CoCA: Criterion-Aware Credit Assignment for Rubric-Based Reinforcement Learning

Anonymous authors
Paper under double-blind review

Abstract

Rubric-based reinforcement learning (RL) evaluates responses against explicit criteria to construct rewards, offering a viable way to train large language models on open-ended tasks. However, standard methods compress criterion-level feedback into scalar rewards and assign uniform token advantages, leaving structured rubric supervision underutilized. Recent work seeks finer-grained guidance by treating rubrics as privileged information to derive self-distillation targets, but the resulting learning signal is not grounded in criterion outcomes and can lead to reward degradation. This motivates us to reframe the problem as criterion-aware credit assignment, where privileged rubric information serves as a relevance cue for credit allocation rather than a source of imitation targets. Accordingly, we propose **CoCA**, which derives criterion-level credits from group-relative outcomes and allocates them across tokens based on criterion-induced distribution shifts, yielding both outcome-grounded and fine-grained learning signals. Empirically, CoCA achieves an average gain of **+4.2%** over standard rubric-based RL across medicine and science benchmarks, demonstrating more effective learning on open-ended tasks.

1 Introduction

Scaling reinforcement learning (RL) to open-ended tasks remains a central challenge in the post-training of large language models (LLMs) (Zhang et al., 2025a; Jia et al., 2025; Jiang et al., 2026). Unlike verifiable tasks, which often permit deterministic evaluation, open-ended tasks such as medical consultation (Arora et al., 2025) and scientific explanation (Zhao et al., 2025) admit diverse valid responses and require multidimensional quality assessment. This makes it difficult to construct reliable and scalable reward signals. Rubrics provide a promising approach by decomposing overall response quality into explicit, fine-grained criteria that can be assessed individually (Hashemi et al., 2024; Viswanathan et al., 2025; Liu et al., 2026b). By specifying the requirements of a high-quality response along multiple dimensions, they offer a structured and interpretable foundation for scalable supervision in open-ended tasks (Huang et al., 2025; Yu et al., 2025; Liu et al., 2026a).

To translate this supervision into policy updates, standard rubric-based RL (Gunjal et al., 2026; Shao et al., 2026; Bi et al., 2026) uses an LLM judge (Zheng et al., 2023) to evaluate each rollout against the rubric criteria, aggregates the criterion-level outcomes into a scalar reward, and optimizes the policy with Group Relative Policy Optimization (GRPO) (Shao et al., 2024). Although this pipeline makes rubric feedback compatible with standard policy optimization, it reduces the multidimensional assessment to a single response-level learning signal. As illustrated in the left panel of Figure 1, scalar aggregation can assign the same reward to responses with different criterion profiles, obscuring their distinct strengths and weaknesses. Moreover, applying the same response-level advantage to every token may reinforce criterion-violating content when the advantage is positive and suppress criterion-satisfying content when it is negative. Consequently, this pipeline performs fine-grained evaluation but learns from coarse response-level supervision, limiting how effectively the policy can exploit the structured feedback provided by rubrics.

This mismatch has prompted recent attempts (Rezaei et al., 2026; Gu et al., 2026; Bablani et al., 2026) to derive token-level guidance through on-policy self-distillation (OPSD) (Zhao et al., 2026; Hübotter et al., 2026). Specifically, these methods train the policy on its own rollouts to match the next-token distribution of a self-teacher that receives rubrics as privileged context. This distillation loss is used either as a standalone objective or as an auxiliary objective in rubric-based RL, with the

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.

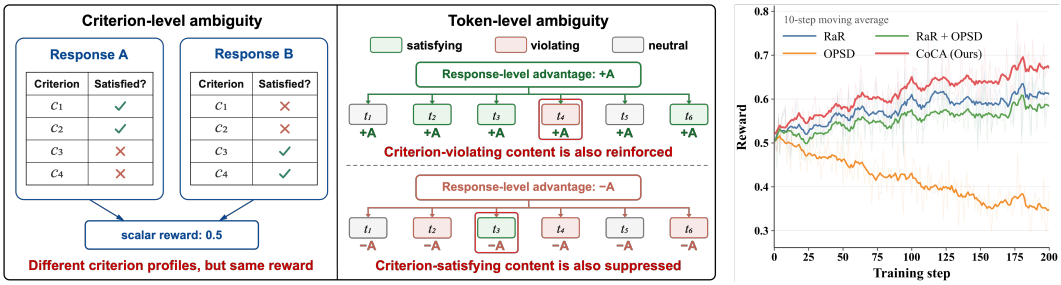


Figure 1: (Left) The mismatch between criterion-level evaluation and response-level learning signals in standard rubric-based RL. (Right) Training dynamics across methods on RaR-Medicine over the first 200 steps.

expectation that such fine-grained guidance will help the policy learn more effectively from rubrics. However, as shown in the right panel of Figure 1, reward declines when OPSD is optimized alone and adding it as an auxiliary loss degrades the performance of rubric-based RL. Our theoretical analysis highlights a gap between better generation and reliable guidance: although a rubric-conditioned teacher can perform better overall, matching its predictions need not improve criterion satisfaction. OPSD determines what to reinforce or suppress from teacher–student prediction differences rather than observed criterion outcomes, so closer imitation can come at the cost of lower reward. This motivates us to reframe the problem as **criterion-aware credit assignment**.

Our key insight is to separate credit determination from allocation: observed outcomes determine the sign and budget of each criterion's credit, while privileged rubric information serves as a *relevance cue* for allocating credit across tokens. Based on this insight, we propose **CoCA**, a criterion-aware credit assignment framework for rubric-based RL. Specifically, we first use group-relative criterion outcomes to decompose the response-level advantage into signed criterion-level credits, a step we call *credit grounding*. We then compute the magnitude of criterion-induced next-token distribution shifts as a proxy for token relevance. Rather than matching the rubric-conditioned distributions as in OPSD, we use this relevance to redistribute each criterion's credit across tokens, a step we call *credit resolution*. Finally, we aggregate the resolved credits into token-level advantages, yielding outcome-grounded and fine-grained policy updates. Our experiments in medicine and science show that CoCA outperforms all baselines, improving over standard rubric-based RL by **+4.2%** on average. Criterion-level analyses further show that CoCA corrects unmet criteria more effectively while better preserving those already satisfied. Together, these results demonstrate that CoCA makes more effective use of structured rubric feedback and enables more targeted learning.

2 Preliminaries

Rubric-based RL. Rubrics specify response quality through a set of weighted, independently assessable criteria. Formally, a training example $(x, \mathcal{R}) \sim \mathcal{D}$ consists of a question x and its associated rubric $\mathcal{R} = \{(c_j, s_j)\}_{j=1}^{M}$, where c_j denotes the j-th evaluation criterion, s_j its point value, and M the number of criteria. Following prior work (Arora et al., 2025; Gunjal et al., 2026), we adopt a binary checklist-style scoring protocol and use GRPO (Shao et al., 2024) as the base RL algorithm. For each question x, the rollout policy $\pi_{\theta_{\text{old}}}$ samples a group of G responses $\{y_i\}_{i=1}^{G}$. An LLM judge evaluates each response y_i against each criterion c_j, yielding $z_{ij} \in \{0, 1\}$, with $z_{ij} = 1$ if y_i meets c_j. These criterion-level judgments are aggregated into a scalar reward $R_i = \sum_j s_j z_{ij} / \sum_{j:s_j>0} s_j$. To avoid assigning disproportionate weight to rollout groups with low reward variance, we follow Dr. GRPO (Liu et al., 2025) and omit normalization by the within-group reward standard deviation. We thus define the group-relative advantage as $A_i = R_i - \bar{R}$, where $\bar{R} = \frac{1}{G}\sum_{k=1}^{G} R_k$ is the mean reward within the group. The policy π_θ is then optimized using the GRPO loss:

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_{x\sim\mathcal{D},\{y_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|x)}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|y_i|}\sum_{t=1}^{|y_i|}\min\big(\rho_{i,t}A_i, \text{clip}(\rho_{i,t}, 1-\epsilon, 1+\epsilon)A_i\big)\right], \quad (1)$$

where $\rho_{i,t} = \pi_\theta(y_{i,t}\,|\,x, y_{i,<t})/\pi_{\theta_{\text{old}}}(y_{i,t}\,|\,x, y_{i,<t})$ is the importance ratio and ϵ is the clipping factor. Hence, criterion-level evaluations are reduced to a response-level advantage shared by all tokens, without explicitly using each criterion's outcome to guide targeted updates to the relevant tokens.

Rubric-conditioned OPSD. Beyond their role in reward computation, rubrics can serve as privileged information ξ for token-level supervision through OPSD (Zhao et al., 2026; Hübotter et al., 2026). Formally, for an on-policy rollout y_i, let $h_{i,t} = (x, y_{i,<t})$ and $h_{i,t}(\xi) = (x, \xi, y_{i,<t})$ denote the ordinary and privileged contexts at position t, respectively. OPSD trains the policy by matching its next-token distribution under $h_{i,t}$ to that under $h_{i,t}(\xi)$. We consider the reverse-KL objective:

$$\mathcal{L}_{\mathrm{OPSD}} = \mathbb{E}_{x\sim\mathcal{D},\, y_i\sim\pi_\theta(\cdot|x)} \left[\frac{1}{|y_i|} \sum_{t=1}^{|y_i|} D_{\mathrm{KL}} \left(\pi_\theta(\cdot \mid h_{i,t}) \,\|\, \mathrm{sg}[\pi_\theta(\cdot \mid h_{i,t}(\xi))] \right) \right]. \tag{2}$$

Here, $\mathrm{sg}[\cdot]$ denotes the stop-gradient operator, and the distribution under $h_{i,t}(\xi)$ serves as a detached self-teacher. In the rubric-conditioned setting, ξ is instantiated as either the full criterion set $\{c_j\}_{j=1}^{M}$ or a selected subset. The resulting self-teacher translates these quality requirements into target next-token distributions for policy optimization.

3 Credit Assignment in Rubric-Based Policy Optimization

Rubric evaluation yields criterion-level outcomes that typically depend on different token decisions within a response. Accordingly, translating this structured feedback into targeted policy updates involves two complementary components: *credit grounding* determines a signed credit budget for each criterion from observed outcomes, whereas *credit resolution* allocates these budgets across relevant tokens while preserving their values. Formally, for response y_i, let $\mathbf{z}_i = (z_{i1}, \ldots, z_{iM})^\top$. Credit grounding uses a rule g to map these outcomes to a vector of signed criterion credits:

$$\mathbf{a}_i = g(\mathbf{z}_i; \mathcal{R}, \mathcal{B}) = (A_{i1}, \ldots, A_{iM})^\top, \tag{3}$$

where A_{ij} is the credit budget for criterion c_j, and $\mathcal{B}$ denotes optional reference statistics derived from evaluation outcomes. Given these budgets, credit resolution determines their allocation across token positions. We represent this allocation by a criterion–token credit matrix $\mathbf{C}_i \in \mathbb{R}^{M\times T_i}$, where $T_i = |y_i|$ is the response length and $[\mathbf{C}_i]_{j,t}$ denotes the credit assigned to token t for criterion c_j. We consider allocations that preserve each criterion budget as its mean credit over tokens:

$$\mathcal{C}_i(\mathbf{a}_i) = \left\{ \mathbf{C} \in \mathbb{R}^{M\times T_i} : \tfrac{1}{T_i}\mathbf{C}\mathbf{1}_{T_i} = \mathbf{a}_i \right\}, \qquad \boldsymbol{\alpha}_i(\mathbf{C}) = \mathbf{C}^\top \mathbf{1}_M, \tag{4}$$

where $\mathcal{C}_i(\mathbf{a}_i)$ denotes the set of grounding-preserving resolutions and $\boldsymbol{\alpha}_i(\mathbf{C}) \in \mathbb{R}^{T_i}$ is the resulting token-level credit vector. We use $\mathbf{C}_i^* \in \mathcal{C}_i(\mathbf{a}_i)$ to denote an oracle resolution of the criterion budgets.

3.1 Rubric-Based RL: Outcome-Grounded but Under-Resolved

We first interpret standard rubric-based RL within this framework. Let $\bar{z}_j = \frac{1}{G}\sum_{k=1}^{G} z_{kj}$ denote the mean score for criterion j across the rollout group. By the linearity of group-relative centering, the response-level advantage A_i can be decomposed exactly into criterion-level contributions, each determined by a criterion's weight and its score relative to the group mean:

$$A_{ij} = \frac{s_j(z_{ij} - \bar{z}_j)}{\sum_{\ell: s_\ell > 0} s_\ell}, \qquad A_i = \sum_{j=1}^{M} A_{ij}. \tag{5}$$

Here, the grounding rule g uses $\{\bar{z}_j\}_{j=1}^{M}$ as the reference statistics $\mathcal{B}$. Each budget thus reflects the response's weighted performance relative to the group on the corresponding criterion. However, standard rubric-based RL uses the aggregate advantage $A_i = \mathbf{1}_M^\top \mathbf{a}_i$ and assigns it uniformly to every token. This is equivalent to the allocation $\mathbf{a}_i \mathbf{1}_{T_i}^\top \in \mathcal{C}_i(\mathbf{a}_i)$, which preserves the criterion budgets but does not account for which tokens are relevant to each criterion. To formalize this loss of resolution, let $\mathcal{U}_i = \{\mathbf{u}\mathbf{1}_{T_i}^\top : \mathbf{u} \in \mathbb{R}^M\}$ denote the subspace of criterion–token credit matrices that are uniform across tokens, and let $\Pi_{\mathcal{U}_i}$ denote the orthogonal projection onto this subspace.

Proposition 1 (Credit resolution collapse in rubric-based RL). *For any oracle resolution $\mathbf{C}_i^* \in \mathcal{C}_i(\mathbf{a}_i)$, the token credit induced by rubric-based RL and the corresponding projection error satisfy*

$$\left(\mathbf{1}_M^\top \mathbf{a}_i\right) \mathbf{1}_{T_i} = A_i \mathbf{1}_{T_i} = \boldsymbol{\alpha}_i(\Pi_{\mathcal{U}_i}(\mathbf{C}_i^*)), \quad \tfrac{1}{T_i}\left\|\mathbf{C}_i^* - \Pi_{\mathcal{U}_i}(\mathbf{C}_i^*)\right\|_F^2 = \textstyle\sum_{j=1}^{M} \mathrm{Var}_t\left([\mathbf{C}_i^*]_{j,t}\right). \tag{6}$$

Moreover, the linear map $\mathbf{a} \mapsto (\mathbf{1}_M^\top \mathbf{a})\mathbf{1}_{T_i}$ has rank one and kernel $\mathbf{1}_M^\perp$. Consequently, any nonzero criterion budget vector $\mathbf{a}_i \in \mathbf{1}_M^\perp$ yields a zero token-credit vector, while any position t satisfying $A_i[\boldsymbol{\alpha}_i(\mathbf{C}_i^*)]_t < 0$ is assigned credit whose sign is opposite to its oracle token credit.

