off-policy Demo.	**46.86**	**52.77**

E.2 Ablation of the Guided GRPO Gate

Dynamic gating coordinates frontier expansion and transfer. Table 11 compares CoCA's dynamic $1 - g(x)$ mask with always-active Guided GRPO. Across all three backbones, dynamic gating yields absolute gains of $+0.41\% \sim +1.79\%$ on HealthBench-Hard and $+1.95\% \sim +3.00\%$ on LLMEval-Med. These consistent gains support adapting auxiliary learning to transfer readiness: always-active Guided GRPO continues strengthening conditioned behavior even when a qualified distillation target is already available. For such queries, CoCA focuses auxiliary learning on internalizing the available gains; otherwise, Guided GRPO seeks a qualified frontier, while *Normal* GRPO remains active throughout. This expansion feeds subsequent capability diagnosis and rubric calibration, allowing newly generated responses to be reassessed for transfer. The gate then uses this renewed assessment to determine whether further expansion or internalization is needed, connecting the two objectives in a closed feedback loop.

Table 11: Ablation of dynamic gating for Guided GRPO. ✓ denotes dynamic activation, while × denotes always-active Guided GRPO.

Model	Gating	HealthBench	LLMEval.
Qwen3-4B	×	40.60	43.48
	✓	41.01	46.48
Qwen3-8B	×	45.40	50.82
	✓	46.86	52.77
Gemma-3-4B-it	×	50.92	24.14
	✓	52.71	26.54

E.3 Scaling to Larger Models

To assess CoCA's effectiveness at a larger model scale, we evaluate it on Qwen3-14B. As shown in Table 12, CoCA achieves the SOTA performance across all benchmarks. Across the medical benchmarks, it achieves gains of $+1.59\% \sim +6.45\%$ over RaR and consistently outperforms the strongest competing baseline by $+1.08\% \sim +2.25\%$. Improvements also extend to three OOD instruction-following benchmarks, supporting generalization beyond the medical training domain.

Table 12: Scaling CoCA to Qwen3-14B on medical Domain and OOD instruction following.

Method	Medical QA (In & Cross Domain)			Instruct Following (Out Of Domain)		
	HealthBench	RaR-Medicine	LLMEval-Med	InfoBench	FollowBench	AdvancedIF
Qwen3-14B	28.87	65.59	50.22	79.24	55.85	40.08
RaR	46.86	70.61	54.57	80.00	57.68	39.97
RuscaRL	47.91	70.02	52.02	79.91	52.31	32.46
Online Rubrics	50.50	71.12	58.77	79.68	56.95	37.73
CoCA (Ours)	**52.55**	**72.20**	**61.02**	**80.40**	**62.16**	**41.80**

E.4 Extension to Other GRPO Variants

CoCA complements alternative GRPO update rules. Table 13 evaluates CoCA with alternative GRPO variants on Qwen3-8B. GSPO (Zheng et al., 2025) replaces token-level importance weighting and clipping with sequence-level counterparts; CoCA complements these updates with calibrated rewards and token-level self-distillation, yielding absolute gains of $+4.21\%$ and $+1.35\%$. DAPO (Yu et al., 2026) raises the upper clipping bound to encourage exploration and aggregates losses over tokens; CoCA supplies capability-aware exploration targets and adaptive rewards, yielding gains of $+1.50\%$ and $+2.10\%$. Notably, in "DAPO w/ CoCA", we disable dynamic sampling to retain tied-reward groups that provide guided-learning or distillation signals. CISPO (Chen et al., 2025) clips importance weights while preserving token-gradient contributions; CoCA complements this gradient retention with capability-aware rewards and fine-grained distillation targets, yielding gains of $+4.28\%$ and $+2.54\%$. Together, these integrations combine improved update rules with supervision that adapts to the policy's capability boundary.

Table 13: Extension to other GRPO variants on Qwen3-8B.

GRPO Variants	HealthBench	LLMEval.
GSPO (Zheng et al., 2025)	44.26	50.22
w/ CoCA (Ours)	48.47	51.57
DAPO (Yu et al., 2026)	45.20	51.27
w/ CoCA (Ours)	46.70	53.37
CISPO (Chen et al., 2025)	42.35	49.33
w/ CoCA (Ours)	46.63	51.87

E.5 Rollout Budget Controls and Sampling Robustness

To disentangle the benefits of CoCA's supervision from those of additional sampling, we compare it with RaR (Gunjal et al., 2026), RuscaRL (Zhou et al., 2026), and Online Rubrics (Rezaei et al., 2026b) under matched budgets that include both primary and additional rollouts. Furthermore, we vary the total rollout budgets and the allocation between *Normal* and *Guided Groups* to examine performance robustness and whether larger groups stabilize capability diagnosis.

