RUBRICEVO: SUSTAINING LEARNING SIGNALS VIA EVOLVING GRAPH-BASED MULTI-LEVEL RUBRICS IN OPEN-ENDED REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Rubric-based reinforcement learning provides a practical way to optimize open-ended tasks without verifiable answers, yet fixed, flat rubrics become increasingly ineffective as the language-model policy evolves. Once broad criteria are mastered, rubric scores saturate and reward gaps among on-policy responses diminish; simultaneously, optimizing against a single global checklist favors a narrow set of rubric-compatible behaviors and suppresses alternative valid strategies. We propose RUBRICEVO, which represents each query-specific rubric as an evolving, multi-level graph. The graph organizes rubric into fixed *Principle* nodes that preserve fundamental requirements, *Aspect* nodes that capture major quality dimensions, and *Specific* nodes that provide fine-grained, strategy-dependent guidance. While Principle nodes remain unchanged, Aspect and Specific nodes can be replaced or branched as the policy evolves. During training, RubricEvo measures reward resolution and semantic diversity over on-policy response groups to guide the evolution of the graph-based rubric. Experiments across multiple domains and Qwen3-1.7B/4B/8B policies show that RubricEvo improves downstream performance while preserving more discriminative rewards and greater high-quality output diversity. All datasets and code are available at `https://anonymous.4open.science/r/RubricEvo/`.

1 INTRODUCTION

Reinforcement learning for large language models depends on evaluators that can reliably distinguish better outputs from worse ones (Ouyang et al., 2022; Lambert et al., 2025; Malik et al., 2026). While tasks such as mathematics and code generation often provide verifiable rewards, open-ended tasks such as writing and dialogue admit no unique correct answer and therefore commonly rely on LLM judges guided by natural-language rubrics (Liu et al., 2023; Zheng et al., 2023). However, a rubric fixed before training may become less informative as the policy improves. Once broad criteria such as relevance, coherence, and completeness are routinely satisfied, rubric scores begin to saturate (Wu et al., 2025). As shown in Figure 1(b) and (c), the smoothed reward gap under static rubrics falls to 72.6% of its initial value after 200 GRPO (Shao et al., 2024) steps, while on-policy diversity declines to as low as 84.5%. Crucially, group-relative optimization can rescale reward differences but cannot recover discriminative information that the evaluator no longer provides. As responses become increasingly indistinguishable under fixed criteria, their relative ordering becomes less reliable, progressively weakening the learning signal.

In fact, output diversity is intrinsic to open-ended generation: the same prompt may admit multiple high-quality responses that follow different reasoning paths, structures, or styles. Prior work shows that RLHF can narrow the output distribution and concentrate generation on a few high-reward modes, even when multiple valid answers exist (Kirk et al., 2024; Sun et al., 2025; Anschel et al., 2025). Repeatedly applying the same criteria can further favor a narrow set of rubric-compatible strategies, suppressing valid alternatives and leaving the evaluator with homogeneous responses.

This loss of diversity is closely tied to the reward design used for open-ended generation. Existing methods typically organize a query-specific rubric as a flat checklist: an LLM judge scores each criterion independently and linearly combines the scores into a scalar reward (Rezaei et al., 2025;

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: (a) Flat rubrics aggregate all criteria jointly; evolving graph-based rubrics distinguish shared requirements and alternative strategies. (b–c) On writing with Qwen3-4B, RUBRICEVO better preserves reward resolution and diversity than static-rubric GRPO.

Bhattarai et al., 2026; Liu et al., 2026b; Chen et al., 2026; He et al., 2026). Such aggregation implicitly treats all listed criteria as jointly applicable, even when some correspond to alternative response strategies, while valid strategies absent from the checklist receive no strategy-specific credit. To quantify how often these limitations arise, we audit 1,000 existing query-specific rubrics and find that 9.3% contain conflicting evaluation criteria, while 27.0% omit at least one distinct strategy capable of producing a high-quality response. For example, a story about grief may succeed through restrained ambiguity or explicit emotional development, yet a single checklist can favor one strategy and progressively narrow the response distribution.

These observations motivate RUBRICEVO, which organizes each query-specific rubric as a versioned graph of fixed Principle Rubrics and evolving Aspect and Specific Rubrics. Compatible criteria are aggregated jointly, whereas strategy-specific alternatives form branches from which only the best-matching path contributes to reward. During training, degradation in reward resolution or semantic diversity triggers validated REPLACE or BRANCH operations, and accepted graph versions are used to score subsequent rollouts.

We evaluate RUBRICEVO across multiple open-ended domains and model scales. Compared with base models, static rubric-based RL and existing rubric-evolution baselines, RUBRICEVO better preserves within-group reward resolution and semantic diversity throughout training while improving downstream generation quality. These results suggest that sustaining learning in open-ended reinforcement learning requires adapting the evaluation structure together with optimizing the policy.

Our main contributions are as follows:

- We identify and empirically characterize two coupled limitations of static rubric-based RL: the progressive loss of within-group reward resolution and the contraction of output diversity as the policy improves.
- We introduce RUBRICEVO, an evolving multi-level rubric graph with validated REPLACE and BRANCH operations. Extensive experiments show that it preserves reward resolution and diversity while improving generation quality.
- We construct a high quality dataset that augments flat, query-specific rubrics with graph annotations, distinguishing indispensable core criteria, mutable frontier criteria, and strategy-dependent alternatives. This dataset provides structured initialization for rubric evolution across open-ended domains.

2 Related Work

Dynamic and Evolving Rubrics. Recent methods adapt rubrics during optimization. DR. TULU evolves retrieval-grounded criteria for deep research, RLCER anchors self-evolving reasoning criteria to verifiable outcomes, and EVORUBRIC jointly optimizes a reasoner and rubric generator (Shao et al., 2025; Sheng et al., 2026; Guan et al., 2026). EVORUBRICS adversarially co-trains policy and rubric-generator LoRA adapters within each training step (Ding et al., 2026; Hu et al., 2021). More recent works include DYNAMICRUBRIC, which generates response-set-conditioned weighted binary criteria while alternating evaluator and policy optimization, and SERPO, which evolves response archives and discriminative criteria for open-ended test-time reinforcement learning (Wang et al., 2026a;b). These methods generally evolve flat criterion sets. In contrast, RUBRICEVO triggers evolution when reward resolution or response diversity deteriorates and organizes shared and strategy-specific criteria in a persistent rubric graph. Broader background appears in Appendix A.1.