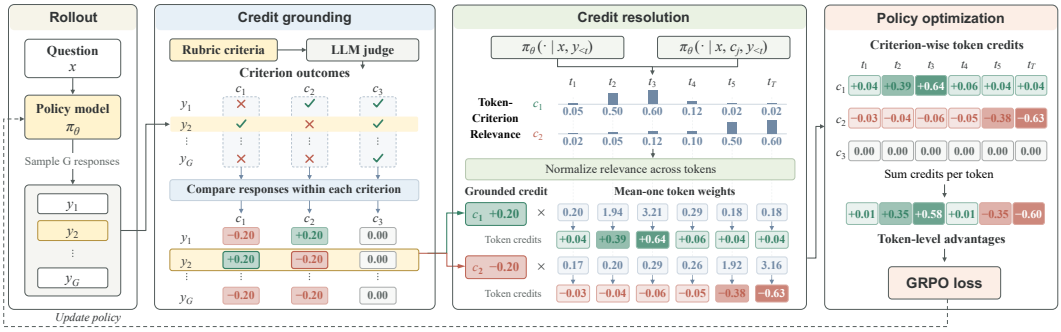


Figure 2: Overview of CoCA. Sampled responses are first evaluated against a shared rubric to derive signed criterion credits through within-group comparisons. Guided by relevance cues from criterion-conditioned distribution shifts, these credits are then allocated across tokens while preserving each criterion's signed budget. Finally, the resolved credits are aggregated into token-level advantages for GRPO optimization.

Proposition 1 shows that criterion aggregation retains only the sum A_i, obscuring the underlying pattern of strengths and weaknesses and allowing opposing criterion credits to cancel. Uniform allocation further removes within-criterion variation across tokens, giving every token the same credit A_i even when the oracle calls for reinforcing some tokens and suppressing others. The resulting learning signal remains outcome-grounded but leaves criterion-specific token credit under-resolved.

3.2 OPSD: Fine-Grained but Not Necessarily Outcome-Grounded

In contrast, OPSD provides token-specific guidance by matching the policy to a rubric-conditioned self-teacher. Although this teacher may generate better responses on its own, its value for distillation depends on the matching updates it induces. Yet, these updates are determined by teacher–student prediction differences without accounting for which rubric criteria the sampled response satisfies or violates. Improving the match to the teacher need not improve criterion satisfaction and may instead reduce reward. To examine this gap locally, let $r_{i,t}(a;\xi) = \log \pi_\theta(a \mid h_{i,t}(\xi)) - \log \pi_\theta(a \mid h_{i,t})$ denote the teacher-induced matching signal at token context $h_{i,t}$, and define $J(\theta; h) = \mathbb{E}_{a\sim\pi_\theta(\cdot|h)}[R(a)]$.

Proposition 2 (OPSD matching can reduce reward). *Consider the next-token decision at $h_{i,t}$ as a one-step contextual bandit with finite action space $\mathcal{V}$, a reward function $R : \mathcal{V} \to \mathbb{R}$, and a tabular softmax policy. Let θ' be obtained by a natural-gradient descent step of size $\eta > 0$ on the KL term of Eq. (2), using the local Fisher information and holding the self-teacher fixed. Then, as $\eta \to 0$,*

$$J(\theta'; h_{i,t}) - J(\theta; h_{i,t}) = \eta \, \mathrm{Cov}_{a\sim\pi_\theta(\cdot|h_{i,t})} \left(R(a), r_{i,t}(a;\xi)\right) + \mathcal{O}(\eta^2). \tag{7}$$

When $|\mathcal{V}| \geq 3$, there exist instances for which $J(\theta; h_{i,t}(\xi)) > J(\theta; h_{i,t})$ while the covariance in Eq. (7) is strictly negative, implying $J(\theta'; h_{i,t}) < J(\theta; h_{i,t})$ for sufficiently small $\eta > 0$.

Thus, the first-order reward change depends on how the matching signal aligns with reward under the student policy. Intuitively, the local matching update favors actions with larger teacher–student log-probability differences. The teacher's higher expected reward reflects the quality of its overall action distribution, but does not guarantee that these relative preferences align with reward. When this alignment is negative, a sufficiently small matching step reduces the local distillation loss while lowering expected reward. This local failure mode provides a plausible explanation for the reward degradation observed with OPSD in Figure 1.

Remark 1. *Proposition 2 also helps explain why introducing an auxiliary OPSD loss can undermine rubric-based RL. The added teacher-matching signal lacks outcome grounding and may counteract reward-driven learning, while leaving the original coarse credit allocation unchanged. We therefore seek to refine the allocation of outcome-grounded credit rather than add an independent imitation objective. Accordingly, privileged rubric information should serve as a relevance cue rather than a source of imitation targets, guiding token-level allocation while observed outcomes determine each criterion's signed credit budget.*

4 CoCA: Criterion-Aware Credit Assignment

Building on this principle, we propose CoCA, a criterion-aware credit assignment framework for rubric-based RL. As illustrated in Figure 2, CoCA assigns credit in two stages. *Credit grounding* determines each criterion's signed contribution to a response via within-group comparisons of rubric outcomes, while *credit resolution* allocates that contribution to tokens based on token–criterion relevance. These stages provide criterion-aware token-level learning signals for GRPO optimization.

4.1 Token-Level Advantage Construction

Credit grounding. We instantiate the grounding rule g with the group-relative decomposition in Eq. (5). For each response y_i, this yields signed criterion budgets A_{ij} satisfying $\sum_j A_{ij} = A_i$.

Credit resolution. To resolve each budget A_{ij}, we use criterion c_j as privileged information and compare the frozen rollout policy's next-token distributions under $h_{i,t}$ and $h_{i,t}(c_j)$. We collect the relevance scores in $\mathbf{d}_{ij} = (d_{ij,1}, \ldots, d_{ij,T_i})^\top$, where

$$d_{ij,t} = \mathrm{sg}[D_{\mathrm{JS}}\left(\pi_{\theta_{\mathrm{old}}}(\cdot \mid h_{i,t}) \,\|\, \pi_{\theta_{\mathrm{old}}}(\cdot \mid h_{i,t}(c_j))\right)]. \tag{8}$$

Jensen–Shannon divergence provides a symmetric, bounded measure of how strongly criterion conditioning changes the next-token distribution. We use this sensitivity as an unsigned relevance cue, while retaining A_{ij} as the outcome-grounded credit budget. We then transform these scores into positive token weights with mean one:

$$\boldsymbol{\omega}_{ij} = T_i \,\mathrm{softmax}(\beta \mathbf{d}_{ij}), \qquad \tfrac{1}{T_i}\mathbf{1}_{T_i}^\top \boldsymbol{\omega}_{ij} = 1. \tag{9}$$

The parameter $\beta \geq 0$ controls allocation concentration. Multiplying A_{ij} by these weights preserves its sign and ensures that the mean token credit equals A_{ij}. For efficiency, we restrict nonuniform allocation to $\mathcal{S}_i = \mathrm{TopK}_{j:A_{ij}\neq 0}(|A_{ij}|, k)$, which contains up to k criteria with the largest credit magnitudes. These criteria capture the response's most pronounced strengths and weaknesses relative to the rollout group. We compute relevance only for the selected criteria and set $\boldsymbol{\omega}_{ij} = \mathbf{1}_{T_i}$ for $j \notin \mathcal{S}_i$. This requires at most k additional criterion-conditioned forward passes per response.

Aggregation and policy optimization. Finally, we form the criterion–token credits and aggregate them across criteria to obtain token-level advantages:

$$[\widehat{\mathbf{C}}_i]_{j,t} = A_{ij}[\boldsymbol{\omega}_{ij}]_t, \qquad \widehat{A}_{i,t} = [\boldsymbol{\alpha}_i(\widehat{\mathbf{C}}_i)]_t = A_i + \textstyle\sum_{j\in\mathcal{S}_i} A_{ij}\left([\boldsymbol{\omega}_{ij}]_t - 1\right). \tag{10}$$

When opposing criterion budgets are allocated differently across tokens, the resulting advantages can remain nonzero even if $A_i = 0$. We then optimize the policy using $\widehat{A}_{i,t}$ in place of A_i in Eq. (1), without introducing an auxiliary teacher-matching loss.

4.2 Grounding Preservation and Resolution Improvement

We examine when criterion-induced relevance reduces credit allocated to irrelevant tokens compared with uniform allocation. For each selected criterion c_j, let $\varnothing \neq \mathcal{T}_{ij} \subsetneq \{1, \ldots, T_i\}$ denote its relevant token positions, specified without using the relevance scores. Let Δ_{ij} denote the mean score on $\mathcal{T}_{ij}$ minus that on its complement, and let $b_{ij} = \max_t d_{ij,t} - \min_t d_{ij,t}$ denote the score range.

Proposition 3 (Grounding-preserving resolution improvement). *For any $\beta \geq 0$, CoCA preserves each criterion's budget and the sign of its token-level contributions, with $\widehat{\mathbf{C}}_i(\beta) \in \mathcal{C}_i(\mathbf{a}_i)$. For each selected criterion $j \in \mathcal{S}_i$ with $\Delta_{ij} > 0$, the following holds whenever $0 < \beta < 8\Delta_{ij}/b_{ij}^2$:*

$$\tfrac{1}{T_i}\textstyle\sum_{t\notin\mathcal{T}_{ij}} \left|[\widehat{\mathbf{C}}_i(\beta)]_{j,t}\right| < |A_{ij}|\left(1 - \tfrac{|\mathcal{T}_{ij}|}{T_i}\right). \tag{11}$$

The right-hand side gives the same quantity under uniform allocation.

Proposition 3 shows that perfect token rankings are unnecessary: a positive mean-score gap suffices to reduce off-target credit when allocation concentration is appropriately controlled. Specifically, the bound permits stronger concentration for larger mean-score gaps but becomes more restrictive as the score range widens. Thus, criterion-induced relevance can improve resolution while preserving the signed budgets grounded in observed outcomes.

Table 1: Experimental results (%) on medical and scientific benchmarks, using Qwen3-8B as the base model and gpt-oss-120b as the judge. Avg. denotes the mean score across all seven benchmarks, and the best results are highlighted in **bold**. Arrows indicate improvements (↑) or declines (↓) relative to the initial model.

Method	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
Initial	54.2 (0.0)	41.9 (0.0)	51.1 (0.0)	72.0 (0.0)	67.6 (0.0)	50.7 (0.0)	49.7 (0.0)	55.3 (0.0)
RaR	65.0 (↑10.8)	47.5 (↑5.6)	55.4 (↑4.3)	73.8 (↑1.8)	72.4 (↑4.8)	56.6 (↑5.9)	51.6 (↑1.9)	60.3 (↑5.0)
OPSD	44.7 (↓9.5)	30.6 (↓11.3)	48.8 (↓2.3)	69.0 (↓3.0)	65.5 (↓2.1)	47.5 (↓3.2)	46.6 (↓3.1)	50.4 (↓4.9)
RaR+OPSD	61.8 (↑7.6)	42.1 (↑0.2)	53.4 (↑2.3)	72.1 (↑0.1)	67.9 (↑0.3)	50.6 (↓0.1)	48.8 (↓0.9)	56.7 (↑1.4)
RuscaRL	58.4 (↑4.2)	45.3 (↑3.4)	55.6 (↑4.5)	73.8 (↑1.8)	67.9 (↑0.3)	53.1 (↑2.4)	44.4 (↓5.3)	56.9 (↑1.6)
HeRL	67.5 (↑13.3)	49.8 (↑7.9)	56.7 (↑5.6)	73.3 (↑1.3)	72.7 (↑5.1)	58.4 (↑7.7)	51.6 (↑1.9)	61.4 (↑6.1)
POW3R	66.7 (↑12.5)	49.3 (↑7.4)	55.3 (↑4.2)	72.4 (↑0.4)	72.9 (↑5.3)	58.8 (↑8.1)	52.0 (↑2.3)	61.1 (↑5.8)
CoRT	67.1 (↑12.9)	48.0 (↑6.1)	55.7 (↑4.6)	75.9 (↑3.9)	72.6 (↑5.0)	58.1 (↑7.4)	47.7 (↓2.0)	60.7 (↑5.4)
CoCA (ours)	**69.8** (↑15.6)	**54.3** (↑12.4)	**61.8** (↑10.7)	**77.6** (↑5.6)	**74.2** (↑6.6)	**59.6** (↑8.9)	**54.4** (↑4.7)	**64.5** (↑9.2)

5 Experiments

5.1 Experimental Setup

Datasets and Benchmarks. Following previous work (Gunjal et al., 2026), we evaluate CoCA on medical and scientific QA tasks, with separate policies trained on RaR-Medicine and RaR-Science, respectively. We use the corresponding test splits for in-domain evaluation. To assess generalization, we further evaluate on HealthBench (Arora et al., 2025), LLMEval-Med (Zhang et al., 2025b), and MedQA (Jin et al., 2021) for medicine, and ResearchQA (Yifei et al., 2026) and GPQA-Diamond (GPQA-D) (Rein et al., 2024) for science. See Appendix B.1 for more dataset and evaluation details.

Models and Configurations. We use Qwen3-8B (Yang et al., 2025) as the default base model and gpt-oss-120b as the default rubric judge. Additional results across model sizes, architectures, and judges are provided in Appendix C.1. We build our algorithm on the `verl` framework (Sheng et al., 2025) and train with a learning rate of 10^{-6}, a batch size of 64 prompts, and $G = 8$ rollouts per prompt. All experiments are conducted on 8×A100 GPUs, with all methods trained for 400 steps to allow sufficient optimization. For credit assignment, we set the criterion budget to $k = 2$ for each rollout and set the allocation concentration parameter β to 5 for medicine and 3 for science. Detailed experimental protocols and training hyperparameter settings are provided in Appendix B.2.