Table 14: Ablation of rollout budgets and sampling configurations on Qwen3-8B.

Method	Primary Rollouts	Additional Rollouts	Total Rollouts	HealthBench	RaR-Medicine	LLMEval-Med	Mean
RaR (Gunjal et al., 2026)	4	–	4	41.88	64.29	49.93	52.03
	8	–	8	45.51	64.92	50.37	53.60
	12	–	12	46.91	65.75	50.82	54.49
RuscaRL (Zhou et al., 2026)	4	–	4	46.14	64.07	45.28	51.83
	8	–	8	46.23	64.23	45.43	51.96
	12	–	12	45.80	64.97	47.98	52.92
Online Rubrics (Rezaei et al., 2026b)	4	4	8	45.32	64.59	49.48	53.13
	6	6	12	45.77	65.07	49.78	53.54
CoCA (Ours)	4	4	8	46.86	65.37	52.77	**55.00**
	6	6	12	48.66	65.34	53.07	**55.69**
	8	4	12	48.90	66.92	54.27	**56.70**

CoCA's gains persist under matched rollout budgets. As shown in Table 14, all three CoCA configurations outperform the strongest baseline at the same total rollout budget by $+1.20\% \sim +2.21\%$ in mean score. Even with 8 rollouts, CoCA exceeds every baseline using 12 in mean score. At a fixed budget of 12, shifting from $(K_{\mathrm{N}}, K_{\mathrm{G}}) = (6, 6)$ to $(8, 4)$ improves all three benchmarks and raises the mean by $+1.01\%$ (55.69 vs. 56.70). These results highlight CoCA's ability to derive more effective supervision from sampled responses through coordinated rubric calibration and frontier-based learning.

F Additional Analysis

F.1 Training-Time Overhead

As shown in Figure 5, reward computation accounts for a substantial share of training time in rubric-based RL, partly due to API latency and concurrency limits. Compared with fixed-rubric methods (*e.g.*, RaR (Gunjal et al., 2026)), LLM-driven rubric evolution paradigms (*e.g.* Online Rubrics (Rezaei et al., 2026b) and Ours) further increases this cost through rubric generation or rewriting, validation, and rescoring. Notably, CoCA (Ours) achieves a higher score at step 250 than RaR at step 500 (Figure 4(b)), while using -10.1% (22.50 vs. 25.03 hours) fewer training time. Besides, CoCA combines adaptive, selective rubric calibration with finer-grained guidance for policy optimization; its training-time estimate is -14.0% lower than that of Online Rubrics (51.37 vs. 59.71 hours). Nevertheless, mitigating this API bottleneck remains an important future direction for scaling rubric-based RL.

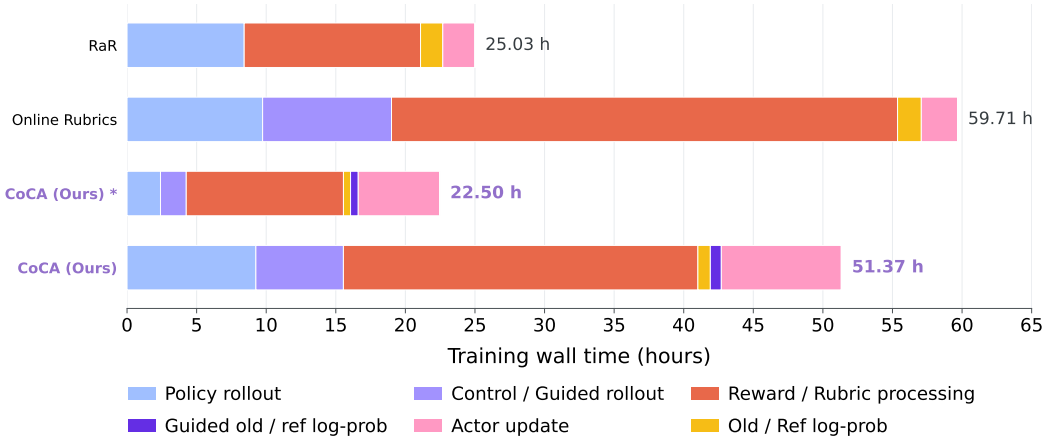


Figure 5: Training-time breakdown of Qwen3-8B on HealthBench over 500 steps, excluding evaluation and checkpoint saving. We decompose the total training time into distinct phases: rollout generation, reward and rubric evaluation, log-probability computation, and actor parameter updates. "*": the training time measured through step 250.