3 Dataset Construction

To provide an initial rubric graph for each training query before online evolution, we construct 9,000 graph-annotated instances across writing, general chat, and medical question answering, with 3,000 instances per domain. Writing prompts are collected from the training split of **LongWriter-6k** (Bai et al., 2025), chat prompts from the **Chat subset of RubricHub** (Li et al., 2026), and medical prompts from the training split of **RaR-Medicine** (Gunjal et al., 2026). We use only training partitions, remove unsuitable or duplicated queries, and ensure no overlap with the evaluation benchmarks. Rubric graphs are annotated by `GPT-5.5` following a human-authored protocol; the initial annotation pass rejects 15.8%, 35.6%, and 35.4% of the Writing, Chat, and Medical candidates, respectively. After automatic structural validation, we conduct stratified review on 300 annotations per domain: human reviewers retain 89% of Writing and 92% of Chat annotations, while an independently prompted, context-isolated `GPT-5.5` reviewer retains 84% of Medical annotations. Rejected annotations are revised or replaced before inclusion. Detailed curation, annotation, and quality-control procedures are provided in Appendix A.6.

4 RubricEvo

Figure 2 presents an overview of RUBRICEVO. Given an open-ended prompt, the policy generates a group of responses that are evaluated using the current rubric graph. The resulting rewards are normalized within the group to optimize the policy with GRPO. In parallel, RUBRICEVO monitors reward resolution and semantic diversity. Persistent degradation activates an online evolution process, in which representative near-tie responses and diverse historical responses guide an external LLM to propose REPLACE or BRANCH operations. A replay gate validates each proposal before committing the next graph version. Detailed proofs for this section are provided in Appendix A.4.

4.1 Problem Formulation

For an input x at training step t, the policy π_{θ_t} samples a group of responses $\mathcal{Y}_x^{(t)} = \{y_{x,1}^{(t)}, \ldots, y_{x,G}^{(t)}\}$. Given a query-specific flat rubric $\mathcal{R}_x = \{c_1, \ldots, c_M\}$, an LLM evaluator scores every criterion and linearly aggregates the scores:

$$\hat{r}_{x,i}^{(t)} = \sum_{j=1}^{M} w_j \hat{s}_{x,i,j}^{(t)}, \qquad \hat{A}_{x,i}^{(t)} = \frac{\hat{r}_{x,i}^{(t)} - \bar{\hat{r}}_x^{(t)}}{\mathrm{Std}(\hat{r}_{x,1}^{(t)}, \ldots, \hat{r}_{x,G}^{(t)}) + \epsilon},$$

where $\hat{A}_{x,i}^{(t)}$ is the group-normalized advantage used by GRPO (Shao et al., 2024).

Reward resolution. We define reward resolution as the mean pairwise reward gap among responses sampled for the same prompt:

$$\Delta_x^{(t)} = \frac{2}{G(G-1)} \sum_{i<j} \left| \hat{r}_{x,i}^{(t)} - \hat{r}_{x,j}^{(t)} \right|, \qquad \Delta^{(t)} = \frac{1}{|\mathcal{A}^{(t)}|} \sum_{x \in \mathcal{A}^{(t)}} \Delta_x^{(t)},$$

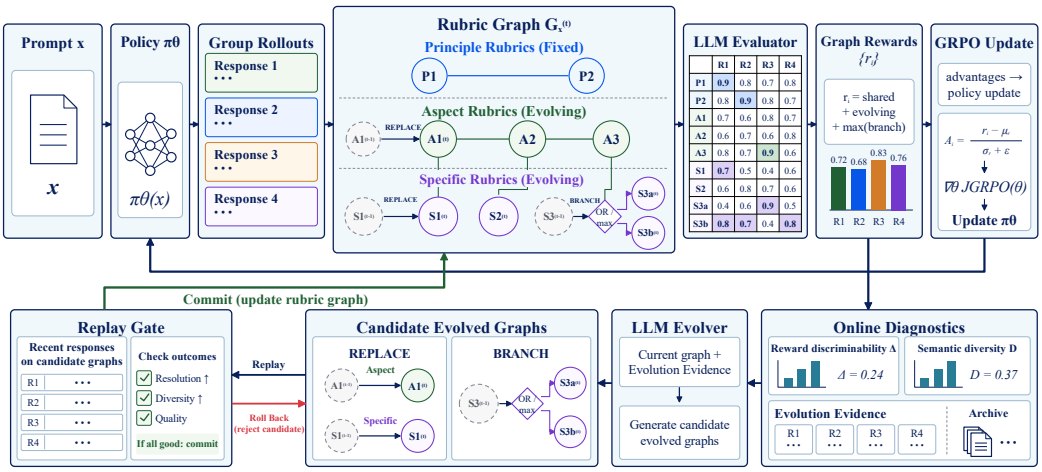

Figure 2: Overview of RUBRICEVO. The current rubric graph evaluates a group of policy responses, producing rewards for both GRPO optimization and rubric-health monitoring. Persistent degradation in reward resolution or semantic diversity triggers evidence-guided graph evolution. Proposed REPLACE and BRANCH operations are validated by a replay gate before the updated graph is used in subsequent training.

where $\mathcal{A}^{(t)}$ denotes the prompts audited at step t, and rewards are measured on a common normalized scale. As the policy masters the broad requirements of a fixed rubric, $\Delta^{(t)}$ contracts. Writing $\hat{r}_{x,i}^{(t)} = r_{x,i}^{(t)} + \eta_{x,i}^{(t)}$, once meaningful reward gaps approach the evaluator-noise scale, ties yield zero advantage and small differences may be dominated by η. Group normalization rescales these differences but cannot recover a reliable response ordering.

Diversity collapse. Let $\mathbf{e}_{x,i}^{(t)}$ denote the normalized BGE-M3 embedding of response $y_{x,i}^{(t)}$. We define semantic diversity as the mean pairwise cosine distance among responses generated for the same prompt:

$$D_{\mathrm{sem},x}^{(t)} = \frac{2}{G(G-1)} \sum_{i<j} \left(1 - \mathbf{e}_{x,i}^{(t)\top} \mathbf{e}_{x,j}^{(t)}\right), \qquad D_{\mathrm{sem}}^{(t)} = \frac{1}{|\mathcal{A}|} \sum_{x \in \mathcal{A}} D_{\mathrm{sem},x}^{(t)},$$

where $\mathcal{A}$ is the set of audit prompts. To describe its evolution, let $p_k^{(t)}$ denote the probability of semantic strategy k and μ_k its expected reward under the fixed rubric. A KL-regularized policy update approximately gives

$$\frac{p_k^{(t+1)}}{p_\ell^{(t+1)}} = \frac{p_k^{(t)}}{p_\ell^{(t)}} \exp(\alpha(\mu_k - \mu_\ell)).$$

Therefore, even a small persistent reward advantage causes the policy to concentrate progressively on a few rubric-compatible strategies, reducing $D_{\mathrm{sem}}^{(t)}$ as training proceeds.