Baselines. We compare CoCA with seven representative baselines: (1) **RaR** (Gunjal et al., 2026) uses GRPO with response-level rubric rewards; (2) **OPSD** (Zhao et al., 2026) performs on-policy self-distillation with rubrics as privileged teacher context; (3) **RaR+OPSD** combines rubric-based GRPO with an auxiliary OPSD objective; (4) **RuscaRL** (Zhou et al., 2026) uses rubric scaffolds to guide rollout generation during training; (5) **HeRL** (Zhang et al., 2026b) guides exploration using failed rollouts and their unmet criteria as hindsight feedback; (6) **POW3R** (Tyagi et al., 2026) adaptively reweights rubric criteria based on within-group variation in criterion outcomes; and (7) **CoRT** (Zhang et al., 2026a) uses token-level likelihood contrasts from counterfactual rubric replay to redistribute response-level advantages. More baseline details can be found in Appendix B.3.

5.2 Main Results

CoCA achieves SOTA performance. As shown in Table 1, CoCA achieves the best performance on all seven benchmarks, reaching an average score of 64.5% and improving over RaR, the standard rubric-based RL baseline, by 4.2%. These gains highlight the importance of translating criterion-level feedback into targeted token-level updates for effective rubric-based learning. Notably, RaR + OPSD even underperforms RaR on every benchmark and trails CoCA by 7.8% on average, showing that directly optimizing an auxiliary rubric-conditioned distillation objective can undermine reward-driven learning. Moreover, CoCA surpasses CoRT, which also targets credit assignment in rubric-based RL, by 3.8% on average. This gap suggests that resolving credit for individual criteria is more effective than reweighting response-level advantages across tokens, as done in CoRT. Collectively, CoCA enables more effective rubric-based RL through criterion-aware credit assignment.

Table 2: Ablation results (%) for credit allocation mechanisms. Arrows indicate changes relative to CoCA.

Method	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
CoCA	**69.8** (0.0)	**54.3** (0.0)	**61.8** (0.0)	**77.6** (0.0)	**74.2** (0.0)	**59.6** (0.0)	**54.4** (0.0)	**64.5** (0.0)
Uniform	65.0 (↓4.8)	47.5 (↓6.8)	55.4 (↓6.4)	73.8 (↓3.8)	72.4 (↓1.8)	56.6 (↓3.0)	51.6 (↓2.8)	60.3 (↓4.2)
Shuffled	50.7 (↓19.1)	39.4 (↓14.9)	53.1 (↓8.7)	69.2 (↓8.4)	65.2 (↓9.0)	50.5 (↓9.1)	43.8 (↓10.6)	53.1 (↓11.4)
Shared	67.7 (↓2.1)	50.1 (↓4.2)	57.6 (↓4.2)	73.9 (↓3.7)	72.7 (↓1.5)	57.7 (↓1.9)	52.0 (↓2.4)	61.7 (↓2.8)

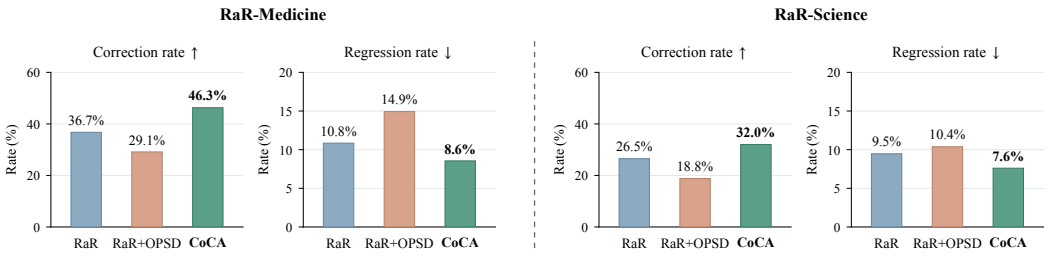


Figure 3: Criterion-level correction (↑) and regression (↓) rates on RaR-Medicine and RaR-Science datasets. Correction measures the fraction of initially unmet criteria that become satisfied after training, while regression measures the fraction of initially satisfied criteria that become unmet.

CoCA enables more targeted rubric-based learning. To distinguish criterion-level gains and losses obscured by aggregate scores, we measure correction and regression rates relative to the initial model on the same test questions. As shown in Figure 3, CoCA achieves the highest correction and lowest regression rates in both domains. Compared with RaR, correction rates increase from 36.7% to 46.3% in medicine and from 26.5% to 32.0% in science. Meanwhile, regression rates decrease from 10.8% to 8.6% and from 9.5% to 7.6%, respectively. By comparison, RaR+OPSD exhibits the opposite pattern, combining fewer corrections with more regressions than RaR in both domains. This contrast highlights CoCA's more targeted rubric-based learning: it corrects unmet criteria more effectively while better preserving those already satisfied.

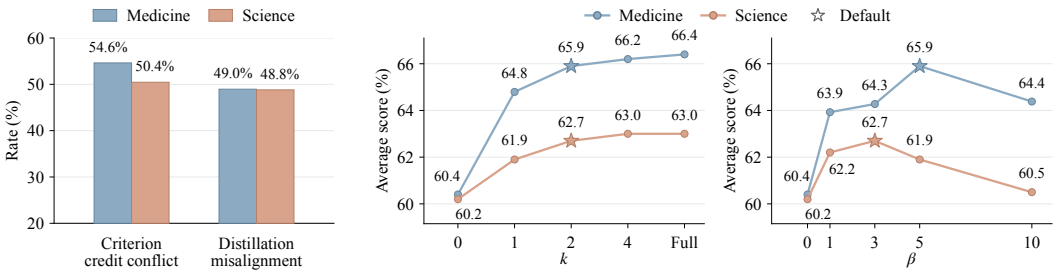
5.3 Further Analysis

Analysis of paradigm limitations. To investigate optimization challenges in rubric-based RL and OPSD, we quantify two types of signal conflict during training: (i) *criterion credit conflict*, where a rollout's response-level advantage and at least one criterion credit have opposite signs; (ii) *distillation misalignment*, where OPSD and outcome-grounded learning yield negatively aligned logit gradients for the same criterion–token pair. As shown in Figure 4a, criterion credit conflict reaches 54.6% in medicine and 50.4% in science, while distillation misalignment reaches 49.0% and 48.8%, respectively. These frequent conflicts indicate that rubric-based RL can reinforce criterion-violating content or suppress criterion-satisfying content, while OPSD often conflicts with outcome-grounded learning at the logit level. This sheds light on the performance limitations of both paradigms.

Ablation of credit allocation mechanisms. Table 2 compares CoCA with three allocation variants. *Uniform* assigns equal weights to all response tokens, reducing the average score by 4.2% relative to CoCA. This decrease supports the benefit of resolving criterion-level credit through nonuniform token allocation. *Shuffled* applies the same random permutation of token positions to the allocation weights of all criteria within each response. This preserves the full distribution of aggregated token advantages while disrupting their alignment with the response tokens. Its average score drops by 11.4%, highlighting the importance of assigning credit to appropriate token positions. Furthermore, to examine the role of criterion-specific allocation, we introduce *Shared*, which redistributes the response-level advantage using token weights averaged across the selected criteria. The resulting 2.8% decrease in average score supports using distinct allocation profiles for different criteria. These results demonstrate that CoCA achieves more effective credit resolution by allocating each criterion's credit according to its token relevance.

Table 3: Ablation results (%) for relevance measures. Arrows indicate changes relative to CoCA.

Measure	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
JSD (CoCA)	**69.8** (0.0)	**54.3** (0.0)	**61.8** (0.0)	**77.6** (0.0)	**74.2** (0.0)	**59.6** (0.0)	**54.4** (0.0)	**64.5** (0.0)
Forward KL	60.8 (↓9.0)	48.8 (↓5.5)	57.3 (↓4.5)	73.5 (↓4.1)	70.3 (↓3.9)	54.5 (↓5.1)	49.1 (↓5.3)	59.2 (↓5.3)
Reverse KL	66.7 (↓3.1)	49.7 (↓4.6)	57.1 (↓4.7)	75.9 (↓1.7)	72.9 (↓1.3)	57.2 (↓2.4)	52.4 (↓2.0)	61.7 (↓2.8)
Log-ratio	53.4 (↓16.4)	41.4 (↓12.9)	50.8 (↓11.0)	70.0 (↓7.6)	66.9 (↓7.3)	50.7 (↓8.9)	46.3 (↓8.1)	54.2 (↓10.3)



(a) Signal conflict rates

(b) Sensitivity to budget k and allocation concentration β

Figure 4: (a) Rates of two signal conflicts: response-level advantages versus criterion-level credits, and distillation gradients versus outcome-grounded gradients. (b) Effects of k and β on average scores in each domain.

Ablation of relevance measures. To examine how relevance estimation affects credit allocation, we compare JSD with alternative measures in Table 3. Replacing JSD with forward KL and reverse KL reduces the average score by 5.3% and 2.8%, respectively. Compared with these asymmetric, unbounded divergences, JSD treats both distributions symmetrically and limits extreme relevance scores, which offers a plausible explanation for its advantage. We further consider the signed log-probability ratio, which favors generated tokens whose probabilities increase under criterion conditioning. This substitution reduces the average score by 10.3%, showing that full-distribution changes provide more informative allocation signals than the directional probability shift of the generated token alone. These comparisons support JSD as an effective relevance measure for credit allocation.

Hyperparameter sensitivity. We study CoCA's sensitivity to two key hyperparameters: the criterion budget k and allocation concentration β. As shown in Figure 4b, even $k = 1$ improves average scores over uniform allocation ($k = 0$) by 4.4% in medicine and 1.7% in science. Beyond $k = 2$, gains are marginal: selecting all criteria adds only 0.5% and 0.3%, respectively. These results indicate that prioritizing criteria with the largest absolute credits captures most of the benefit of full allocation. Thus, $k = 2$ offers a practical balance between performance and computation, requiring at most two additional forward passes per rollout. For β, performance peaks at 5 in medicine and 3 in science. This pattern favors moderate concentration, suggesting that small values dilute updates to relevant tokens, while large values can neglect other tokens needed for criterion satisfaction.

Case study of criterion–token relevance. To qualitatively assess whether criterion-induced distribution shifts capture token-level relevance, Figure 5 presents a medical response under a criterion concerning when to seek medical attention. Higher relevance scores concentrate on the parts of the response that provide the corresponding recommendation and describe the conditions under which it applies, while the general care advice receives much lower scores. Notably, this contrast indicates that the relevance signal captures criterion-specific semantics rather than general topical relevance. Such precise localization enables CoCA to perform targeted credit assignment at the token level.

6 Related Work

Rubrics as rewards. Rubric-based RL evaluates open-ended responses against explicit criteria, typically aggregating the resulting judgments into a scalar reward (Gunjal et al., 2026; Viswanathan et al., 2025; Huang et al., 2025). To strengthen the learning signal, existing methods refine criteria as

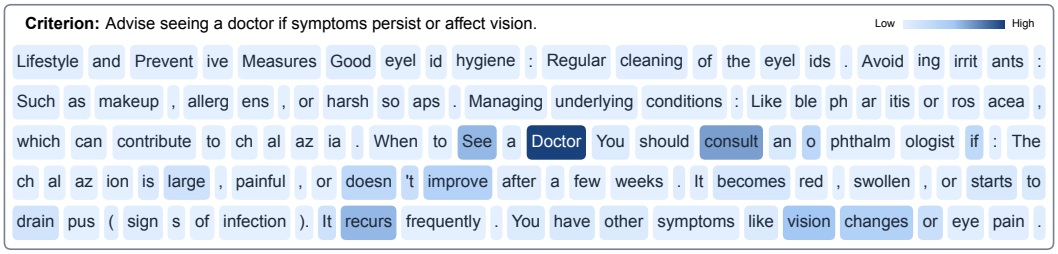


Figure 5: Visualization of criterion–token relevance in a medical response. Darker colors indicate higher JSD between next-token distributions with and without criterion conditioning.

the policy evolves (Shao et al., 2026; Li et al., 2026), adjust criterion weights (Tyagi et al., 2026), or alternate optimization across criterion categories (Lan et al., 2026). However, these methods retain response-level rewards. To provide finer-grained supervision, another line of work localizes criterion feedback to reasoning steps or token spans (Xu et al., 2026b; Xie et al., 2026; Gandhi et al., 2026; Jin et al., 2026). Their reliance on explicit attribution signals or separately trained relevance models adds supervision and computational overhead that complicates scaling. More fundamentally, these advances do not by themselves ensure that each criterion's signed contribution to the response-level advantage is preserved, motivating our focus on criterion-aware credit assignment.

Rubrics as guidance. Beyond defining rewards, rubrics also provide training-time guidance for policy learning. One line of work guides rollout generation with rubric instructions or feedback to promote exploration (Zhou et al., 2026; Bi et al., 2026; Zhang et al., 2026b; Hou et al., 2026). However, training on rubric-guided prefixes creates a mismatch with unguided inference, introducing exposure bias and a risk of compounding errors. To avoid this mismatch, another line adopts OPSD, which uses a rubric-informed teacher to provide token-level supervision on rollouts sampled from the unguided student (Gu et al., 2026; Rezaei et al., 2026; Bablani et al., 2026; Ye et al., 2026; Wang et al., 2026). Such distillation can serve as a standalone optimization objective or be combined with GRPO as an auxiliary objective. Although these methods provide dense supervision, they introduce a separate imitation objective that may conflict with reward-driven optimization. We therefore use privileged rubric information as a relevance cue for outcome-grounded credit allocation.

Credit assignment. In classical RL, temporal credit is assigned by estimating advantages from value functions (Schulman et al., 2016), redistributing delayed rewards (Arjona-Medina et al., 2019), or inferring action contributions from subsequent outcomes (Harutyunyan et al., 2019; Meulemans et al., 2023). For LLMs trained with RL on verifiable tasks, methods infer local rewards or advantages using implicit reward models (Cui et al., 2026) or shared rollout prefixes (Tran et al., 2026), while others concentrate policy updates on high-entropy tokens (Wang et al., 2025). Rubric-based RL poses an additional challenge: a response may satisfy some criteria while violating others, so redistributing an aggregate advantage alone cannot distinguish their opposing contributions. CoRT derives positive token weights through counterfactual rubric replay (Zhang et al., 2026a), but applies them to a single response-level advantage, tying all token advantages to its sign. Hence, the key is to preserve each criterion's signed contribution during token-level credit assignment.