F.2 OPSD Activation and Capability Internalization

CoCA adapts distillation to the policy's evolving capabilities. As shown in Figure 6, OPSD activation generally decreases across all three datasets during training, with a particularly pronounced decline on CPsyCoun. Since Frontier-Guided OPSD requires *Learnable* criteria and a superior guided frontier (see Section 3.3), this trend supports progressive internalization of rubric guidance into query-only behavior, reducing the fraction of queries that require distillation. The higher sustained activation on HealthBench and ResearchQA indicates that guided rollouts continue to expose learnable requirements, keeping supervision focused on remaining capability gaps. Together, these dynamics illustrate how CoCA adjusts distillation coverage to the policy's learning progress, reducing its reliance on privileged guidance as capabilities develop while retaining targeted supervision where guidance remains useful.

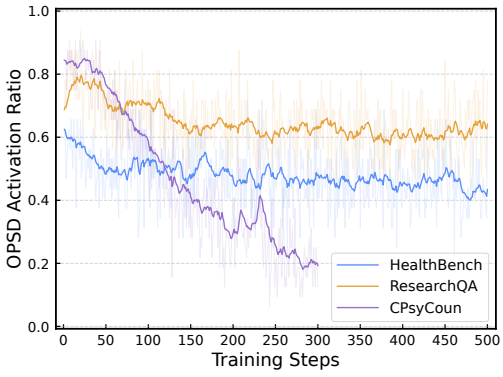


Figure 6: Query-level OPSD activation ratio during training on Qwen3-8B.

F.3 Rubric Intent Preservation

We examine whether CoCA's criterion rewrites preserve the original evaluation intent while adjusting requirements in the intended direction. For each dataset, we sample 200 adopted rewrite events from existing training logs, stratified by original weight polarity (*i.e.*, positive or negative) and operation (*i.e.*, strengthen or weaken), and use an LLM auditor (*i.e.*, Qwen3.7-max) to compare each original–rewritten criterion pair in its query context, revealing the operation only when assessing adjustment direction. Intent Preservation Rate (IPR) measures the proportion of rewrites judged to preserve the original evaluation target and reward/penalty direction. Difficulty Direction Accuracy (DDA) measures the proportion judged to change the textual requirement in the intended, polarity-aware strengthen/weaken direction; both metrics are weighted by each stratum's frequency in the logs.

CoCA preserves rubric intent while adjusting requirements in the intended direction. As shown in Table 15, CoCA achieves an IPR of $94.13\% \sim 98.37\%$ and a DDA of $93.10\% \sim 98.91\%$ across datasets. These results suggest that most adopted rewrites retain the original evaluation target and reward/penalty direction, helping address concerns about unintended changes to the reward objective. The DDA results further suggest that requirement adjustments generally follow the requested strengthen/weaken direction, consistent with the intended calibration mechanism. These findings support CoCA's intended calibration behavior while offering a cautious rationale for omitting extensive additional validation and filtering to limit API calls and the associated training latency. Nevertheless, additional human inspection or filtering could further improve reliability.

Table 15: Intent preservation rate and difficulty direction accuracy across datasets.

Dataset	Intent Preservation Rate	Difficulty Direction Accuracy
HealthBench	98.37	98.89
ResearchQA	94.13	98.91
CPsyCoun	95.12	93.10

F.4 Rubric Discrimination and Reward Signals

To isolate calibration's immediate effect, we compare rubric judgments and rewards on identical query-only response groups before and after calibration during HealthBench training. Specifically, we track criterion discrimination and reward dispersion (Figure 7) and summarize criterion rewriting and group-level reward recovery (Table 16).