Conflicts in flat rubrics. Let $S_0(y)$ represent shared requirements and $S_k(y)$ the quality of response y under strategy k. A flat rubric aggregates the represented strategies as

$$r_{\mathrm{flat}}(y) = S_0(y) + \sum_{k \in \mathcal{K}'} w_k S_k(y).$$

If $\mathcal{K}'$ omits a valid strategy, that strategy receives no specific credit; if incompatible strategies are all included, their scores are incorrectly treated as simultaneous requirements. An alternative-aware composition instead uses $S_0(y) + \max_k w_k S_k(y)$, rewarding a response for strongly realizing one valid strategy rather than weakly satisfying several incompatible ones. These failures motivate a graph-based rubric that evolves its quality dimensions and explicitly represents alternative evaluation paths.

4.2 MULTI-LEVEL GRAPH-BASED RUBRICS

4.2.1 MULTI-LEVEL RUBRIC HIERARCHY

For each input x, RUBRICEVO organizes query-specific evaluation requirements into three semantic levels, as summarized in Table 1. Fixed **Principle Rubrics** encode immutable constraints, evolving **Aspect Rubrics** capture high-level quality dimensions, and evolving **Specific Rubrics** instantiate these dimensions as directly scorable criteria, which may form alternative branches for substitutable strategies.

Table 1: Three semantic levels in a graph-based rubric, illustrated for the same story-writing query.

Level	Definition	Example
Principle Rubrics *Fixed*	Immutable requirements and hard constraints explicitly specified by the query.	The response must not exceed 200 words.
Aspect Rubrics *Evolving*	High-level aspects through which the quality of responses to the query can be examined.	The response should sustain narrative tension.
Specific Rubrics *Evolving*	Atomic and directly scorable requirements that realize an aspect in a concrete manner.	Reveal the secret early and build tension through its consequences. **OR** Withhold the secret until the final confrontation to build suspense.

4.2.2 GRAPH ORGANIZATION AND REWARD AGGREGATION

Graph Organization. We organize the three rubric levels into a versioned directed graph, as illustrated in Figure 1. Principle Rubrics encode immutable requirements, Aspect Rubrics represent evolving quality dimensions, and Specific Rubrics provide directly scorable realizations. Each Aspect Rubric is connected to its Specific realizations, which may be grouped into alternative branches when they represent substitutable strategies. Each Specific Rubric has one parent and belongs to at most one branch, while every active criterion occupies exactly one scoring role, preventing duplicate counting. Hierarchical edges encode refinement, whereas alternative edges connect substitutable Specific Rubrics.

Graph-Based Reward. Let $\mathcal{V}_\wedge$ denote the active rubrics that are jointly applicable and $\mathcal{S}_b$ the set of alternative Specific Rubrics in branch b. Given criterion score $s_v(y)$ and weight w_v, we compute

$$r_{\mathcal{G}}(y) = \sum_{v \in \mathcal{V}_\wedge} w_v s_v(y) + \sum_{b \in \mathcal{B}} \max_{v \in \mathcal{S}_b} w_v s_v(y).$$

The sets $\mathcal{V}_\wedge$ and $\{\mathcal{S}_b\}_{b \in \mathcal{B}}$ are mutually disjoint, ensuring that each active rubric is counted exactly once. Jointly applicable rubrics are linearly aggregated, whereas only the highest-scoring Specific Rubric within each alternative branch contributes to the reward.

4.3 ONLINE GRAPH EVOLUTION

Evolution Trigger. During training, we monitor reward resolution and semantic diversity from on-policy response groups using the measures defined in Section 4.1. Evolution is triggered when response quality exceeds a minimum threshold and either signal exhibits persistent degradation.

Graph Update. An external LLM evolver receives the current graph, its audit statistics, and representative high-quality responses. It proposes an atomic edit to an active Aspect or Specific Rubric:

$$\text{REPLACE}(c, c') \qquad \text{or} \qquad \text{BRANCH}(c, \{c_1, c_2\}).$$

REPLACE sharpens a saturated rubric, whereas BRANCH introduces alternative strategy-specific rubrics. Principle Rubrics remain fixed. For diversity-triggered evolution, the evolver also receives high-quality responses representing semantic modes that have become rare. Detailed constraints and prompts for graph updates are provided in Appendix A.8.

Quality-Gated Validation and Rollback. To ensure that only reliable and quality-preserving rubric updates enter policy training, the validation gate jointly checks structural validity, redundancy, response quality, reward resolution, and semantic diversity. We use the frozen initial rubric as a shadow evaluator, denoted by $q^{(0)}$, and restrict diversity validation to the high-quality subset

$$\mathcal{Y}_x^+ = \left\{ y_i : q^{(0)}(y_i) \geq \mathrm{Median}_j \, q^{(0)}(y_j) \right\}.$$

Since BGE-M3 distance may reflect surface variation rather than distinct strategies, we additionally estimate the effective number of semantic modes through threshold-free spectral clustering (Friedman & Dieng, 2022; Farquhar et al., 2024). Given normalized embeddings $\mathbf{z}_i$ and the similarity matrix $K_{ij} = \mathbf{z}_i^\top \mathbf{z}_j$ over $\mathcal{Y}_x^+$, we compute

$$N_{\mathrm{eff}}^+ = \exp\left(-\sum_k \lambda_k \log \lambda_k \right),$$

where $\{\lambda_k\}$ are the eigenvalues of $K/\operatorname{tr}(K)$. For diversity-oriented validation, we measure the high-quality semantic variation retained under candidate-induced selection:

$$D_{\mathrm{sel}}(\mathcal{G}) = \sum_{i,j \in \mathcal{Y}_x^+} p_i^+(\mathcal{G}) p_j^+(\mathcal{G}) \left(1 - \mathbf{z}_i^\top \mathbf{z}_j\right),$$

where $p_i^+(\mathcal{G}) \propto \exp(\beta r_{\mathcal{G}}(y_i))$ is normalized over $\mathcal{Y}_x^+$. A candidate edit is provisionally accepted only if it preserves the shadow quality measured by $q^{(0)}$ and improves the signal responsible for the update: the reward gap defined in Section 4.1 for resolution-oriented evolution or D_{sel} for diversity-oriented evolution. It is rolled back if the targeted gain is not sustained or shadow quality deteriorates beyond the permitted tolerance.

4.4 Policy Optimization

For each prompt, we score the sampled response group using the current rubric graph and optimize the policy with clipped GRPO and KL regularization (Shao et al., 2024), using the group-normalized advantages defined in Section 4.1. The evaluator and rubric graph remain external to the policy and receive no gradients, while accepted graph updates apply only to subsequent response groups.