7 Conclusion

In this work, we propose CoCA, a criterion-aware credit assignment framework for rubric-based RL. Our core insight is that effective learning from rubrics requires preserving the distinct contributions of individual criteria and assigning their credit to relevant parts of a response. CoCA realizes this principle by deriving signed criterion credits from observed outcomes and using privileged rubric information to guide their allocation across tokens. This translates criterion-level feedback into fine-grained, outcome-grounded policy updates. Experiments across medicine and science show CoCA outperforms standard rubric-based RL by 4.2%, with higher correction and lower regression rates. We hope our work encourages greater attention to credit assignment as a bridge between structured evaluation and policy optimization, paving the way toward more effective RL for open-ended tasks.

AI USE STATEMENT

We used generative AI for literature retrieval and discovery, coding assistance, and improving the grammar, clarity, and readability of author-written text. In our experimental pipeline, gpt-oss-120b and DeepSeek-V4.1-Flash served as rubric-based judges, while GPT-6 Astra generated word-level annotations for the criterion–token relevance analysis. These experimental uses and their validation procedures are described in the appendix. The authors reviewed all AI-assisted work and take full responsibility for the content of the paper.

ETHICS STATEMENT

This work develops and evaluates a criterion-aware reinforcement learning method using only publicly available datasets and benchmarks in medical and scientific QA. The medical benchmarks serve as research testbeds for assessing open-ended response quality and generalization. Accordingly, our results should not be interpreted as evidence of clinical safety or suitability for clinical use.

REPRODUCIBILITY STATEMENT

Complete proofs of the theoretical results are provided in Appendix A. We document the datasets and data splits, evaluation protocols, prompts, training and implementation settings, and baseline reproduction details in Appendix B. The source code, configuration files, and evaluation scripts are included in the anonymous supplementary material.

REFERENCES

Jose A. Arjona-Medina, Michael Gillhofer, Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, and Sepp Hochreiter. RUDDER: Return decomposition for delayed rewards. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

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.

Deepika Bablani, Ajay Gupta, and Wanming Chen. Rubrics as privileged information for open-ended generation. *arXiv preprint arXiv:2608.02948*, 2026.

Baolong Bi, Shenghua Liu, Yiwei Wang, Siqian Tong, Lingrui Mei, Yuyao Ge, Yilong Xu, Jiafeng Guo, and Xueqi Cheng. Reward and guidance through rubrics: Promoting exploration to improve multi-domain reasoning. In *Forty-third International Conference on Machine Learning*, 2026.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Yuchen Zhang, Jiacheng Chen, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, Zhiyuan Liu, Maosong Sun, Bowen Zhou, and Ning Ding. Process reinforcement through implicit rewards. *Transactions on Machine Learning Research*, 2026.

Shubham Gandhi, Saurabh Goyal, Kiran Kate, and Yara Rizk. DRACO: Fine-grained credit assignment with dynamic rubrics for long-horizon agent training. *arXiv preprint arXiv:2609.04094*, 2026.

Siyi Gu, Jialin Chen, Sophia Zhou, Arman Cohan, and Rex Ying. Rethinking reward supervision: Rubric-conditioned self-distillation. *arXiv preprint arXiv:2606.19327*, 2026.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. In *International Conference on Learning Representations*, pp. 127924–127945, 2026.

Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Gheshlaghi Azar, Bilal Piot, Nicolas Heess, Hado van Hasselt, Gregory Wayne, Satinder Singh, Doina Precup, and Remi Munos. Hindsight credit assignment. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. LLM-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13806–13834. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.745.

Jinkun Hou, Zhuo Liu, Huimin Ren, Hongsheng Xin, Pan Zhou, and Kun Zhan. RISE-RL: Rubric-informed selective exploration for open-ended reinforcement learning. *arXiv preprint arXiv:2608.09123*, 2026.

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 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.

Ruipeng Jia, Yunyi Yang, Yongbo Gai, Kai Luo, Shihao Huang, Jianhe Lin, Xiaoxi Jiang, and Guanjun Jiang. Writing-Zero: Bridge the gap between non-verifiable tasks and verifiable rewards. *arXiv preprint arXiv:2506.00103*, 2025.

Yuxin Jiang, Yufei Wang, Qiyuan Zhang, Xingshan Zeng, Liangyou Li, Jierun Chen, Chaofan Tao, Haoli Bai, and Lifeng Shang. From verifiable dot to reward chain: Harnessing verifiable reference-based rewards for reinforcement learning of open-ended generation. In *International Conference on Learning Representations*, pp. 16575–16593, 2026.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021. doi: 10.3390/app11146421.

Rui Jin, Jikai Chen, Yihan Chen, Hao Zhou, Demin Zhu, Kaichen Yang, Dong Wang, Linjian Mo, and Chenyi Zhuang. Rubric-to-code credit assignment for reinforcement learning. *arXiv preprint arXiv:2608.27906*, 2026.

Guangchen Lan, Lian Xiong, Xin Zhou, Hejie Cui, Yuwei Zhang, Mao Li, Zhenyu Shi, Besnik Fetahu, Lihong Li, and Xian Li. Alternating reinforcement learning with contextual rubric rewards: Beyond the scalarization strategy. *arXiv preprint arXiv:2603.15646*, 2026.

Siyuan Li, Xinxin Song, Ruinian Chen, Jingjing Fan, Tingxiong Xiao, Yangen Hu, Ke Zeng, and Jinli Suo. CARE: Contrastive anchor-based rubric evolution for large language model post-training. *arXiv preprint arXiv:2609.00892*, 2026.

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. Association for Computational Linguistics, 2026a. doi: 10.18653/v1/2026.acl-long.791.

Wenhan Liu, Jiajie Jin, Zhaoheng Huang, Tongyu Wen, Guanting Dong, Ziliang Zhao, Yutao Zhu, Zhicheng Dou, and Ji-Rong Wen. The rules of the game: A survey of rubrics for large language models, 2026b.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective. In *Conference on Language Modeling (COLM)*, 2025.

Alexander Meulemans, Simon Schug, Seijin Kobayashi, Nathaniel Daw, and Gregory Wayne. Would i have gotten that reward? long-term credit assignment by counterfactual contribution analysis. In *Advances in Neural Information Processing Systems*, volume 36, pp. 68685–68735. Curran Associates, Inc., 2023. doi: 10.52202/075280-3006.

Team Olmo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, et al. Olmo 3. *arXiv preprint arXiv:2512.13961*, 2025.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.

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*, 2026.

John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations*, 2016.

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. Reinforcement learning with evolving rubrics for deep research. In *Forty-third International Conference on Machine Learning*, 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

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*, pp. 1279–1297. Association for Computing Machinery, 2025. doi: 10.1145/3689031.3696075.

Hieu Tran, Zonghai Yao, and Hong Yu. Exploiting tree structure for credit assignment in reinforcement learning with large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 10795–10810. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.524.

Utkarsh Tyagi, Xingang Guo, MohammadHossein Rezaei, Daniel George, Anas Mahmoud, Jackson Lee, Bing Liu, and Yunzhong He. Not every rubric teaches equally: Policy-aware rubric rewards for RLVR. *arXiv preprint arXiv:2605.20164*, 2026.

Vijay Viswanathan, Yanchao Sun, Xiang Kong, Meng Cao, Graham Neubig, and Sherry Wu. Checklists are better than reward models for aligning language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 114728–114754. Curran Associates, Inc., 2025. doi: 10.52202/085713-3824.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xiong-Hui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 115452–115486. Curran Associates, Inc., 2025. doi: 10.52202/085713-3850.

Yuhan Wang, Zhengxi Lu, Yuchen Yan, Kaitao Song, Wenqi Zhang, Weiming Lu, Jun Xiao, Yueting Zhuang, and Yongliang Shen. PaperGym: Rubric-centered evolution for research-plan generation. *arXiv preprint arXiv:2608.31119*, 2026.

Weichu Xie, Haozhe Zhao, Wenpu Liu, Yongfu Zhu, Liang Chen, Minghao Ye, Zirong Chen, Yuqi Xu, Shuai Dong, Ziyue Wang, et al. Step-wise rubric rewards for LLM reasoning. *arXiv preprint arXiv:2605.17291*, 2026.

Anyi Xu, B Li, Bangcai Lin, Bing Xue, BingCheng Xian, Bingzheng Xu, Bochao Wu, Bowei Zhang, Boyi Deng, CC Yu, et al. Deepseek-v4. 1-flash: Pushing the limits of kv cache compression. *arXiv preprint arXiv:2609.19969*, 2026a.

Tianze Xu, Yanzhao Zheng, Pengrui Lu, Lyumanshan Ye, Yong Wu, Zhentao Zhang, Yuanqiang Yu, Chao Ma, Jihuai Zhu, Pengfei Liu, et al. Rubrics to tokens: Bridging response-level rubrics and token-level rewards in instruction following tasks. *arXiv preprint arXiv:2604.02795*, 2026b.

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.

Tianzhu Ye, Li Dong, Guanheng Chen, He Zhu, Xun Wu, Shaohan Huang, and Furu Wei. LLM-as-a-coach: Experiential learning for non-verifiable tasks. *arXiv preprint arXiv:2607.18110*, 2026.

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*, 14:1365–1389, 2026. doi: 10.1162/tacl.a.732.

Fangyi Yu, Nabeel Seedat, Dasha Herrmannova, Frank Schilder, and Jonathan Richard Schwarz. Beyond pointwise scores: Decomposed criteria-based evaluation of LLM responses. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1931–1954. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-industry.136.

Bo-Wen Zhang, Junwei He, Wen Wang, Song-Lin Lv, Wentao Ma, Rongyi Lin, Shuhan Zhong, and Lan-Zhe Guo. CoRT: Counterfactual replay for token-level rubric-guided policy optimization. *arXiv preprint arXiv:2607.25659*, 2026a.

Kaiyan Zhang, Yuxin Zuo, Bingxiang He, Youbang Sun, Runze Liu, Che Jiang, Yuchen Fan, Kai Tian, Guoli Jia, Pengfei Li, et al. A survey of reinforcement learning for large reasoning models. *arXiv preprint arXiv:2509.08827*, 2025a.

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 *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 4888–4914. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.findings-emnlp.263.

Wenjian Zhang, Kongcheng Zhang, Jiaxin Qi, Jianqiang Huang, and Baisheng Lai. Experience is the best teacher: Motivating effective exploration in reinforcement learning for LLMs. In *Forty-third International Conference on Machine Learning*, 2026b.

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.

Yilun Zhao, Kaiyan Zhang, Tiansheng Hu, Sihong Wu, Ronan Le Bras, Yixin Liu, Robert Tang, Joseph Chee Chang, Jesse Dodge, Jonathan Bragg, Chen Zhao, Hanna Hajishirzi, Doug Downey, and Arman Cohan. SciArena: An open evaluation platform for non-verifiable scientific literature-grounded tasks. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-3532.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623. Curran Associates, Inc., 2023. doi: 10.52202/075280-2020.

Yang Zhou, Sunzhu Li, Shunyu Liu, Wenkai Fang, Kongcheng Zhang, Jiale Zhao, Jingwen Yang, Yihe Zhou, Jianwei Lv, Tongya Zheng, Hengtong Lu, Wei Chen, Yan Xie, 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.

A Theoretical Proofs

A.1 Proof of Proposition 1

Proposition 1 (Credit resolution collapse in rubric-based RL). *For any oracle resolution $\mathbf{C}_i^* \in \mathcal{C}_i(\mathbf{a}_i)$, the token credit induced by rubric-based RL and the corresponding projection error satisfy*

$$\left(\mathbf{1}_M^\top \mathbf{a}_i\right) \mathbf{1}_{T_i} = A_i \mathbf{1}_{T_i} = \boldsymbol{\alpha}_i(\Pi_{\mathcal{U}_i}(\mathbf{C}_i^*)), \quad \tfrac{1}{T_i} \left\|\mathbf{C}_i^* - \Pi_{\mathcal{U}_i}(\mathbf{C}_i^*)\right\|_F^2 = \textstyle\sum_{j=1}^M \mathrm{Var}_t\left([\mathbf{C}_i^*]_{j,t}\right). \quad (6)$$

Moreover, the linear map $\mathbf{a} \mapsto (\mathbf{1}_M^\top \mathbf{a})\mathbf{1}_{T_i}$ has rank one and kernel $\mathbf{1}_M^\perp$. Consequently, any nonzero criterion budget vector $\mathbf{a}_i \in \mathbf{1}_M^\perp$ yields a zero token-credit vector, while any position t satisfying $A_i[\boldsymbol{\alpha}_i(\mathbf{C}_i^*)]_t < 0$ is assigned credit whose sign is opposite to its oracle token credit.

Proof. Fix a rollout i and an oracle resolution $\mathbf{C}_i^* \in \mathcal{C}_i(\mathbf{a}_i)$. Recall that $\mathcal{U}_i = \{\mathbf{u}\mathbf{1}_{T_i}^\top : \mathbf{u} \in \mathbb{R}^M\}$ consists of matrices whose rows are constant across tokens. For any $\mathbf{C} \in \mathbb{R}^{M \times T_i}$, write $\mu_j = T_i^{-1} \sum_t C_{j,t}$ for its j-th row mean. Centering each row gives

$$\begin{aligned}\|\mathbf{C} - \mathbf{u}\mathbf{1}_{T_i}^\top\|_F^2 &= \sum_{j=1}^M \sum_{t=1}^{T_i} \big((C_{j,t} - \mu_j) + (\mu_j - u_j)\big)^2 \\ &= \sum_{j=1}^M \sum_{t=1}^{T_i} (C_{j,t} - \mu_j)^2 + T_i \sum_{j=1}^M (u_j - \mu_j)^2.\end{aligned}$$

The cross terms vanish because $\sum_t (C_{j,t} - \mu_j) = 0$ for every row. The first term is independent of $\mathbf{u}$, while the second is uniquely minimized at $u_j = \mu_j$. Thus, projection replaces each row by its mean. Since $T_i^{-1}\mathbf{C}_i^* \mathbf{1}_{T_i} = \mathbf{a}_i$ by the budget constraint,

$$\Pi_{\mathcal{U}_i}(\mathbf{C}) = \left(\frac{1}{T_i}\mathbf{C}\mathbf{1}_{T_i}\right)\mathbf{1}_{T_i}^\top, \qquad \Pi_{\mathcal{U}_i}(\mathbf{C}_i^*) = \mathbf{a}_i \mathbf{1}_{T_i}^\top.$$

Using $\boldsymbol{\alpha}_i(\mathbf{C}) = \mathbf{C}^\top \mathbf{1}_M$ and $\mathbf{1}_M^\top \mathbf{a}_i = A_i$, we obtain

$$\boldsymbol{\alpha}_i(\Pi_{\mathcal{U}_i}(\mathbf{C}_i^*)) = (\mathbf{a}_i \mathbf{1}_{T_i}^\top)^\top \mathbf{1}_M = (\mathbf{1}_M^\top \mathbf{a}_i)\mathbf{1}_{T_i} = A_i \mathbf{1}_{T_i}.$$

This is the token-credit vector used by rubric-based RL, regardless of the oracle's allocation within each row. Moreover, because the j-th row of $\mathbf{C}_i^*$ has mean A_{ij},

$$\frac{1}{T_i}\left\|\mathbf{C}_i^* - \Pi_{\mathcal{U}_i}(\mathbf{C}_i^*)\right\|_F^2 = \sum_{j=1}^M \frac{1}{T_i} \sum_{t=1}^{T_i} \big([\mathbf{C}_i^*]_{j,t} - A_{ij}\big)^2 = \sum_{j=1}^M \mathrm{Var}_t([\mathbf{C}_i^*]_{j,t}),$$

where Var_t uses the uniform distribution over the T_i token positions. Thus, the projection discards exactly the within-criterion variation measured by this identity.