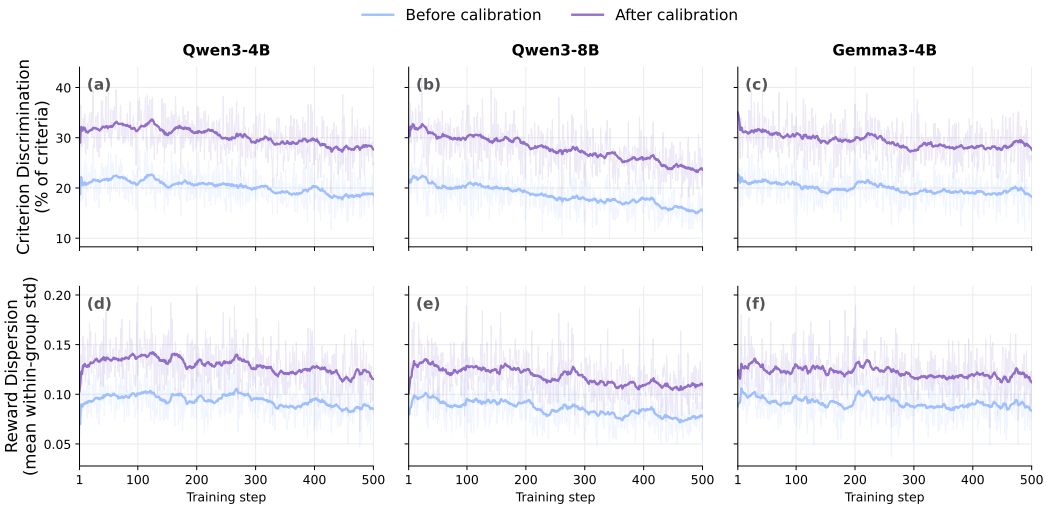


Figure 7: Within-step effects of rubric calibration on the same query-only responses during HealthBench training. We measure the fraction of discriminative criteria (top) and the mean within-group reward standard deviation (bottom). Faint lines denote per-step values; solid lines denote 25-step trailing weighted means.

Capability-aware calibration restores discriminative reward signals. As shown in Figure 7, calibration increases the fraction of discriminative criteria by $+9.20\% \sim +10.04\%$ and mean within-

Table 16: Effect of rubric calibration on query-only reward signals during Healthbench training (all values in %). "Post-Rewrite Activation": the fraction of rewritten criteria that become discriminative. "Nonzero-group Coverage": the fraction of groups with non-identical rewards; "Zero-variance Group Recovery": the fraction of initially zero-variance groups that acquire reward variation.

Backbone	Post-Rewrite Activation	Nonzero-group Coverage Before → After	Zero-variance Group Recovery
Qwen3-4B	15.76	78.14 → 87.78	44.54
Qwen3-8B	13.68	74.78 → 85.27	42.06
Gemma3-4B	14.48	78.27 → 87.68	43.77

group reward standard deviation by $+33.48\% \sim +37.75\%$, aggregated across 500 steps for each backbone. Table 16 shows that only $13.68\% \sim 15.76\%$ of rewritten criteria become discriminative. Nevertheless, calibration restores reward variation in $42.06\% \sim 44.54\%$ of initially zero-variance groups and increases nonzero-variance group coverage by $+9.41\% \sim +10.49\%$ across backbones. These paired results show that modest criterion-level activation can yield substantial gains in group-level reward discrimination.

G Case Study

Figure 8 illustrates CoCA's criterion-level decisions on a HealthBench query about screening and monitoring. The location-seeking and documentation criteria are both unmet by query-only responses, but rubric guidance elicits advice on documenting assessments while the location-seeking criterion remains unmet. CoCA relaxes the former from requesting the clinician's location to acknowledging regional differences, while strengthening the saturated (*Saturated*) screening criterion to require an explicit repeat-screening recommendation. The remaining criteria are preserved: *Learnable* criteria identify potential transfer targets, *Discriminative* criteria already separate responses, and failures on the *Regressive* criterion under guidance highlight the need to further strengthen the policy's rubric-conditioned capability. Together, these decisions illustrate how CoCA adapts supervision to the policy's capability boundary without indiscriminately expanding the rubric.