5 Experiments

5.1 Experimental Setup

Models. We use Qwen3-1.7B, Qwen3-4B, and Qwen3-8B (Yang et al., 2025) as policy backbones, with thinking disabled for all training and evaluation runs. We employ BGE-M3 (Chen et al., 2024) to encode on-policy responses and measure their semantic diversity in the embedding space. During training, GPT-4.1-mini (OpenAI, 2025) serves as the rubric-conditioned reward evaluator, while GPT-4o (Hurst et al., 2024) acts as the rubric evolver that proposes graph updates from on-policy evidence. Detailed training and implementation settings are provided in Appendix A.7.

Benchmarks. We evaluate open-ended generation across three domains. For writing, we use **WritingBench** (Wu et al., 2026) and **Creative Writing v3** (Paech, 2025); for open-ended dialogue, we use **WildBench v2** (Lin et al., 2025); and for medical assistance, we use **HealthBench** (Arora et al., 2025) and the English subset of **LiveMedBench** (**LMB-EN**) (Yan et al., 2026). We restrict LiveMedBench to English to match our medical training data. We follow the benchmark-specific evaluation protocols, using the released Qwen-7B critic for WritingBench, and GPT-4o otherwise. All methods use identical generation and evaluation settings.

Baselines. We compare RubricEvo with three baselines. **Base Model** denotes the original policy without reinforcement learning. **RaR** (Gunjal et al., 2026) uses query-specific rubrics as fixed rewards throughout GRPO. We instantiate it with the same initial criteria represented as a flat checklist and aggregate their scores into a scalar reward; it therefore corresponds to the flat static rubric in Figure 1. **EvoRubrics** (Ding et al., 2026) is a dynamic baseline that represents the policy and a rubric generator as two trainable LoRA adapters over a shared backbone. At every step, it samples multiple responses and flat rubric sets, cross-evaluates all response–rubric pairs, and jointly updates both adapters with GRPO.

Table 2: Main results across three domains and five benchmarks. All scores are reported on a 0–100 scale. The best result within each backbone and benchmark is shown in **bold**.

	Medical		Writing		Chat
Method	**HealthBench**	**LMB-EN**	**WritingBench**	**Creative Writing**	**WildBench**
Qwen3-1.7B Backbone					
Base Model	19.49	2.74	72.00	54.37	39.79
RaR	25.17	4.62	75.32	55.56	41.17
EvoRubrics	27.84	5.11	73.19	55.97	40.66
RUBRICEVO (Ours)	**31.55**	**7.34**	**77.70**	**56.64**	**42.21**
Qwen3-4B Backbone					
Base Model	31.00	6.03	74.62	62.39	51.93
RaR	37.48	7.56	76.18	64.35	53.84
EvoRubrics	40.21	8.07	74.20	64.50	54.87
RUBRICEVO (Ours)	**42.83**	**9.67**	**78.32**	**66.21**	**55.38**
Qwen3-8B Backbone					
Base Model	36.66	8.35	77.47	68.79	56.19
RaR	44.72	9.71	79.32	70.95	58.21
EvoRubrics	45.63	11.02	79.36	70.83	58.74
RUBRICEVO (Ours)	**48.59**	**12.16**	**81.29**	**72.03**	**59.80**

5.2 EXPERIMENTAL RESULTS

Main results. Table 2 shows that RUBRICEVO consistently achieves the best performance across all model scales, domains, and benchmarks. Compared with the strongest baseline on each benchmark, it improves the average score by 2.01, 1.72, and 1.63 points for Qwen3-1.7B, Qwen3-4B, and Qwen3-8B, respectively. The gains are largest on HealthBench, where RUBRICEVO reaches 31.55, 42.83, and 48.59, outperforming the strongest baselines by 3.71, 2.62, and 2.96 points. It also yields consistent gains of 1.93–2.38 points on WritingBench and 0.51–1.06 points on WildBench, demonstrating the effectiveness of evolving rubric graphs across open-ended domains. Statistical significance results are reported in Appendix A.5.

Response diversity. Figure 3 compares response diversity normalized by the corresponding base model. RaR and Static Graph-RL reduce diversity in every setting, while EvoRubrics remains close to the base-model level. In contrast, RUBRICEVO exceeds the base model on all fifteen backbone–benchmark combinations, achieving normalized diversity scores of 1.02–1.19. Its average diversity increases from 1.046 with Qwen3-1.7B to 1.096 with Qwen3-4B and 1.130 with Qwen3-8B. Together with the performance gains in Table 2, these results show that RUBRICEVO improves generation quality without sacrificing output diversity. Using the same threshold-free spectral clustering employed in quality-gated validation, Appendix A.3 shows that RUBRICEVO consistently preserves broader semantic-mode coverage across model scales.

5.3 ABLATION STUDY

Ablation setup. We conduct ablations on writing with Qwen3-1.7B, evaluating all variants on WritingBench and Creative Writing v3 under otherwise identical settings. We additionally report normalized reward resolution $\widetilde{\Delta}$ and BGE-M3 semantic diversity $\widetilde{D}_{\mathrm{sem}}$, both computed on the same audit prompts and normalized to the Base Model value of 1.0.

Does Graph Structure Matter? The upper half of Table 3 isolates the contribution of graph organization while retaining online rubric evolution. Removing the graph and linearly aggregating all criteria reduces the WritingBench and Creative Writing scores from $77.70/56.64$ to $76.12/55.87$, while both reward resolution and semantic diversity fall below their initial levels. Removing the hierarchy results in a smaller degradation, whereas removing alternative paths reduces $\widetilde{D}_{\mathrm{sem}}$ from 1.073 to 0.980. These results show that the multi-level organization improves evaluation quality, while alternative paths are particularly important for preserving diverse response strategies.

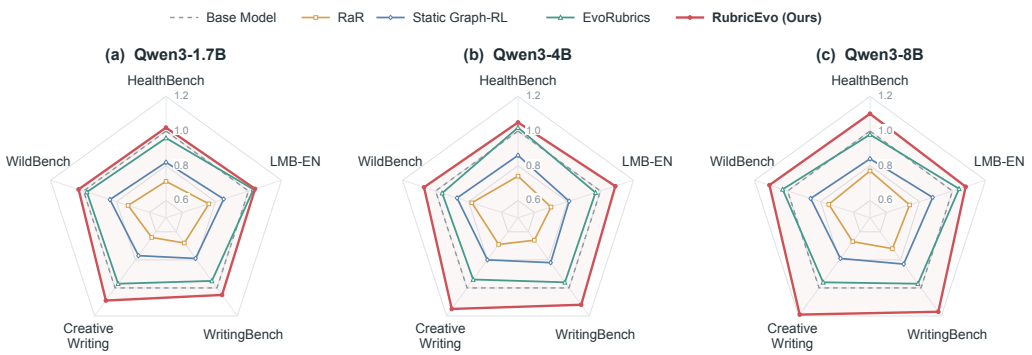

Figure 3: Response diversity across five benchmarks with Qwen3-1.7B, Qwen3-4B, and Qwen3-8B backbones. Scores are normalized by the corresponding base model (1.00); higher values indicate greater response diversity.