Finally, the budget-to-token map is represented by $\mathbf{L}_i = \mathbf{1}_{T_i}\mathbf{1}_M^\top$. Both factors in this outer product are nonzero, so $\mathrm{rank}(\mathbf{L}_i) = 1$. Furthermore,

$$\mathbf{L}_i \mathbf{a} = 0 \quad \Longleftrightarrow \quad \mathbf{1}_M^\top \mathbf{a} = 0, \qquad \ker \mathbf{L}_i = \mathbf{1}_M^\perp.$$

Every nonzero vector in this kernel gives zero token credit despite its nonzero criterion contributions. Since rubric-based RL assigns A_i at every position, the condition $A_i[\boldsymbol{\alpha}_i(\mathbf{C}_i^*)]_t < 0$ means that the assigned and oracle token credits are nonzero and have opposite signs. □

A.2 Proof of Proposition 2

Proposition 2 (OPSD matching can reduce reward). *Consider the next-token decision at $h_{i,t}$ as a one-step contextual bandit with finite action space $\mathcal{V}$, a reward function $R : \mathcal{V} \to \mathbb{R}$, and a tabular softmax policy. Let θ' be obtained by a natural-gradient descent step of size $\eta > 0$ on the KL term of Eq. (2), using the local Fisher information and holding the self-teacher fixed. Then, as $\eta \to 0$,*

$$J(\theta'; h_{i,t}) - J(\theta; h_{i,t}) = \eta \, \mathrm{Cov}_{a \sim \pi_\theta(\cdot|h_{i,t})}\left(R(a), r_{i,t}(a; \xi)\right) + \mathcal{O}(\eta^2). \quad (7)$$

When $|\mathcal{V}| \geq 3$, there exist instances for which $J(\theta; h_{i,t}(\xi)) > J(\theta; h_{i,t})$ while the covariance in Eq. (7) is strictly negative, implying $J(\theta'; h_{i,t}) < J(\theta; h_{i,t})$ for sufficiently small $\eta > 0$.

Proof. We first derive the local natural-gradient update and its first-order effect on reward. We then construct a higher-reward teacher for which this effect is negative. Throughout the proof, the action space is finite and all logarithms are natural.

Local matching update. At the fixed context $h = h_{i,t}$, write p for the student distribution, q for the detached teacher distribution, and $\mathbf{r}$ for their matching signal:

$$p_a = \pi_\theta(a \mid h), \qquad q_a = \pi_\theta(a \mid h_{i,t}(\xi)), \qquad r_a = \log \frac{q_a}{p_a}.$$

These quantities are evaluated before the update, and q is held fixed when differentiating the matching objective. Finite softmax logits ensure $p_a, q_a > 0$, so every r_a is finite. Let $n = |\mathcal{V}|$ and let $\boldsymbol{\vartheta} \in \mathbb{R}^n$ denote the student logits at this fixed context. The local reverse-KL loss is

$$\ell(\boldsymbol{\vartheta}) = D_{\mathrm{KL}}(p\|q) = \sum_{a\in\mathcal{V}} p_a \log \frac{p_a}{q_a}, \qquad p_a = \frac{e^{\vartheta_a}}{\sum_{b\in\mathcal{V}} e^{\vartheta_b}}.$$

Writing $\mathbf{e}_a$ for the coordinate vector of action a, softmax differentiation gives

$$\nabla_{\boldsymbol{\vartheta}} \log p_a = \mathbf{e}_a - p, \qquad \frac{\partial p_a}{\partial \vartheta_b} = p_a\big(\mathbf{1}\{a = b\} - p_b\big).$$

Consequently, the local Fisher matrix equals the softmax Jacobian:

$$F = \sum_a p_a(\mathbf{e}_a - p)(\mathbf{e}_a - p)^\top = \mathrm{diag}(p) - pp^\top.$$

Differentiating the loss through the student probabilities yields

$$\frac{\partial \ell}{\partial \vartheta_b} = \sum_a \frac{\partial p_a}{\partial \vartheta_b}\left(\log \frac{p_a}{q_a} + 1\right),$$
$$\nabla_{\boldsymbol{\vartheta}} \ell = F(\log p - \log q + \mathbf{1}_n) = -F\mathbf{r},$$

Here $\mathbf{r} = (r_a)_{a\in\mathcal{V}}$, vector logarithms are componentwise, and $F\mathbf{1}_n = 0$ gives the last equality.

The Fisher matrix is singular because softmax is invariant to a common shift of all logits. Its nullspace can be identified from the following quadratic form, valid for every $\mathbf{v} \in \mathbb{R}^n$:

$$\mathbf{v}^\top F \mathbf{v} = \sum_a p_a v_a^2 - \left(\sum_a p_a v_a\right)^2 = \mathrm{Var}_{a\sim p}(v_a).$$

Since p has full support, this variance vanishes exactly when every v_a has the same value. Thus $\ker F = \mathrm{span}(\mathbf{1}_n)$, and its Moore–Penrose pseudoinverse satisfies

$$F^\dagger F = I_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top.$$

The natural-gradient descent step, with F evaluated at the original student policy, is therefore

$$\boldsymbol{\vartheta}' = \boldsymbol{\vartheta} - \eta F^\dagger \nabla_{\boldsymbol{\vartheta}} \ell = \boldsymbol{\vartheta} + \eta F^\dagger F \mathbf{r} = \boldsymbol{\vartheta} + \eta(\mathbf{r} - \kappa \mathbf{1}_n), \qquad \kappa = \frac{1}{n}\sum_a r_a.$$

The common shift $-\eta\kappa$ cancels in softmax, giving the exact probability update

$$p'_a = \frac{e^{\vartheta_a + \eta(r_a - \kappa)}}{\sum_b e^{\vartheta_b + \eta(r_b - \kappa)}} = \frac{p_a e^{\eta r_a}}{Z(\eta)}, \qquad Z(\eta) = \sum_b p_b e^{\eta r_b}.$$

First-order changes in reward and matching loss. Let $\bar{r} = \sum_a p_a r_a$ be the student-weighted mean of the matching signal. This differs from the unweighted mean κ used only to fix the logit shift above. Expanding the normalizing factor and its reciprocal at $\eta = 0$ gives

$$Z(\eta) = 1 + \eta\bar{r} + \mathcal{O}(\eta^2), \qquad Z(\eta)^{-1} = 1 - \eta\bar{r} + \mathcal{O}(\eta^2).$$

Combining these expansions with $e^{\eta r_a} = 1 + \eta r_a + \mathcal{O}(\eta^2)$ yields

$$p'_a = p_a\big(1 + \eta r_a + \mathcal{O}(\eta^2)\big)\big(1 - \eta\bar{r} + \mathcal{O}(\eta^2)\big) = p_a + \eta p_a(r_a - \bar{r}) + \mathcal{O}(\eta^2).$$

For fixed p, q, every log-ratio is finite and $Z(0) = 1$, so the updated probabilities are smooth near $\eta = 0$. The finite action space allows a common bound on their second-order remainders. Since $\mathcal{V}$ is finite and R is real-valued, $\max_{a\in\mathcal{V}} |R(a)| < \infty$. Multiplying the probability expansion by $R(a)$ and summing therefore gives

$$\begin{aligned} J(\theta'; h) - J(\theta; h) &= \sum_a R(a)(p'_a - p_a) \\ &= \eta \left(\sum_a p_a R(a) r_a - \left(\sum_a p_a R(a) \right) \left(\sum_a p_a r_a \right) \right) + \mathcal{O}(\eta^2) \\ &= \eta \operatorname{Cov}_{a\sim p}(R(a), r_a) + \mathcal{O}(\eta^2). \end{aligned}$$

Restoring $h = h_{i,t}$ and $r_a = r_{i,t}(a; \xi)$ proves the first assertion.

The same step also decreases the local matching loss for sufficiently small positive η whenever $\mathbf{r}$ is nonconstant. Indeed, a Taylor expansion of the loss along the update direction gives

$$\begin{aligned} \ell(\boldsymbol{\vartheta}') - \ell(\boldsymbol{\vartheta}) &= -\eta (\nabla_{\boldsymbol{\vartheta}} \ell)^\top F^\dagger \nabla_{\boldsymbol{\vartheta}} \ell + \mathcal{O}(\eta^2) \\ &= -\eta \mathbf{r}^\top F F^\dagger F \mathbf{r} + \mathcal{O}(\eta^2) = -\eta \operatorname{Var}_{a\sim p}(r_a) + \mathcal{O}(\eta^2), \end{aligned}$$

using $F F^\dagger F = F$. A negative reward covariance is possible only when $\mathbf{r}$ is nonconstant. Thus, in that case, small matching steps decrease both the distillation loss and expected reward.

A higher-reward teacher with a harmful matching direction. Consider three actions with rewards R, student probabilities p, and teacher probabilities q, listed in the same order:

$$R = \left(1, \frac{1}{2}, 0\right), \qquad p = \left(\frac{1}{20}, \frac{9}{20}, \frac{1}{2}\right), \qquad q = \left(\frac{3}{10}, \frac{1}{20}, \frac{13}{20}\right).$$

Both distributions have full support and sum to one. Their expected rewards are

$$\mathbb{E}_p[R] = \frac{1}{20} + \frac{9}{20} \cdot \frac{1}{2} = \frac{11}{40}, \qquad \mathbb{E}_q[R] = \frac{3}{10} + \frac{1}{20} \cdot \frac{1}{2} = \frac{13}{40}.$$

The teacher therefore has strictly higher expected reward. Nevertheless, its matching signal $\mathbf{r} = (\log 6, -\log 9, \log(13/10))$ has negative covariance with reward under the student:

$$\begin{aligned} \operatorname{Cov}_p(R, r) &= \sum_{a=1}^{3} p_a \left(R(a) - \frac{11}{40} \right) r_a \\ &= \frac{1}{20} \cdot \frac{29}{40} \log 6 - \frac{9}{20} \cdot \frac{9}{40} \log 9 - \frac{1}{2} \cdot \frac{11}{40} \log \frac{13}{10} \\ &= \frac{29 \log 6 - 81 \log 9 - 110 \log(13/10)}{800} < 0. \end{aligned}$$

The sign is strict because $\log 9 > \log 6 > 0$ and $\log(13/10) > 0$. This construction is realizable by a tabular softmax policy: choose the logits to be $\log p$ at the ordinary context and $\log q$ at the privileged context. All logits are finite, and the two context-specific tables can be chosen independently.

With $c = -\operatorname{Cov}_p(R, r) > 0$, the reward expansion becomes

$$J(\theta'; h) - J(\theta; h) = -c\eta + \mathcal{O}(\eta^2).$$

For all sufficiently small positive η, the remainder has magnitude at most $c\eta/2$. The reward change is then at most $-c\eta/2 < 0$, even though the teacher has higher expected reward. The matching-loss expansion also ensures that the same step strictly decreases the distillation objective.

Finally, extend the construction to any $n > 3$ by keeping the first two actions and replacing the third by $n - 2$ zero-reward actions. For each $a = 3, \ldots, n$, set

$$R(a) = 0, \qquad p_a = \frac{1}{2(n-2)}, \qquad q_a = \frac{13}{20(n-2)}.$$

These actions retain the original third action's total student and teacher probabilities, and each has the same log-ratio $\log(13/10)$. Their combined contribution to the reward–signal covariance is therefore unchanged. Both expected rewards are also unchanged, and full support is preserved. The counterexample consequently holds for every $|\mathcal{V}| \geq 3$. □

A.3 Proof of Proposition 3

Proposition 3 (Grounding-preserving resolution improvement). *For any $\beta \geq 0$, CoCA preserves each criterion's budget and the sign of its token-level contributions, with $\widehat{\mathbf{C}}_i(\beta) \in \mathcal{C}_i(\mathbf{a}_i)$. For each selected criterion $j \in \mathcal{S}_i$ with $\Delta_{ij} > 0$, the following holds whenever $0 < \beta < 8\Delta_{ij}/b_{ij}^2$:*

$$\frac{1}{T_i} \sum_{t \notin \mathcal{T}_{ij}} \left| [\widehat{\mathbf{C}}_i(\beta)]_{j,t} \right| < |A_{ij}| \left(1 - \frac{|\mathcal{T}_{ij}|}{T_i}\right). \tag{11}$$

The right-hand side gives the same quantity under uniform allocation.

Proof. Fix a rollout i and hold its criterion budgets, selected set, relevance scores, and relevant token sets fixed as β varies. We first prove budget and sign preservation, then compare the credit magnitude assigned to irrelevant positions with its value under uniform allocation.