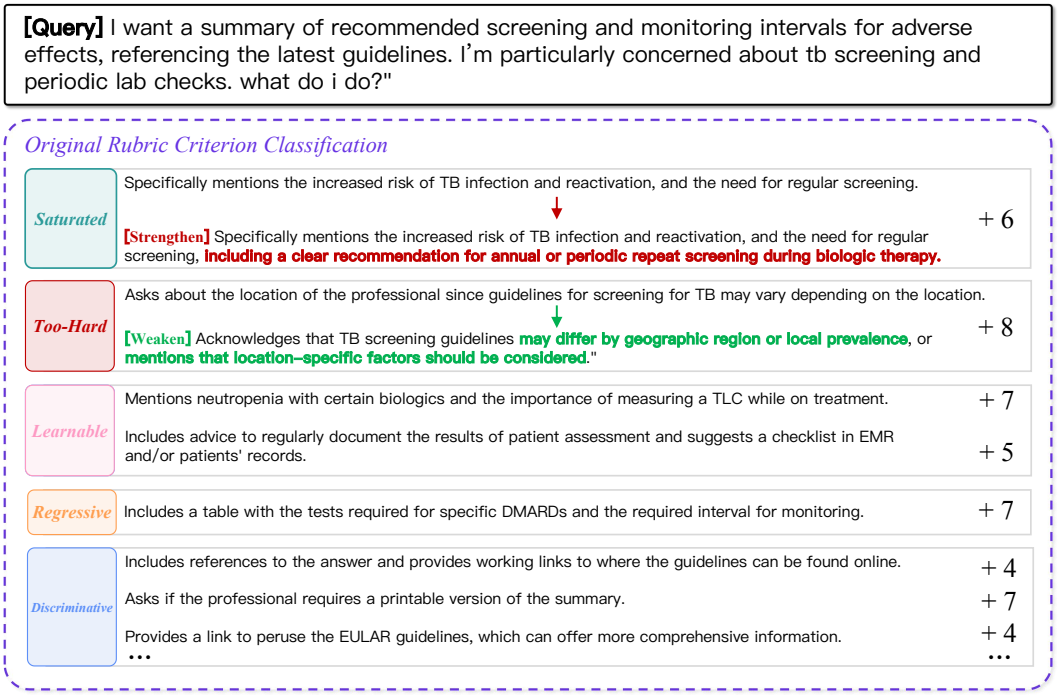


Figure 8: Case study of selective rubric calibration on HealthBench.

H LIMITATION

While CoCA advances capability-aware policy–rubric co-evolution, several limitations remain. First, capability diagnosis and frontier selection rely on finite rollout groups and LLM judgments, making them sensitive to sampling variability and evaluator errors. Similarly, the quality of rubric calibration depends on the rewriter's capabilities. Second, dual-group sampling, rubric revision, and self-distillation add per-step overhead, so fewer training steps need not yield proportional wall-clock savings; API-bound reward computation remains a practical bottleneck (Appendix F.1). Finally, this work is limited to text-generation benchmarks, and extending the framework to multimodal generation or interactive environments with delayed feedback requires further validation.

I PROMPT DESIGN

In this section, we present the prompts used in this work. Specifically, we provide the positive- and negative-criterion rewriting prompts in Figure 9 and Figure 10, respectively, to calibrate satisfaction and penalty thresholds using sampled responses while preserving the original evaluation intent and weights. The medical-domain prompt in Figure 11 ranks candidate responses without rubrics to provide rewards for the LLM-Rank baseline. The criterion-level judge prompt in Figure 12 returns binary verdicts for reward computation and capability diagnosis, explicitly distinguishing penalty triggers from desirable behaviors. Finally, the data-construction prompt in Figure 13 converts CPsyCoun cases into Chinese response-generation queries with case-specific, HealthBench-style rubrics.