Does Online Evolution Matter? The lower half of Table 3 examines whether the rubric graph must evolve with the policy. Disabling online evolution produces the largest degradation, reducing the WritingBench and Creative Writing scores to $75.94/55.62$ and lowering reward resolution and semantic diversity to $0.820/0.890$. Removing REPLACE primarily weakens reward resolution, whereas removing BRANCH largely preserves resolution but reduces semantic diversity. The two operations are therefore complementary: REPLACE sharpens saturated criteria, while BRANCH introduces alternative paths for underrepresented response strategies.

Table 3: Ablations of graph organization and online evolution in writing.

Method	WB	CW	$\widetilde{\Delta} \uparrow$	$\widetilde{D}_{\text{sem}} \uparrow$
RUBRICEVO w/o Graph	76.12	55.87	0.910	0.950
RUBRICEVO w/o Hierarchy	77.08	56.31	0.980	1.040
RUBRICEVO w/o Alternative Paths	76.54	56.06	0.960	0.980
RUBRICEVO w/o Online Evolution	75.94	55.62	0.820	0.890
RUBRICEVO w/o REPLACE	76.63	56.18	0.950	1.050
RUBRICEVO w/o BRANCH	76.91	56.29	1.000	0.980
RUBRICEVO (Full)	**77.70**	**56.64**	**1.014**	**1.073**

5.4 ANALYSIS

5.4.1 DOES RUBRIC EVOLUTION SUSTAIN LEARNING SIGNALS?

We analyze reward-resolution changes around rubric updates across Writing, Chat, and Medical, normalizing each event by the mean of the two preceding audit windows. We compare retained and rolled-back patches to examine whether improvements persist after an update. To assess whether the resulting rewards distinguish response quality, we score the same responses with the original and evolved rubrics and compare their rankings with independent quality judgments. Blind human raters also choose which ranking better reflects response quality. Finally, we resume a plateauing Qwen3-1.7B Writing run from its step-100 checkpoint in two branches: one continues static graph-based training, while the other enables online evolution. We track WritingBench performance and BGE-M3 diversity, each normalized to its value at the start of training.

Figure 4(a) shows reward resolution recovering from 68.7% at the update window to 126.4% and 122.1% in the next two windows. Median post-/pre-update resolution ratios are $1.24\times$ for retained patches and $0.59\times$ for rolled-back patches (Figure 4(b)). Agreement with independent judgments rises from 66.9% to 73.8% (Figure 4(c)). Human raters prefer rankings from evolved rubrics in 54.4% of cases and original rubrics in 15.6%, with 30.0% ties. Figure 4(d) shows gradual, partial recovery after enabling evolution: by step 200, WritingBench rises from 75.58 to 76.18, versus 75.94 under static continuation. Diversity recovers from 0.910 to 0.952, still below its initial level but above static continuation at 0.890. A separate blind-voting study evaluates pre- and post-evolution rubric quality (Appendix A.2).

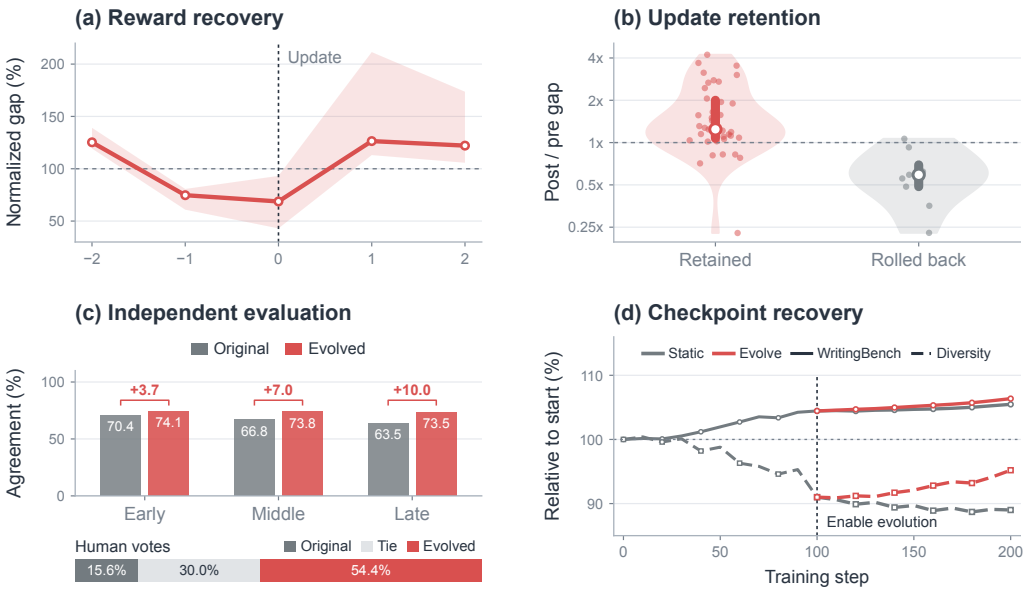

Figure 4: Reward diagnostics, independent evaluation, and recovery from a shared checkpoint.

These findings suggest that evolution helps maintain informative supervision. Reward-gap recovery accompanies better agreement with independent quality judgments, suggesting improved response ordering rather than merely reward rescaling. The retention analysis highlights the need to verify sustained improvements. Enabling evolution also yields modest performance gains and partial diversity recovery in the selected plateauing run. Together, these results support rubric evolution in sustaining reward discrimination and policy improvement.

5.4.2 Case Study: How Do Rubrics Evolve?

Finer-grained criteria. Rubric evolution can make an existing evaluation criterion more precise. In one example, the original rubric broadly assessed *"whether the causal analysis is reasonable."* As this criterion became less discriminative, it evolved to examine *"whether the proposed mechanism is concrete, whether its hypotheses are testable, and whether competing explanations can be distinguished."* This evolution directly refines the scoring criterion.

Emergent evaluation dimensions. Rubric evolution can also introduce evaluation dimensions that were not covered by the original rubric. In a science-fiction writing example, the initial criteria primarily examined *"whether the fictional technology and its bodily consequences are internally consistent."* During training, a new criterion emerged to assess *"whether the technological setting meaningfully serves character emotion and the narrative theme."* This extends the evaluation beyond technical consistency to the narrative function of the fictional technology.