Budget and sign preservation. Write $\omega_{ij,t}(\beta) = [\boldsymbol{\omega}_{ij}(\beta)]_t$. For each selected criterion $j \in \mathcal{S}_i$, positivity of the exponential and normalization of the weights give

$$\omega_{ij,t}(\beta) = \frac{T_i e^{\beta d_{ij,t}}}{\sum_{u=1}^{T_i} e^{\beta d_{ij,u}}} > 0, \qquad \frac{1}{T_i} \sum_{t=1}^{T_i} \omega_{ij,t}(\beta) = 1.$$

For $j \notin \mathcal{S}_i$, the weights equal one, so positivity and normalization hold for every criterion. Multiplication by the fixed budget then gives the following row-mean identity:

$$\frac{1}{T_i} \sum_{t=1}^{T_i} [\widehat{\mathbf{C}}_i(\beta)]_{j,t} = A_{ij} \frac{1}{T_i} \sum_{t=1}^{T_i} \omega_{ij,t}(\beta) = A_{ij}.$$

These row-mean identities show $\widehat{\mathbf{C}}_i(\beta) \in \mathcal{C}_i(\mathbf{a}_i)$. Positive weights ensure that each nonzero row has its budget's sign, while zero budgets yield zero rows. Taking absolute values rowwise and summing the signed credits over criteria gives

$$\frac{1}{T_i} \sum_{t=1}^{T_i} \left| [\widehat{\mathbf{C}}_i(\beta)]_{j,t} \right| = |A_{ij}|, \qquad \frac{1}{T_i} \sum_{t=1}^{T_i} \widehat{A}_{i,t} = \sum_{j=1}^{M} A_{ij} = A_i.$$

Thus, redistribution also preserves each row's total credit magnitude and the response-level mean advantage. At $\beta = 0$, all weights equal one, recovering uniform allocation.

Reduction to a comparison of exponential averages. Fix a selected criterion $j \in \mathcal{S}_i$ with $\Delta_{ij} > 0$. Because selection is restricted to nonzero budgets, $A_{ij} \neq 0$. For readability, write $T = T_i$, $A = A_{ij}$, $d_t = d_{ij,t}$, $\Delta = \Delta_{ij}$, and $b = b_{ij}$. Let $\mathcal{T} = \mathcal{T}_{ij}$ and $\mathcal{N} = \{1, \ldots, T\} \setminus \mathcal{T}$. The assumption $\varnothing \neq \mathcal{T} \subsetneq \{1, \ldots, T\}$ ensures $m = |\mathcal{T}| > 0$, $n = |\mathcal{N}| > 0$, and $m + n = T$. The relevant set is specified without using the scores, and its mean-score advantage is

$$\mu_{\mathcal{T}} = \frac{1}{m} \sum_{t \in \mathcal{T}} d_t, \qquad \mu_{\mathcal{N}} = \frac{1}{n} \sum_{t \in \mathcal{N}} d_t, \qquad \Delta = \mu_{\mathcal{T}} - \mu_{\mathcal{N}} > 0.$$

Define the mean exponential scores on the two sets as

$$U(\beta) = \frac{1}{m} \sum_{t \in \mathcal{T}} e^{\beta d_t}, \qquad V(\beta) = \frac{1}{n} \sum_{t \in \mathcal{N}} e^{\beta d_t}.$$

Substituting the normalized allocation weights gives the off-target credit magnitude

$$L(\beta) := \frac{1}{T} \sum_{t \in \mathcal{N}} \left| [\widehat{\mathbf{C}}_i(\beta)]_{j,t} \right| = |A| \frac{\sum_{t \in \mathcal{N}} e^{\beta d_t}}{\sum_{u=1}^{T} e^{\beta d_u}} = |A| \frac{nV(\beta)}{mU(\beta) + nV(\beta)}.$$

Under uniform allocation, $U(0) = V(0) = 1$, so $L(0) = |A|n/T$. As $|A|$, m, n, and both exponential averages are positive, $L(\beta) < L(0)$ is equivalent to $U(\beta) > V(\beta)$. It remains to show that the positive mean-score gap guarantees this ordering under the stated restriction on β.

Controlling the effect of exponentiation. For the relevant positions, Jensen's inequality gives

$$U(\beta) = \frac{1}{m} \sum_{t \in \mathcal{T}} e^{\beta d_t} \geq \exp\left(\frac{\beta}{m} \sum_{t \in \mathcal{T}} d_t \right) = e^{\beta \mu_{\mathcal{T}}}.$$

For the irrelevant positions, we also need to control the increase in the exponential average caused by variation in the scores. For $s \geq 0$, define

$$\phi(s) = \log\left(\frac{1}{n}\sum_{t\in\mathcal{N}} e^{sd_t}\right), \qquad \nu_s(t) = \frac{e^{sd_t}}{\sum_{u\in\mathcal{N}} e^{sd_u}}.$$

Here ν_s is a probability distribution over the fixed set $\mathcal{N}$. Differentiating the finite sum twice gives

$$\phi'(s) = \frac{\sum_{t\in\mathcal{N}} d_t e^{sd_t}}{\sum_{t\in\mathcal{N}} e^{sd_t}} = \mathbb{E}_{t\sim\nu_s}[d_t],$$

$$\phi''(s) = \frac{\sum_{t\in\mathcal{N}} d_t^2 e^{sd_t}}{\sum_{t\in\mathcal{N}} e^{sd_t}} - \left(\frac{\sum_{t\in\mathcal{N}} d_t e^{sd_t}}{\sum_{t\in\mathcal{N}} e^{sd_t}}\right)^2 = \mathrm{Var}_{t\sim\nu_s}(d_t).$$

To bound this variance uniformly in s, let $d_{\min} = \min_t d_t$, $d_{\max} = \max_t d_t$, and $c = (d_{\min} + d_{\max})/2$. The score range is $b = d_{\max} - d_{\min}$, so $|d_t - c| \leq b/2$ for every position. Therefore,

$$\phi''(s) = \mathbb{E}_{\nu_s}[(d_t - c)^2] - \big(\mathbb{E}_{\nu_s}[d_t] - c\big)^2 \leq \mathbb{E}_{\nu_s}[(d_t - c)^2] \leq \frac{b^2}{4}.$$

At $s = 0$, the distribution ν_0 is uniform on $\mathcal{N}$, giving $\phi(0) = 0$ and $\phi'(0) = \mu_{\mathcal{N}}$. Applying Taylor's formula with integral remainder and the uniform variance bound yields

$$\begin{aligned}\log V(\beta) = \phi(\beta) &= \beta\mu_{\mathcal{N}} + \int_0^\beta (\beta - s)\phi''(s)\,\mathrm{d}s \\ &\leq \beta\mu_{\mathcal{N}} + \frac{b^2}{4}\int_0^\beta (\beta - s)\,\mathrm{d}s = \beta\mu_{\mathcal{N}} + \frac{\beta^2 b^2}{8}.\end{aligned}$$

Strict improvement over uniform allocation. Combining the upper bound on $\log V(\beta)$ with the lower bound $\log U(\beta) \geq \beta\mu_{\mathcal{T}}$ yields

$$\log\frac{U(\beta)}{V(\beta)} \geq \beta(\mu_{\mathcal{T}} - \mu_{\mathcal{N}}) - \frac{\beta^2 b^2}{8} = \beta\left(\Delta - \frac{\beta b^2}{8}\right).$$

The condition $\Delta > 0$ implies $b > 0$: otherwise all scores would coincide and the mean gap would be zero. For $0 < \beta < 8\Delta/b^2$, the right-hand side is strictly positive, and hence $U(\beta) > V(\beta)$. This bound ensures that the mean-score advantage survives exponentiation.

Finally, substituting $U(\beta) > V(\beta)$ into the off-target magnitude gives

$$L(\beta) = |A|\frac{nV(\beta)}{mU(\beta) + nV(\beta)} < |A|\frac{nV(\beta)}{mV(\beta) + nV(\beta)} = |A|\frac{n}{T} = |A|\left(1 - \frac{m}{T}\right).$$

Strictness uses $|A| > 0$, $m, n > 0$, and $V(\beta) > 0$. Restoring the indices proves the claimed reduction relative to uniform allocation. Because the comparison uses absolute credit while every row retains its budget sign, it applies equally to positive and negative criterion budgets. □

B Additional Experimental Details

B.1 Datasets, Benchmarks, and Evaluation

Training datasets. We focus on medical and scientific QA tasks, training a separate policy for each domain on RaR-Medicine and RaR-Science (Gunjal et al., 2026). Both datasets pair reasoning questions with instance-specific rubrics that identify essential, important, optional, and pitfall criteria. The medical data cover diagnosis, treatment, and related medical reasoning; the science data cover scientific knowledge and multi-step problem solving. We retain the criterion text and signed point values in the prepared data. The training splits contain 17,926 medical questions and 18,333 scientific questions, with an average of 7.46 and 7.51 criteria per question, respectively.

Table 4: Training and evaluation configuration.

Settings	Hyperparameters
CoCA	$k = 2$, $\beta = 5$ (medicine), $\beta = 3$ (science)
Sampling	rollout_n $= 8$, top_k $= -1$, top_p $= 1.0$, temperature $= 1.0$ max_prompt_length $= 2{,}048$, max_response_length $= 4{,}096$ enable_thinking $=$ False
Training	ppo_mini_batch_size $= 32$, ppo_micro_batch_size_per_gpu $= 2$ log_prob_micro_batch_size_per_gpu $= 16$, train_batch_size $= 64$ AdamW, learning_rate $= 10^{-6}$, weight_decay $= 0.01$ schedule $=$ constant, warmup ratio $= 0$, grad_clip $= 1.0$ ppo_epochs $= 1$, clip_ratio $= 0.2$, training_steps $= 400$ entropy_coeff $= 0$, use_kl_loss $=$ False, use_kl_in_reward $=$ False
Evaluation	top_p $= 0.95$, temperature $= 0.6$, max_response_length $= 4{,}096$
Optimizations	FSDP, vLLM (tensor parallel size $= 2$), FlashAttention-2 bf16 mixed precision, gradient checkpointing, padding removal Actor parameter and optimizer offload: disabled

Evaluation benchmarks. We evaluate each domain-specific policy on its corresponding RaR test split (2,242 questions for medicine and 2,292 for science) and on additional benchmarks that assess generalization across question sources and task requirements.

We use three additional medical benchmarks. (1) **HealthBench** (Arora et al., 2025) provides 5,000 realistic medical conversations with physician-authored rubrics; we evaluate all final assistant responses in the context of preceding messages. (2) **LLMEval-Med** (Zhang et al., 2025b) assesses clinical knowledge, understanding, reasoning, safety, and generation; we use its 667-question public test set. (3) **MedQA** (Jin et al., 2021) contains medical board-examination questions; we use its 1,273-question, four-option English test split.

The scientific suite includes two additional benchmarks. (1) **ResearchQA** (Yifei et al., 2026) evaluates answers to research questions using question-specific criteria derived from survey literature. We use a prepared subset of 3,750 questions and preserve the original task instructions, including any citation, length, or temporal constraints. (2) **GPQA-D** (Rein et al., 2024) is the Diamond subset of GPQA, comprising 198 expert-written multiple-choice questions in biology, chemistry, and physics.

Evaluation criteria. Each benchmark is evaluated using rubric criteria with associated point values. RaR-Medicine, RaR-Science, and HealthBench assess response quality through weighted criteria, including negatively weighted penalties. ResearchQA criteria describe the expected answer content and carry unit weights. For LLMEval-Med, clinical usability is assessed through a single unit-weight criterion based on the expert reference answer and checklist. The reported metric is the clinical-usability pass rate, with a passing standard corresponding to a rating of at least 4 out of 5. MedQA and GPQA-D assess final-answer correctness through a single unit-weight criterion. All criteria are assessed using the grading procedure described below.

LLM-based evaluation. We use gpt-oss-120b to evaluate all benchmarks and methods, with temperature 0 and a 4,096-token output limit. Given the task prompt, generated response, and rubric, the judge assesses all criteria in one call using the template in Figure 6. For each evaluation example e, let s_{ej} denote the point value of criterion j and $z_{ej} \in \{0, 1\}$ its satisfaction indicator. We compute HealthBench-style response and benchmark scores as

$$P_e = \sum_{j:s_{ej}>0} s_{ej}, \qquad r_e = \max\left\{0, \frac{\sum_j s_{ej} z_{ej}}{P_e}\right\}, \qquad \mathrm{Score}(\mathcal{D}) = \frac{100}{|\mathcal{D}|} \sum_{e \in \mathcal{D}} r_e. \tag{12}$$

To reduce sampling variability, eight responses per question are generated for GPQA-D (avg@8) and two for LLMEval-Med (avg@2), with binary scores averaged across all questions and samples. All other benchmarks use one response per question (avg@1). Benchmark scores are reported as percentages, and the overall score is the unweighted mean of the seven benchmark scores.

Evaluation Prompt Template

You are an expert evaluator. Given a user prompt, a generated response, and a list of quality rubrics, please evaluate the response against EACH rubric.
For each rubric,
- Mark "PRESENT" if the criterion is satisfied, or "NOT_PRESENT" if it is not. For example, given the response "Apples are red", the rubric "Mentions apples" is PRESENT, "Does not mention strawberries" is also PRESENT since the response doesn't mention strawberries and "Mentions oranges" is NOT_PRESENT. Also, "Avoids mentioning strawberries" is PRESENT because the response doesn't mention strawberries. However, "Avoids mentioning apples" is NOT_PRESENT because the response mentions apples.
- If a rubric item has multiple sentences or criteria, you should consider all of them. If any of the criteria is not met, the answer should be NOT_PRESENT. Only return PRESENT if all of the criteria are met.
- One important exception: if a criterion says "such as", "for example", or "including", the response does not have to include all of the examples listed to meet the criteria.
Start your response with a valid JSON object that starts with "```json" and ends with "```".
The keys must be the numbers of the rubrics provided and the values must be either "PRESENT" or "NOT_PRESENT" based on your evaluation. Ensure the JSON is valid and contains no extra text or explanations.
Example response:

```json
{ "1": "PRESENT", "2": "NOT_PRESENT", "3": "PRESENT" }
```

<Prompt> {prompt_str} **</Prompt>**
<Response> {response} **</Response>**
<Rubrics> {rubrics_str} **</Rubrics>**

Figure 6: Prompt template for the LLM judge. Blue placeholders denote the task prompt, model response, and numbered rubric criteria.

B.2 Training and Implementation Details

Optimization and decoding. We use `verl` (Sheng et al., 2025) with FSDP for policy training and vLLM for response generation. Table 4 summarizes the optimization and decoding settings. We disable top-k truncation during rollout sampling. Evaluation responses are generated using only the task context, with rubric criteria reserved for the judge.