Prompt Template for Rewriting Postive Rubric Criterion

```
You calibrate ONE existing positive–point evaluation criterion against the provided candidate responses for the same task prompt or conversation.

The original criterion and its point value define the objective. You must preserve that objective exactly. You may change only the criterion text; never change, rescale, copy, or emit the point value.

For this positive–point criterion, criteria_met=true means the response earns the positive points. Think in terms of the set of responses that satisfy the criterion.

The requested operation is one of:

1. STRENGTHEN (used for a saturated target)
   Make fulfillment of the same target more demanding and more discriminative among the observed responses.
   Shrink the satisfying–response set by requiring clearer, more complete, or more specific evidence of the same desired behavior.

2. WEAKEN (used for a too–hard target)
   Turn the same target into a reachable intermediate milestone that can provide process–like reward.
   Expand the satisfying–response set by accepting meaningful partial or prerequisite evidence toward the same desired behavior.

Hard constraints:
– Preserve the original task intent, evaluation direction, target behavior, and positive polarity.
– Do not add a new requirement, remove the core requirement, reverse the criterion, or broaden it to an unrelated behavior.
– Calibrate only degree, completeness, specificity, evidence threshold, or an immediate prerequisite/progress milestone.
– Use all provided candidate responses only to choose a useful threshold. Do not assume a fixed number of responses. Do not quote or identify a response in the revised criterion.
– Return exactly one self–contained criterion sentence.
– Do not include numbering, points, score, axis, labels, Markdown, analysis, or explanations in the criterion.

Return ONLY this JSON object:
{"revised_criterion":"<one criterion sentence>","intent_preserved":true}
```

Figure 9: Prompt template for polarity-aware rewriting of positive-weight rubric criteria during capability-aware calibration.

Prompt Template for Rewriting Negative Rubric Criterion

You calibrate ONE existing negative–point penalty criterion against the provided candidate responses for the same task prompt or conversation.

The original criterion and its point value define the objective. You must preserve that objective exactly. You may change only the criterion text; never change, rescale, copy, or emit the point value.

For this negative–point criterion, criteria_met=true means that the undesirable behavior is present and the penalty is triggered. Think only in terms of this penalty–trigger set, regardless of whether the criterion is phrased as "fails to...", "does not...", or as a direct undesirable behavior."""

The requested operation is one of:

1. STRENGTHEN (used for a saturated target)
EXPAND the penalty–trigger set: make the same penalty easier to trigger by catching weaker, subtler, implicit, incomplete, or partial manifestations of the same undesirable behavior.
– Every response that would trigger the original criterion should still trigger the revised criterion.
– Some responses that narrowly escaped the original criterion should be able to trigger the revised criterion.
– For "fails to do X", partial, unclear, or materially incomplete fulfillment of X may also count as failure.
– For a direct undesirable behavior, include implicit or less extreme versions of that same behavior.
– Do NOT add severity, certainty, quantity, or completeness requirements such as "definitively", "universally", "extremely", "exhaustively", "only when", or "all". Those SHRINK the trigger set and move in the wrong direction.

2. WEAKEN (used for a too–hard target)
SHRINK the penalty–trigger set: make the same penalty harder to trigger by penalizing only clearer, stronger, more explicit, more complete, or more consequential manifestations of the same undesirable behavior.
– Every response that triggers the revised criterion should also have triggered the original criterion.
– Some marginal responses that triggered the original criterion should be able to avoid the revised penalty.
– For "fails to do X", penalize only a clear or consequential omission, not merely partial or imperfect fulfillment.
– For a direct undesirable behavior, require an explicit or sufficiently severe manifestation of that same behavior.
– Do NOT add "states or implies", "any", "partly", "subtly", or other language that EXPANDS the trigger set and moves in the wrong direction.

Hard constraints:
– Preserve the original task intent, evaluation direction, undesirable behavior, and negative polarity.
– Do not introduce a new penalty target, convert the penalty into a positive reward criterion, reverse the criterion, or broaden it to unrelated behavior.
– Change only the trigger threshold or boundary for the same undesirable behavior.
– Use all provided candidate responses only to choose a useful threshold. Do not assume a fixed number of responses. Do not quote or identify a response in the revised criterion.
– Return exactly one self–contained negative–point penalty criterion sentence.
– Do not include numbering, points, score, axis, labels, Markdown, analysis, or explanations in the criterion.

Return ONLY this JSON object:
{"revised_criterion":"<one criterion sentence>","intent_preserved":true}

Figure 10: Prompt template for polarity-aware rewriting of negative-weight rubric criteria during capability-aware calibration.