6 Conclusion

We introduced RubricEvo, a framework for sustaining learning signals in open-ended reinforcement learning through graph-based, multi-level rubric evolution. RubricEvo organizes evaluation criteria into fixed principles and evolving aspect and specific rubrics, while using validated replacement and branching operations to restore reward resolution and preserve diverse response strategies. We additionally constructed 9,000 graph-annotated training instances across writing, dialogue, and medical assistance. Experiments with Qwen3-1.7B, 4B, and 8B show consistent improvements across all three domains, together with stronger reward discrimination and greater response diversity than static or flat evolving rubrics. Our analyses further demonstrate that retained updates recover deteriorating reward signals and that rubric evolution can both refine existing criteria and reveal previously uncovered evaluation dimensions.

AI USE STATEMENT

In this work, we used generative AI tools primarily for language polishing, data annotation, evaluation as LLM judges, and to assist with the preparation and refinement of figures and tables. All AI-assisted content was carefully reviewed, verified, and revised by the authors to ensure accuracy and consistency with the original research. Generative AI was not used to independently generate research results or draw scientific conclusions. We take full responsibility for the final content of this work, including any text, annotations, evaluations, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies rubric-based reinforcement learning for open-ended generation, including medical question answering. We use existing research datasets and do not collect new patient data, interact with patients, or perform clinical interventions. The generated medical responses are evaluated solely for research purposes and should not be interpreted as medical advice or deployed in clinical decision-making without appropriate expert oversight. Because automatic rubric evolution may introduce criteria that reward unsupported or potentially unsafe medical responses, medical safety requirements are retained as fixed Principle Rubrics throughout training, and candidate graph updates are rejected if they conflict with these requirements. Nevertheless, automated evaluators cannot guarantee clinical correctness or eliminate biases inherited from the underlying models and datasets; we therefore restrict our claims to research benchmarks and do not claim clinical readiness.

REFERENCES

Oron Anschel, Alon Shoshan, Adam Botach, Shunit Haviv Hakimi, Asaf Gendler, Emanuel Ben Baruch, Nadav Bhonker, Igor Kviatkovsky, Manoj Aggarwal, and Gerard Medioni. Group-aware reinforcement learning for output diversity in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 32382–32403, 2025.

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.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longwriter: Unleashing 10,000+ word generation from long context llms. In *International Conference on Learning Representations*, volume 2025, pp. 36528–36546, 2025.

Manish Bhattarai, Ismael Boureima, Nishath Rajiv Ranasinghe, Scott Pakin, and Dan O'Malley. Rubric-grounded rl: Structured judge rewards for generalizable reasoning. *arXiv preprint arXiv:2605.08061*, 2026.

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

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

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

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *arXiv preprint arXiv:2210.02410*, 2022.

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

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

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Pei Ke, Bosi Wen, Andrew Feng, Xiao Liu, Xuanyu Lei, Jiale Cheng, Shengyuan Wang, Aohan Zeng, Yuxiao Dong, Hongning Wang, et al. Critiquellm: Towards an informative critique generation model for evaluation of large language model generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13034–13054, 2024.

Seungone Kim, Jay Shin, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Ryan Shin, Sungdong Kim, James Thorne, Minjoon Seo, et al. Prometheus: Inducing fine-grained evaluation capability in language models. In *International Conference on Learning Representations*, volume 2024, pp. 29927–29962, 2024a.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4334–4353, 2024b.

Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, et al. The biggen bench: A principled benchmark for fine-grained evaluation of language models with language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5877–5919, 2025.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and diversity. In *International Conference on Learning Representations*, volume 2024, pp. 20620–20653, 2024.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, et al. Rewardbench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1755–1797, 2025.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, et al. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*, 2023.

Sunzhu Li, Jiale Zhao, Huimin Ren, Zhenlin Wei, Yang Zhou, Jingwen Yang, Shunyu Liu, Kaike Zhang, and Chen Wei. Rubrichub: A comprehensive and highly discriminative rubric dataset via automated coarse-to-fine generation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 31320–31344, 2026.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. In *International Conference on Learning Representations*, volume 2025, pp. 47852–47870, 2025.

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

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.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 2511–2522, 2023.

Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah Smith, Hanna Hajishirzi, and Nathan Lambert. Rewardbench 2: Advancing reward model evaluation. In *International Conference on Learning Representations*, volume 2026, pp. 144839–144866, 2026.

Tong Mu, Alec Helyar, Johannes Heidecke, Joshua Achiam, Andrea Vallone, Ian Kivlichan, Molly Lin, Alex Beutel, John Schulman, and Lilian Weng. Rule based rewards for language model safety. *Advances in Neural Information Processing Systems*, 37:108877–108901, 2024.

OpenAI. Introducing gpt-4.1 in the api. `https://openai.com/index/gpt-4-1/`, 2025.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Samuel J Paech. Eq-bench creative writing benchmark v3. `https://github.com/EQ-bench/creative-writing-bench`, 2025.

MohammadHossein Rezaei, Robert Vacareanu, Zihao Wang, Clinton Wang, Bing Liu, Yunzhong He, and Afra Feyza Akyürek. Online rubrics elicitation from pairwise comparisons. *arXiv preprint arXiv:2510.07284*, 2025.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G Finlayson, David Sontag, et al. Dr tulu: Reinforcement learning with evolving rubrics for deep research. *arXiv preprint arXiv:2511.19399*, 2025.

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

Leheng Sheng, Wenchang Ma, Ruixin Hong, Xiang Wang, An Zhang, and Tat-Seng Chua. Reinforcing chain-of-thought reasoning with self-evolving rubrics. *arXiv preprint arXiv:2602.10885*, 2026.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, et al. Paperbench: evaluating ai's ability to replicate ai research. *arXiv preprint arXiv:2504.01848*, 2025.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Haoran Sun, Yekun Chai, Shuohuan Wang, Yu Sun, Hua Wu, and Haifeng Wang. Curiosity-driven reinforcement learning from human feedback. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23517–23534, 2025.

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

Jianze Wang, Kunwang Zheng, Ying Liu, Yu Cao, Qilong Zhang, Jinlong Chen, Hua Yang, and Qianglong Chen. SERPO: Self-evolving rubric policy optimization for open-ended test-time reinforcement learning. *arXiv preprint arXiv:2607.26873*, 2026b. URL https://arxiv.org/abs/2607.26873.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason E Weston, and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving alignment with llm-as-a-meta-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 11548–11565, 2025.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Jin, et al. Writingbench: A comprehensive benchmark for generative writing. *Advances in Neural Information Processing Systems*, 38, 2026.

Zhiling Yan, Dingjie Song, Zhe Fang, Yisheng Ji, Xiang Li, Quanzheng Li, and Lichao Sun. Livemedbench: A contamination-free medical benchmark for llms with automated rubric evaluation. *arXiv preprint arXiv:2602.10367*, 2026.

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.

Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. Flask: Fine-grained language model evaluation based on alignment skill sets. In *International Conference on Learning Representations*, volume 2024, pp. 55361–55414, 2024.

Junkai Zhang, Zihao Wang, Lin Gui, Swarnashree Mysore Sathyendra, Jaehwan Jeong, Victor Veitch, Wei Wang, Yunzhong He, Bing Liu, and Lifeng Jin. Chasing the tail: Effective rubric-based reward modeling for large language model post-training. In *International Conference on Learning Representations*, volume 2026, pp. 133430–133457, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. In *International Conference on Learning Representations*, volume 2025, pp. 51257–51296, 2025.

A Appendix

A.1 More Related Work

Rubric-Based Evaluation and Reinforcement Learning. Preference-based alignment learns scalar rewards from human comparisons (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022). Constitutional AI and RLAIF introduce AI-generated supervision (Bai et al., 2022; Lee et al., 2023), while LLM-as-a-Judge enables scalable open-ended evaluation (Liu et al., 2023;

Zheng et al., 2023). Fine-grained evaluators further incorporate instruction-dependent skills, customized criteria, and critiques (Ye et al., 2024; Kim et al., 2024a; Zhu et al., 2025; Ke et al., 2024; Kim et al., 2024b). Instance-specific rubrics now support benchmark evaluation (Kim et al., 2025; Wu et al., 2026; Arora et al., 2025; Starace et al., 2025) and provide explicit rewards beyond verifiable domains (Mu et al., 2024; Gunjal et al., 2026; Zhang et al., 2026; Liu et al., 2026a).

A.2 Blind Evaluation of Evolved Rubrics

To examine whether online evolution improves rubric quality, we collect all accepted rubric updates across our experiments and reconstruct the complete graph-based rubric immediately before and after each update. We present each pair to `GPT-5.6-Terra` and ask it to select the rubric graph that provides the more appropriate and discriminative evaluation criteria, with ties allowed. The judge receives neither version labels nor training information. To control for positional bias, we repeat the evaluation after reversing the presentation order of the two graphs.

Table 4: Blind comparison between rubric graphs before and after evolution.

Presentation Order	Evolved Better	Tie	Original Better
Original first, evolved second	71.17%	14.26%	14.56%
Evolved first, original second	81.72%	11.89%	6.39%

As shown in Table 4, the evolved rubric graph is preferred under both presentation orders, indicating that the observed improvement is robust to positional bias.

A.3 Effective Semantic-Mode Analysis

To complement the pairwise diversity analysis in Figure 3, we examine whether repeated generations occupy multiple semantic modes. For each benchmark query, we sample 16 responses from each method under identical decoding settings and encode them with BGE-M3. Given the normalized embedding similarity matrix K, we compute the threshold-free effective number of modes as $N_{\text{eff}} = \exp(-\sum_k \lambda_k \log \lambda_k)$, where $\{\lambda_k\}$ are the eigenvalues of $K/\operatorname{tr}(K)$. Unlike hard clustering, this measure does not require a manually selected distance threshold and can represent partially overlapping or imbalanced modes. We average N_{eff} across queries within each benchmark.

As shown in Table 5, RubricEvo achieves the highest average effective-mode count at all three model scales and the best result on 14 of the 15 backbone–benchmark combinations. Relative to the corresponding base model, its average mode coverage increases by 6.5%, 10.0%, and 12.1% for Qwen3-1.7B, 4B, and 8B, respectively. In contrast, RaR consistently reduces the effective number of modes, while EvoRubrics remains close to the base model. These results are consistent with the normalized diversity trends in Figure 3 and indicate that the diversity retained by RubricEvo is accompanied by broader semantic-mode coverage rather than only larger pairwise embedding distances.

A.4 Analysis of Reward Resolution and Semantic Diversity

This subsection derives the mechanisms discussed in Section 4.1. We fix a query x and suppress its dependence where unambiguous. All expectations are conditional on this query. We first examine how saturation reduces the reliability of group-relative rewards, and then analyze how persistent reward preferences can concentrate the response distribution.

A.4.1 Reward Saturation and Loss of Reliable Resolution

Reward model and resolution. Let $Y \sim \pi_{\theta_t}(\cdot \mid x)$ denote a response at training step t. After rescaling criterion scores to $[0, 1]$, a fixed rubric defines the noise-free reward

$$R_t = r(Y) = \sum_{j=1}^{M} w_j s_j(Y), \qquad w_j \geq 0, \qquad \sum_{j=1}^{M} w_j = 1. \tag{1}$$

Table 5: Effective number of semantic response modes across five benchmarks. Higher values indicate that repeated generations are distributed across more distinct semantic modes. Avg. is the macro-average across benchmarks, and Rel. Avg. normalizes this value by the corresponding base model. The best result within each backbone and benchmark is shown in **bold**.

Method	Health Bench	LMB-EN	Writing Bench	Creative Writing v3	WildBench v2	Avg.	Rel. Avg.
Qwen3-1.7B Backbone							
Base Model	**1.461**	1.146	1.735	2.159	1.781	1.656	1.000
RaR	1.344	1.066	1.579	1.922	1.639	1.510	0.911
EvoRubrics	1.432	1.158	1.718	2.116	1.799	1.645	0.994
RUBRICEVO (Ours)	1.420	**1.237**	**1.934**	**2.335**	**1.892**	**1.764**	**1.065**
Qwen3-4B Backbone							
Base Model	1.503	1.184	1.806	2.284	1.842	1.724	1.000
RaR	1.353	1.089	1.607	2.010	1.658	1.543	0.894
EvoRubrics	1.488	1.208	1.824	2.261	1.879	1.732	1.005
RUBRICEVO (Ours)	**1.548**	**1.303**	**2.054**	**2.558**	**2.012**	**1.895**	**1.100**
Qwen3-8B Backbone							
Base Model	1.537	1.216	1.861	2.376	1.901	1.778	1.000
RaR	1.368	1.107	1.638	2.067	1.692	1.574	0.883
EvoRubrics	1.552	1.240	1.898	2.400	1.939	1.806	1.016
RUBRICEVO (Ours)	**1.614**	**1.363**	**2.158**	**2.714**	**2.119**	**1.994**	**1.121**

Thus, $R_t \in [0, 1]$. Here, $r(Y)$ denotes quality captured by the rubric, rather than an assumption that the rubric perfectly represents task quality. We model the observed evaluator output as $\widehat{R}_t = R_t + \eta$, where η has zero mean and variance $\sigma_\eta^2 > 0$. For the following calculations, evaluator errors are independent of the responses and independent across response evaluations.