Criterion-conditioned inputs. For each selected criterion, we append the corresponding instruction in Figure 7 to the final user message, separated from the original content by a blank line. All preceding conversation messages are retained. The template is selected according to the sign of the criterion's original point value s_j, independently of the sign of its group-relative credit budget A_{ij}.

Response alignment and masking. We apply the model's chat template to each conditioned conversation and append the original sampled response token IDs. This preserves the response sequence across scoring passes despite differences in context length. We extract the logits that predict response tokens and align them by response position before computing relevance. The response mask excludes prompt and padding positions from token-weight normalization and the policy loss.

B.3 Baseline Reproduction

We reproduce RaR (Gunjal et al., 2026), RuscaRL (Zhou et al., 2026), HeRL (Zhang et al., 2026b), POW3R (Tyagi et al., 2026), and CoRT (Zhang et al., 2026a) following their official implementations or the standard procedures described in the original papers.

OPSD. We implement rubric-conditioned OPSD using the self-teacher scoring and distillation components of the official SDPO codebase (Hübotter et al., 2026). For each question, the privileged

Positive-Point Criterion

{prompt}

When answering, you should satisfy the following rubric criterion:
- {criterion}

Negative-Point Criterion

{prompt}

When answering, do not make the following mistake:
- {criterion}

Figure 7: Prompt templates for criterion-conditioned forward passes in CoCA. Blue placeholders denote the original final user message and the selected criterion.

information consists of the complete set of rubric criteria. The student generates a response from the ordinary task prompt, and the self-teacher scores the same response under a context augmented with these criteria. Teacher predictions are conditioned on the sampled response prefix and detached from backpropagation. OPSD alone optimizes the reverse-KL objective in Eq. (2) over response tokens.

RaR+OPSD. We combine the RaR policy loss with the OPSD loss:

$$\mathcal{L}_{\mathrm{RaR+OPSD}} = \mathcal{L}_{\mathrm{GRPO}} + \lambda \mathcal{L}_{\mathrm{OPSD}}, \qquad \lambda = 0.01. \tag{13}$$

Here, $\mathcal{L}_{\mathrm{GRPO}}$ is the policy loss in Eq. (1), computed using response-level advantages derived from rubric rewards, and $\mathcal{L}_{\mathrm{OPSD}}$ is the teacher-matching loss in Eq. (2). Both terms are computed on the same sampled responses and jointly optimized in each policy update. The student receives the ordinary task prompt, with rubric information confined to the teacher context. We select $\lambda = 0.01$ according to the observed gradient magnitudes: during training, the gradient norm of the weighted distillation term $\lambda \mathcal{L}_{\mathrm{OPSD}}$ is approximately one fifth of that of the policy loss $\mathcal{L}_{\mathrm{GRPO}}$. This keeps distillation active as an auxiliary objective while retaining policy optimization as the primary training signal. An excessively large λ can let distillation dominate the combined objective, whereas an excessively small one can make its contribution negligible.

C Additional Experimental Results

C.1 Results Across Base Models and Judges

Consistent gains across base models. To examine whether our findings extend beyond Qwen3-8B, we evaluate CoCA on Qwen3-4B (Yang et al., 2025) and Olmo3-7B-Instruct-SFT (Olmo et al., 2025) against three representative baselines: RaR, RaR+OPSD, and HeRL, using gpt-oss-120b as the judge. As shown in Table 5, CoCA achieves average scores of 58.0% and 41.5%, exceeding RaR by 4.1% and 1.3%, respectively. CoCA also surpasses the strongest baseline HeRL, by 2.4% and 1.4% on average. This consistent advantage supports the benefit of resolving criterion-level credit beyond response-level rubric rewards. Meanwhile, RaR+OPSD underperforms RaR on all seven benchmarks for both models, with average declines of 1.2% and 2.1%, reinforcing our finding that an auxiliary rubric-conditioned distillation objective can undermine reward-driven learning. Together, these results reproduce the main-table trends across model scales and architectures, supporting the effectiveness of outcome-grounded criterion-level credit assignment beyond the default base model.

Consistent gains across judge models. We extend the evaluation to DeepSeek-V4.1-Flash (Xu et al., 2026a) and rescore the same set of responses from all Qwen3-8B-based methods, keeping all other evaluation settings unchanged. As shown in Table 6, CoCA outperforms all baselines on all seven benchmarks, achieving an average score of 64.8%. It improves over RaR, the standard rubric-based RL baseline, by an absolute 3.4%, and surpasses HeRL, the strongest baseline on average, by

Table 5: Experimental results (%) for Qwen3-4B and Olmo3-7B-Instruct-SFT with gpt-oss-120b as the judge. Avg. denotes the mean across all seven benchmarks. The best results within each model block are highlighted in **bold**. Arrows indicate improvements (↑) or declines (↓) relative to the corresponding initial model.

Method	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
Qwen3-4B								
Initial	45.6 (0.0)	37.8 (0.0)	48.3 (0.0)	66.0 (0.0)	64.3 (0.0)	43.8 (0.0)	43.2 (0.0)	49.9 (0.0)
RaR	55.2 (↑9.6)	43.6 (↑5.8)	50.3 (↑2.0)	66.2 (↑0.2)	69.2 (↑4.9)	48.3 (↑4.5)	44.6 (↑1.4)	53.9 (↑4.0)
RaR+OPSD	52.4 (↑6.8)	42.2 (↑4.4)	49.1 (↑0.8)	65.8 (↓0.2)	68.0 (↑3.7)	47.6 (↑3.8)	43.7 (↑0.5)	52.7 (↑2.8)
HeRL	58.0 (↑12.4)	45.7 (↑7.9)	51.9 (↑3.6)	68.7 (↑2.7)	70.3 (↑6.0)	49.0 (↑5.2)	45.4 (↑2.2)	55.6 (↑5.7)
CoCA (ours)	**61.9** (↑16.3)	**49.1** (↑11.3)	**54.0** (↑5.7)	**70.6** (↑4.6)	**72.1** (↑7.8)	**50.7** (↑6.9)	**47.9** (↑4.7)	**58.0** (↑8.1)
Olmo3-7B-Instruct-SFT								
Initial	31.8 (0.0)	26.1 (0.0)	12.0 (0.0)	49.3 (0.0)	52.6 (0.0)	39.3 (0.0)	32.2 (0.0)	34.8 (0.0)
RaR	43.0 (↑11.2)	32.8 (↑6.7)	13.5 (↑1.5)	51.8 (↑2.5)	62.0 (↑9.4)	45.8 (↑6.5)	32.6 (↑0.4)	40.2 (↑5.4)
RaR+OPSD	40.1 (↑8.3)	31.0 (↑4.9)	11.3 (↓0.7)	49.7 (↑0.4)	61.3 (↑8.7)	44.8 (↑5.5)	28.6 (↓3.6)	38.1 (↑3.3)
HeRL	42.7 (↑10.9)	32.4 (↑6.3)	**15.8** (↑3.8)	50.9 (↑1.6)	62.6 (↑10.0)	45.7 (↑6.4)	30.7 (↓1.5)	40.1 (↑5.3)
CoCA (ours)	**44.1** (↑12.3)	**33.9** (↑7.8)	15.5 (↑3.5)	**53.5** (↑4.2)	**63.1** (↑10.5)	**47.1** (↑7.8)	**33.4** (↑1.2)	**41.5** (↑6.7)

Table 6: Experimental results (%) on medical and scientific benchmarks, using Qwen3-8B as the base model and DeepSeek-V4.1-Flash as the judge. Avg. denotes the mean score across all seven benchmarks, and the best results are highlighted in **bold**. Arrows indicate improvements (↑) or declines (↓) relative to the initial model.

Method	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
Initial	57.9 (0.0)	52.1 (0.0)	44.3 (0.0)	72.0 (0.0)	67.7 (0.0)	51.1 (0.0)	49.7 (0.0)	56.4 (0.0)
RaR	68.2 (↑10.3)	57.5 (↑5.4)	47.4 (↑3.1)	73.8 (↑1.8)	73.1 (↑5.4)	58.0 (↑6.9)	51.6 (↑1.9)	61.4 (↑5.0)
OPSD	48.9 (↓9.0)	38.2 (↓13.9)	42.9 (↓1.4)	69.0 (↓3.0)	66.1 (↓1.6)	48.7 (↓2.4)	46.6 (↓3.1)	51.5 (↓4.9)
RaR+OPSD	64.6 (↑6.7)	52.3 (↑0.2)	44.6 (↑0.3)	72.1 (↑0.1)	68.3 (↑0.6)	51.2 (↑0.1)	48.8 (↓0.9)	57.4 (↑1.0)
RuscaRL	60.1 (↑2.2)	54.5 (↑2.4)	44.9 (↑0.6)	73.8 (↑1.8)	66.4 (↓1.3)	54.1 (↑3.0)	44.4 (↓5.3)	56.9 (↑0.5)
HeRL	70.6 (↑12.7)	60.3 (↑8.2)	48.6 (↑4.3)	73.3 (↑1.3)	72.6 (↑4.9)	58.9 (↑7.8)	51.6 (↑1.9)	62.3 (↑5.9)
POW3R	69.6 (↑11.7)	58.5 (↑6.4)	48.1 (↑3.8)	72.4 (↑0.4)	72.3 (↑4.6)	59.7 (↑8.6)	52.0 (↑2.3)	61.8 (↑5.4)
CoRT	68.9 (↑11.0)	57.8 (↑5.7)	47.8 (↑3.5)	75.9 (↑3.9)	71.6 (↑3.9)	58.4 (↑7.3)	47.7 (↓2.0)	61.2 (↑4.8)
CoCA (ours)	**73.1** (↑15.2)	**63.3** (↑11.2)	**51.1** (↑6.8)	**77.6** (↑5.6)	**74.2** (↑6.5)	**60.0** (↑8.9)	**54.4** (↑4.7)	**64.8** (↑8.4)

2.5%. These gains are comparable to the 4.2% and 3.1% margins under gpt-oss-120b. Although the absolute scores vary across judges, the relative performance gaps among methods remain broadly consistent. This consistency supports the robustness of CoCA's advantage across judge models.

C.2 Quantitative Validation of Criterion–Token Relevance

Annotation protocol. To test whether CoCA's criterion-induced distribution shifts localize the parts of a response relevant to each rubric criterion, we randomly sample 500 criterion–response pairs from each domain (Medicine and Science). For each pair, GPT-6 Astra annotates words that directly express criterion-relevant content or functionally support it, such as a heading or transition introducing that content. We then manually review 100 pairs per domain (20%) to check the quality of these annotations. For analysis, we map the word-level spans to the corresponding response-token positions. Figure 8 presents one complete response from each domain. Dark blue marks annotated words and pale blue marks the remaining words. We refer to the unmarked positions in the eligible response-token set as *remaining tokens*.

Pair-level separation. For each criterion–response pair p, we compute the mean JSD over annotated positions $\mathcal{T}_p$ and remaining eligible positions $\mathcal{N}_p$, together with their enrichment ratio:

$$\mu_p^{\text{ann}} = \frac{1}{|\mathcal{T}_p|} \sum_{t \in \mathcal{T}_p} d_{p,t}, \qquad \mu_p^{\text{rem}} = \frac{1}{|\mathcal{N}_p|} \sum_{t \in \mathcal{N}_p} d_{p,t}, \qquad r_p = \frac{\mu_p^{\text{ann}}}{\mu_p^{\text{rem}}}. \tag{14}$$

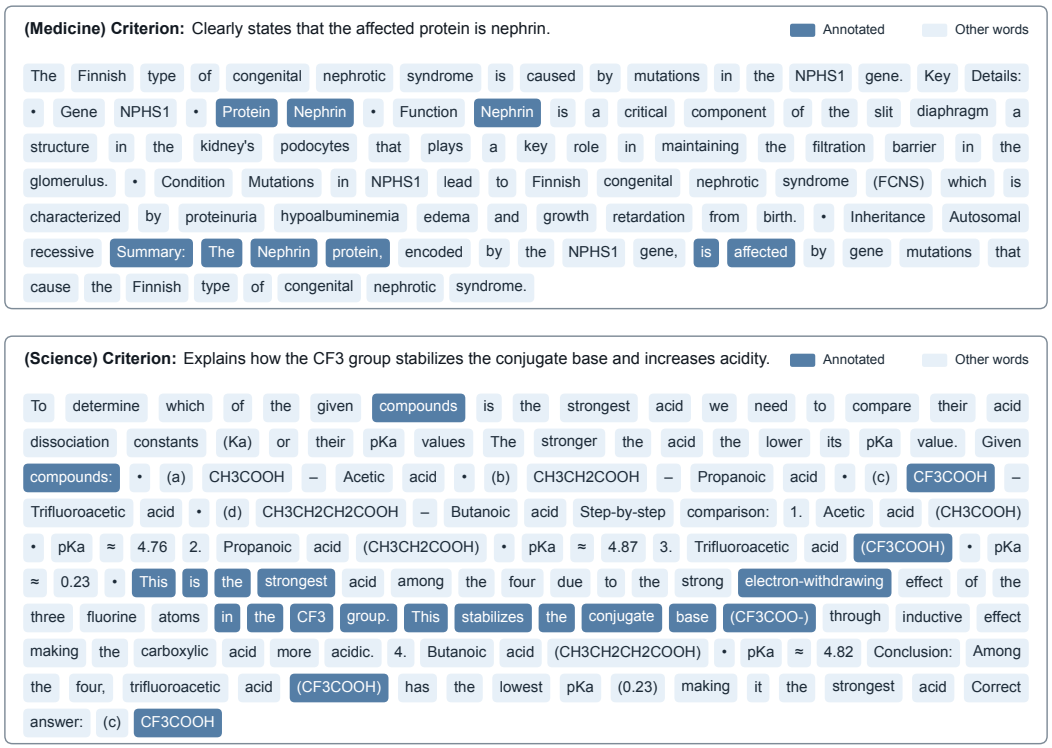


Figure 8: Complete word-level annotation examples from Medicine (top) and Science (bottom), with the corresponding rubric criterion shown above each response. Dark and pale blue denote annotated and remaining words, respectively. Subword pieces are merged into words for display.