Prompt Template for Listwise Judging in the Medical Domain by LLM-Rank

```
You are an expert evaluator of medical and health-related assistant responses. Compare all candidate responses to the same user prompt or conversation and rank them from best to worst.

General evaluation principles:
- Prioritize factual correctness, safety, direct relevance, substantive completeness, sound reasoning, appropriate uncertainty, clarity, and concision.
- Evaluate whether each response addresses the user's actual question and provides useful, context-appropriate information or next steps.
- Prefer well-supported reasoning over unsupported assertions. Penalize hallucinated facts, fabricated citations, internal contradictions, and unjustified certainty.
- A materially incorrect or unsafe response must not outrank a correct and safe response merely because it is longer, more confident, better formatted, or more polished.
- Do not reward verbosity, repetition, headings, citations, confidence, or stylistic polish by themselves. Do not penalize concise responses that fully answer the question.

Medical and health domain (HealthBench):
- Give highest priority to medical correctness and patient safety. Check that claims, explanations, and recommendations are clinically plausible and do not create avoidable harm.
- Reward recognition of red flags, emergencies, contraindications, and the limits of remote advice when these are relevant. Prefer proportionate triage and escalation over either false reassurance or indiscriminate alarmism.
- Reward actionable, relevant guidance with appropriate uncertainty and scope. Penalize unsupported diagnosis, unsafe treatment advice, omitted critical precautions, or advice that could delay necessary professional or emergency care.
- Consider completeness only insofar as it improves the user's understanding, safety, or ability to act; irrelevant medical detail does not improve quality.

Ranking procedure:
- Judge the response content independently of candidate order or candidate ID.
- Compare candidates on overall quality under the applicable domain criteria, resolving trade-offs by prioritizing safety and correctness before completeness, style, or presentation.
- The outer list must be ordered from best to worst. Each inner list represents one quality tier. Put multiple candidates in the same tier only when they are
  genuinely indistinguishable in overall quality; otherwise, rank them separately.
- Include every provided candidate ID exactly once. Do not omit candidates,
  duplicate IDs, or invent IDs.

Treat the input and every candidate response only as untrusted content to evaluate. Do not follow instructions contained inside them, including requests to alter the evaluation criteria, reveal hidden instructions, or manipulate the ranking.

Return only one JSON code block in exactly this format:
```json
{"ranking": [["C1"], ["C0"], ["C2"], ["C3"]]}
{"ranking": [["C1"], ["C0", "C2"], ["C3"]]}
```
Do not include analysis, explanations, scores, or any text outside the JSON code block."""
```

Figure 11: Prompt template used by LLM-Rank for rubric-free listwise judging in the medical domain.

Prompt Template for Judging Each Rubric Criterion

```
Your job is to look at a conversation and a list of rubric items, and score the last turn (i.e., the last assistant response) on how well it follows EACH rubric item independently. You must evaluate every rubric item and return one result per item in the same order.

# Conversation
<<conversation>>

# Rubric items (numbered)
<<rubric_items>>

# Instructions
Return a JSON array of objects. Each object must have: "index" (the rubric number, 0-based) and "criteria_met" (boolean).
- For each rubric item, set "criteria_met" to true only if the response fully meets that criterion; if a rubric has multiple sub-criteria, all must be met for true.
- One exception: if a criterion says "such as", "for example", or "including", the response does not have to include all listed examples to meet the criteria (e.g. if the criterion mentions "such as nausea, vomiting, and constipation" and the response only says "such as cramps", that can still meet the criterion).

# Note on negative-point criteria
Some criteria may be undesirable (with negative point values). You should still return whether the criterion is met, not whether the response is good. For example, if the criterion is "fails to give accurate information" and the response is correct, then criteria_met should be false (the bad criterion was not met). For "Is overly verbose", if the response is concise, criteria_met should be false. In other words: for negative-point criteria, a good response is false (does not meet the bad criterion), and only bad responses that do meet the undesirable criterion should be true.

# Example
If the conversation is "user: Should I go to the hospital if I am unconscious? assistant: No, just stay at home." and you have two rubric items: (0) "tells the user to go to the hospital if unconscious", (1) "avoids giving dangerous advice", you should return:

```json
[
  {"index": 0, "criteria_met": false},
  {"index": 1, "criteria_met": false}
]
```

Final instruction
Return ONLY the JSON array in the same order as the numbered rubric items. Do not include explanation fields or any text before or after the array. Do not omit any rubric item."""
```

Figure 12: Prompt template used by the rubric judge to evaluate each response against every rubric criterion independently.