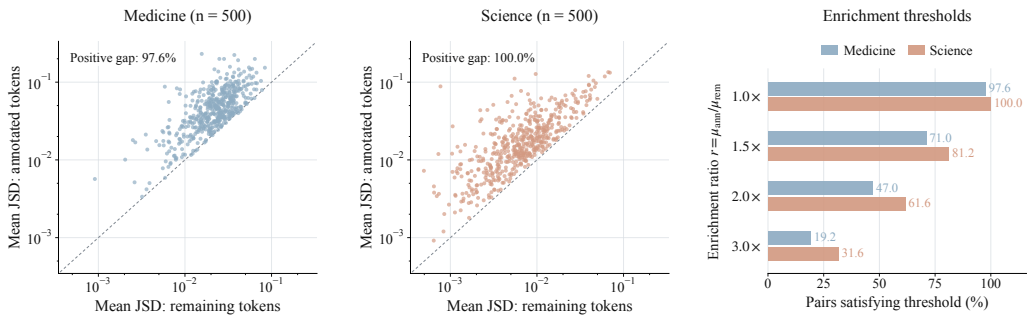


Figure 9: Criterion–token relevance over 1,000 annotated pairs. The scatterplots compare mean JSD at annotated and remaining positions; the dashed diagonal denotes equality. The right panel reports the percentage of pairs meeting each enrichment-ratio threshold.

Here, $d_{p,t}$ denotes the JSD relevance score at position t. We report the *positive-gap rate*, defined as the percentage of pairs satisfying $\mu_p^{\text{ann}} > \mu_p^{\text{rem}}$. For each threshold τ, we also report the percentage of pairs with an enrichment ratio $r_p \geq \tau$ to quantify the magnitude of this separation. As shown in Figure 9, the positive-gap rate is 97.6% in Medicine and 100.0% in Science. The fractions with $r_p \geq 1.5$ are 71.0% and 81.2%, those with $r_p \geq 2.0$ are 47.0% and 61.6%, and those with $r_p \geq 3.0$ are 19.2% and 31.6%, respectively. Overall, 988 of the 1,000 pairs (98.8%) satisfy $\Delta_p = \mu_p^{\text{ann}} - \mu_p^{\text{rem}} > 0$. These results show that JSD is higher on average at criterion-relevant positions, supporting its use for criterion-level credit allocation. The observed separation also provides empirical support for the positive mean-gap condition underlying the credit-resolution guarantee in Proposition 3.

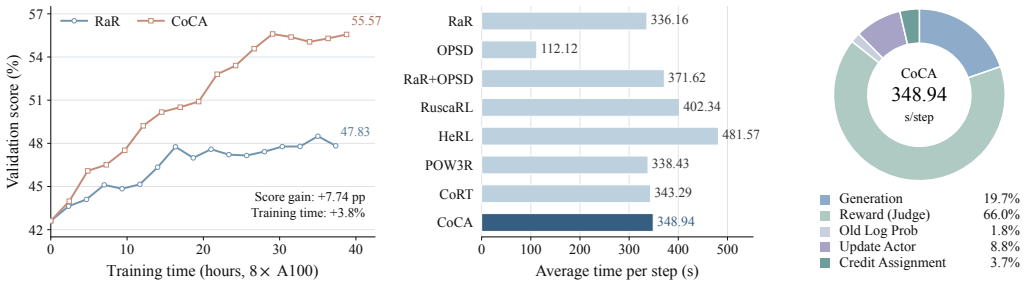


Figure 10: Training efficiency of Qwen3-8B on Medicine using 8×A100 GPUs. Left: validation performance versus training time. Middle: average training step time for all methods. Right: CoCA's runtime breakdown, with the total step time shown at the center and component proportions listed below.

C.3 Analysis of Training Efficiency

Performance versus training time. We evaluate Qwen3-8B trained with CoCA and RaR on the Medicine validation set every 25 steps over 400 training steps, using the same 8×A100 GPUs. Given the differences in per-step cost across methods, we compare validation performance as a function of training time rather than training steps. As shown in the left panel of Figure 10, CoCA reaches 55.57% after 38.77 hours, compared with 47.83% after 37.35 hours for RaR. This corresponds to a gain of 7.74 percentage points with only 3.8% more training time. Moreover, CoCA achieves 49.22% at step 125 after only 12.12 hours, already surpassing RaR's best performance at step 375 (35.02 hours). Thus, CoCA achieves higher performance with a 2.89× speedup in training time. These results demonstrate that the learning gains from CoCA's additional forward passes outweigh their computational overhead, yielding a net improvement in training efficiency.

Per-step cost and runtime breakdown. To quantify CoCA's additional training cost, we measure the average step time of all eight methods under the same model, domain, and hardware configuration and profile its runtime. As shown in the middle panel of Figure 10, CoCA takes 348.94 seconds per step compared with 336.16 seconds for RaR, an increase of only 3.8% despite its additional criterion-conditioned forward passes. Its step time is also lower than those of RaR+OPSD, RuscaRL, and HeRL, showing that its finer-grained credit assignment remains computationally competitive with other approaches to rubric supervision. The right panel shows that credit assignment takes only 12.78 seconds per step (3.7% of the total), while judge-based reward evaluation and response generation together account for 85.7%. Most training time is therefore spent producing responses and obtaining rubric feedback, with comparatively little devoted to allocating that feedback across tokens. This is consistent with CoCA's design: it reuses criterion-level judgments from reward evaluation and restricts additional criterion-conditioned relevance scoring to selected criteria. Together with the gains in learning efficiency, these results indicate that CoCA makes more effective use of costly rubric feedback with only modest additional computation.

C.4 Analysis of Training Dynamics

To better understand how criterion-level credit assignment affects policy optimization, we compare the training dynamics of CoCA and RaR in Medicine and Science. Specifically, we track training reward, response length, and policy entropy over 400 training steps to characterize learning progress, generation behavior, and exploration, as shown in Figure 11.

Reward optimization. We first examine reward progression to assess the effectiveness of rubric-based optimization. In Medicine, CoCA establishes a reward advantage early in training and maintains a clear gap as both methods approach a plateau. In Science, the two methods exhibit similar initial progress, but CoCA gradually achieves higher rewards, with the difference becoming more apparent in the later stages. This advantage persists through the end of training in both domains, suggesting that criterion-level credit assignment helps the policy make more effective use of rubric feedback and reach higher reward levels.

Table 7: Ablation results (%) for criterion selection strategies. Arrows indicate changes relative to CoCA.

Strategy	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
CoCA	**69.8** (0.0)	**54.3** (0.0)	**61.8** (0.0)	**77.6** (0.0)	**74.2** (0.0)	**59.6** (0.0)	**54.4** (0.0)	**64.5** (0.0)
Random	67.3 (↓2.5)	49.6 (↓4.7)	56.1 (↓5.7)	74.1 (↓3.5)	73.4 (↓0.8)	57.9 (↓1.7)	51.3 (↓3.1)	61.4 (↓3.1)
Top $\|s_j\|$	69.4 (↓0.4)	52.3 (↓2.0)	59.0 (↓2.8)	76.8 (↓0.8)	74.0 (↓0.2)	58.6 (↓1.0)	52.2 (↓2.2)	63.2 (↓1.3)

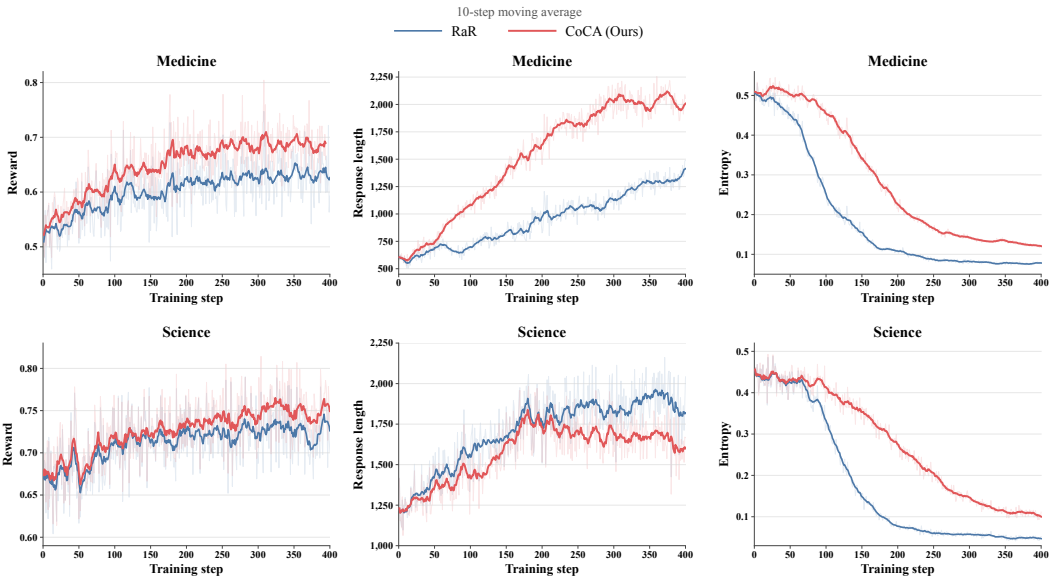


Figure 11: Training dynamics of CoCA and RaR in Medicine (top) and Science (bottom). Columns show reward, response length, and policy entropy, with faint stepwise curves and bold 10-step moving averages.

Response length. We additionally compare response length to assess whether the reward gains depend on producing longer outputs. In Medicine, CoCA's responses become substantially longer than RaR's as training progresses, alongside its higher reward. However, in Science, CoCA's response length peaks around the middle of training and then decreases, yielding shorter responses than RaR in the second half even as its reward improves further. These contrasting trends show that CoCA can achieve higher reward with either longer or shorter responses, depending on the domain, and that its gains do not require uniformly increasing output length.

Exploration during training. Finally, we track policy entropy to characterize how exploration evolves during optimization. In both domains, RaR exhibits a rapid entropy decline early in training, followed by a prolonged period at a low level. CoCA's entropy decreases more gradually and remains higher through the later stages, indicating slower concentration of its token distributions. Together with the higher rewards, this pattern suggests that CoCA supports effective reward optimization while retaining more scope for exploration.

C.5 Effect of Criterion Selection

To assess the effect of criterion selection, we compare CoCA's default strategy with two alternatives, keeping the selection budget at $k = 2$ and other training settings unchanged. CoCA selects criteria with the largest absolute credits $|A_{ij}|$, whereas *Random* selects criteria randomly and *Top* $|s_j|$ selects those with the largest absolute rubric point values. Unselected criteria retain uniform allocation.

As shown in Table 7, *Random* and *Top* $|s_j|$ achieve average scores of 61.4% and 63.2%, respectively, falling below CoCA by 3.1% and 1.3%. Both alternatives underperform CoCA on all seven benchmarks, while *Top* $|s_j|$ consistently outperforms *Random*. The advantage of *Top* $|s_j|$ over *Random*

Table 8: Standard deviations (%) of benchmark scores across three random seeds. Avg. reports the standard deviation of the per-seed average score over all seven benchmarks.

Method	Medicine				Science			Avg.
	RaR-Medicine	HealthBench	LLMEval-Med	MedQA	RaR-Science	ResearchQA	GPQA-D	
RaR	0.26	0.25	0.44	0.51	0.18	0.39	0.53	0.24
CoCA	0.31	0.22	0.37	0.45	0.29	0.30	0.48	0.17

suggests that rubric point values provide a useful basis for prioritization. CoCA further incorporates the magnitude of each criterion outcome's deviation from the rollout-group average through $|A_{ij}|$. Its additional gains support directing the limited criterion-conditioned computation toward criteria with larger response-specific credit magnitudes.

C.6 Robustness Across Random Seeds

To assess sensitivity to training randomness, we repeat the CoCA and RaR experiments with two additional random seeds, keeping all training and evaluation settings unchanged. Table 8 reports the standard deviations across all three runs. CoCA's benchmark-level standard deviations range from 0.22% to 0.48%, comparable to RaR's range of 0.18% to 0.53%. For the seven-benchmark average, CoCA exhibits a lower standard deviation than RaR (0.17% versus 0.24%). Moreover, its 3.1% average lead over the strongest baseline in Table 1 far exceeds the observed variation, indicating that CoCA's SOTA average performance remains robust across the evaluated seeds.

D Pseudocode of CoCA

We describe the CoCA pipeline in Algorithm 1.

Algorithm 1 CoCA: Criterion-Aware Credit Assignment

Require: Dataset $\mathcal{D}$, policy π_θ, judge J, group size G, allocation budget k, concentration β.

for each training iteration **do**
  Freeze the rollout policy $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$
  Sample a batch of prompt–rubric pairs from $\mathcal{D}$
  for each $(x, \mathcal{R})$ in the batch **do**
    Sample $\{y_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$
    Obtain criterion outcomes $\{z_{ij}\}$ using J
    $R_i \leftarrow \sum_j s_j z_{ij} / \sum_{j:s_j>0} s_j$ for all i
    $\bar{z}_j \leftarrow G^{-1} \sum_i z_{ij}$ for all j
    $A_{ij} \leftarrow s_j(z_{ij} - \bar{z}_j)/[\sum_{\ell:s_\ell>0} s_\ell]$ for all i, j
    for $i = 1, \ldots, G$ **do**
      $T_i \leftarrow |y_i|$; $\widehat{A}_{i,t} \leftarrow \sum_j A_{ij}$ for all t
      $\mathcal{S}_i \leftarrow \mathrm{TopK}_{j:A_{ij}\neq 0}(|A_{ij}|, k)$
      $p_{i,t} \leftarrow \pi_{\theta_{\text{old}}}(\cdot \mid h_{i,t})$ for all t
      for each $j \in \mathcal{S}_i$ **do**
        $q_{ij,t} \leftarrow \pi_{\theta_{\text{old}}}(\cdot \mid h_{i,t}(c_j))$ for all t
        $d_{ij,t} \leftarrow \mathrm{sg}[D_{\mathrm{JS}}(p_{i,t} \| q_{ij,t})]$ for all t
        $\boldsymbol{\omega}_{ij} \leftarrow T_i \,\mathrm{softmax}(\beta \mathbf{d}_{ij})$
        $\widehat{A}_{i,t} \leftarrow \widehat{A}_{i,t} + A_{ij}([\boldsymbol{\omega}_{ij}]_t - 1)$ for all t
      end for
    end for
  end for
  Update θ on the batch using Eq. (1), replacing A_i with fixed $\widehat{A}_{i,t}$
end for
return π_θ