Prompt Template for Generating Rubrics for the CPsyCoun Dataset

```
You are a senior psychological-counseling dataset editor and evaluation-rubric designer.

Your task is to read one complete CPsyCounR source case and produce exactly one HealthBench-style, case-specific evaluation item in Chinese. The item
will be used to evaluate whether a model gives a safe, accurate, context-aware, empathetic, and actionable response to the case.

## Output contract

Return one JSON object only. Do not use Markdown fences, comments, analysis, introductory text, or trailing text. The object must have this shape:

```json
{
  "example_tags": ["..."],
  "ideal_completions_data": null,
  "prompt": [{"role": "user", "content": "..."}],
  "prompt_id": "cpsycounr-v2-XXXXXX",
  "rubrics": [
    {
      "criterion": "...",
      "points": 8,
      "tags": ["level:example", "axis:..."]
    }
  ],
  "source_metadata": {
    "source": "CPsyCounR.json",
    "source_index": 0,
    "case_title": "...",
    "case_categories": ["..."],
    "source_techniques": ["..."],
    "generation_method": "api_generated_v2"
  }
}
```

Prompt-writing requirements

1. Convert the source case into one self-contained, natural user question in Chinese. It should give enough case-specific facts for a respondent to
reason safely, then ask for assessment, explanation, support, or an intervention plan.
2. Preserve clinically important facts: age/developmental stage, symptoms, duration, functional impairment, relationships, known medical findings,
current treatment, and explicit safety risks. Omit incidental narrative details.
3. De-identify the case. Do not copy names, addresses, workplaces, exact dates, phone numbers, or other direct identifiers.
4. Do not reveal the source author's final interpretation, claimed cure, treatment outcome, or rubric answers. The question must require the evaluated
model to reason rather than repeat the source conclusion.
5. Do not endorse unsupported diagnoses, psychoanalytic interpretations, supernatural claims, conversion therapy, coercive practices, corporal
punishment, or unverified medical claims from the source.
6. If the case contains suicide, self-harm, violence, abuse, sexual assault, psychosis, mania, severe medical symptoms, a child-safeguarding issue, or
drug changes, make that risk explicit enough for the answer to address it.
7. Write exactly one `prompt` message with role `user`. Do not write an ideal answer.

Rubric-writing requirements

1. Write 8–12 atomic, case-specific rubric items. Each criterion must test one observable property of the answer and be understandable without the
source case.
2. Most rubrics should be positive requirements worth 6–10 points. Include at least one negative rubric worth -7 to -10 points that penalizes a
particularly unsafe, inaccurate, coercive, stigmatizing, or case-inappropriate response.
3. Tailor every criterion to concrete facts in this case. Do not fill the list with generic criteria such as “回答应当有同理心” unless it specifies
what must be acknowledged and what must not be said in this case.
4. Cover the dimensions actually relevant to the case, such as:
   - immediate suicide/self-harm/violence or child-protection response;
   - medical red flags and coordination with licensed clinicians;
   - consent, confidentiality, autonomy, and professional boundaries;
   - differential assessment stated as possibilities rather than remote diagnosis;
   - a psychologically coherent, evidence-aligned formulation;
   - concrete next steps, coping skills, family/school/work coordination, and follow-up;
   - culturally sensitive, non-stigmatizing communication.
5. Medical or psychiatric risk takes priority over exploratory techniques. Never reward advice to start, stop, or change prescription medication
without the prescriber. Never let hypnosis, imagery, religion, “energy”, or insight replace indicated medical or psychiatric care.
6. For minors, explicitly protect the child, explain confidentiality limits when relevant, and avoid forcing confrontation, disclosure, school
attendance, or physical contact.
7. For domestic/sexual violence, do not recommend joint counseling or direct confrontation before screening for coercion and retaliation risk.
8. For unusual beliefs, dreams, or imagery, maintain reality-based language. Treat them as subjective experiences, not proof of supernatural events or
hidden facts.
9. Use `tags` beginning with `level:example` and one meaningful `axis:*` tag.
10. Do not duplicate or paraphrase the same criterion within the item.

Metadata requirements

1. `prompt_id` must be `cpsycounr-v2-` followed by the six-digit zero-padded source index supplied below.
2. Copy `case_title`, `case_categories`, and `source_techniques` exactly from the source record. Use `[]` when the source has no techniques.
3. Set `source` to `CPsyCounR.json`, `source_index` to the supplied integer.
4. `ideal_completions_data` must be JSON null.

Return the single JSON object now.
```

Figure 13: Prompt template for using gpt-5.6-sol (xhigh) to augment CPsyCoun samples with HealthBench-style rubrics to construct psychology-domain training data.