For $G \geq 2$ independently sampled responses, the observed reward-resolution statistic is

$$\widehat{\Delta}_t = \frac{2}{G(G-1)} \sum_{i<j} |\hat{r}_i - \hat{r}_j| \,. \tag{2}$$

Its noise-free population counterpart is

$$\Delta_t^\star = \mathbb{E}\left[|R_t - R_t'|\right], \tag{3}$$

where R_t' is an independent copy of R_t.

From saturation to contracting reward gaps. Write $\mu_t = \mathbb{E}[R_t]$ and $\sigma_t^2 = \mathrm{Var}(R_t)$. Because $0 \leq R_t \leq 1$, we have $R_t^2 \leq R_t$, and therefore

$$\sigma_t^2 = \mathbb{E}[R_t^2] - \mu_t^2 \leq \mu_t(1 - \mu_t). \tag{4}$$

Consequently, if the policy increasingly satisfies the fixed rubric so that $\mu_t \to 1$, then $\sigma_t^2 \to 0$. By the Cauchy–Schwarz inequality and independence of R_t and R_t',

$$\begin{aligned} \Delta_t^\star &= \mathbb{E}\left[|R_t - R_t'|\right] \\ &\leq \sqrt{\mathbb{E}\left[(R_t - R_t')^2\right]} \\ &= \sqrt{2\sigma_t^2} \leq \sqrt{2\mu_t(1-\mu_t)} \longrightarrow 0. \end{aligned} \tag{5}$$

Approaching the reward ceiling is therefore a sufficient condition for resolution loss. More generally, the same conclusion holds whenever the rubric-captured variance contracts toward zero, even if rewards concentrate below the ceiling.

Reliability after normalization. Consider the population-normalized noise-free and observed advantages

$$A_t^\star = \frac{R_t - \mu_t}{\sigma_t}, \qquad \widehat{A}_t = \frac{R_t + \eta - \mu_t}{\sqrt{\sigma_t^2 + \sigma_\eta^2}}, \tag{6}$$

defined for $\sigma_t > 0$. Independence gives

$$\mathrm{Cov}(R_t + \eta, R_t) = \sigma_t^2, \qquad \mathrm{Var}(R_t + \eta) = \sigma_t^2 + \sigma_\eta^2. \tag{7}$$

Since correlation is invariant to positive affine transformations,

$$\begin{aligned} \mathrm{Corr}(\widehat{A}_t, A_t^\star) &= \frac{\sigma_t^2}{\sqrt{\sigma_t^2 + \sigma_\eta^2}\,\sigma_t} \\ &= \sqrt{\frac{\sigma_t^2}{\sigma_t^2 + \sigma_\eta^2}}. \end{aligned} \tag{8}$$

Thus, when $\sigma_t^2/\sigma_\eta^2 \to 0$, the correlation approaches zero. Normalization changes the numerical scale of the advantage but does not restore the information lost to evaluator noise.

For completeness, actual GRPO uses sample statistics within each response group. Define $\delta_i = r_i - \bar{r}$, $\tilde{\eta}_i = \eta_i - \bar{\eta}$, and

$$V_G = \frac{1}{G}\sum_i \delta_i^2, \qquad N_G = \frac{1}{G}\sum_i \tilde{\eta}_i^2, \qquad C_G = \frac{1}{G}\sum_i \delta_i \tilde{\eta}_i. \tag{9}$$

When both reward vectors have nonzero variance, their empirical advantage correlation is exactly

$$\mathrm{Corr}_G(\hat{A}, A^\star) = \frac{V_G + C_G}{\sqrt{V_G}\sqrt{V_G + N_G + 2C_G}}. \tag{10}$$

This identity also holds with the positive numerical stabilizer used in GRPO, because each advantage vector is scaled by a single positive group-level constant. Independence implies zero population covariance, but a finite sample need not have $C_G = 0$. Under independent sampling and finite second moments, Equation equation 10 converges to Equation equation 8 as G increases.

Pairwise ordering under noise. A direct interpretation follows from response ordering. Suppose the evaluation errors are independent Gaussian variables, $\eta_i \sim \mathcal{N}(0, \sigma_\eta^2)$. For two responses with $r_i > r_j$, let $g_{ij} = r_i - r_j > 0$. Then

$$\hat{r}_i - \hat{r}_j = g_{ij} + (\eta_i - \eta_j), \qquad \eta_i - \eta_j \sim \mathcal{N}(0, 2\sigma_\eta^2). \tag{11}$$

Writing Φ for the standard normal cumulative distribution function, the probability of preserving the correct ordering is

$$\Pr(\hat{r}_i > \hat{r}_j \mid r_i, r_j) = \Phi\left(\frac{g_{ij}}{\sqrt{2}\sigma_\eta}\right). \tag{12}$$

As $g_{ij}/\sigma_\eta \to 0$, this probability approaches $1/2$. Group normalization preserves exactly the same ordering:

$$\hat{A}_i - \hat{A}_j = \frac{\hat{r}_i - \hat{r}_j}{\mathrm{Std}(\hat{r}_1, \ldots, \hat{r}_G) + \epsilon}. \tag{13}$$

It therefore cannot repair an ordering already corrupted by noise. If all observed rewards are tied, every group-relative advantage is zero.

Observed reward gaps need not vanish even when meaningful differences disappear. Under the same Gaussian model, as R_t concentrates at a constant,

$$\mathbb{E}[\widehat{\Delta}_t] \longrightarrow \mathbb{E}[|\eta - \eta'|] = \frac{2\sigma_\eta}{\sqrt{\pi}}. \tag{14}$$

Accordingly, reward resolution is an observable diagnostic, while its interpretation as a useful learning signal depends on the size of meaningful gaps relative to evaluator noise.

A.4.2 Persistent Reward Preferences and Diversity Contraction

Semantic diversity and response modes. Let $e(y)$ be the normalized BGE-M3 embedding of response y, with $\|e(y)\|_2 = 1$. The implementation measures semantic diversity as the mean pairwise cosine distance:

$$\widehat{D}_{\mathrm{sem},t} = \frac{2}{G(G-1)} \sum_{i<j} \left(1 - e(y_i)^\top e(y_j)\right). \tag{15}$$

For independent responses $Y, Y' \sim \pi_{\theta_t}(\cdot \mid x)$, its population counterpart is

$$D_{\mathrm{sem},t} = \mathbb{E}\left[1 - e(Y)^\top e(Y')\right], \qquad \mathbb{E}[\widehat{D}_{\mathrm{sem},t}] = D_{\mathrm{sem},t}. \tag{16}$$

Suppose responses can be partitioned into K strategy modes, with $Z(Y) \in \{1, \ldots, K\}$ and $p_k^{(t)} = \Pr(Z(Y) = k)$. Define the conditional pairwise distance

$$d_{k\ell}^{(t)} = \mathbb{E}\left[1 - e(Y)^\top e(Y') \mid Z(Y) = k,\ Z(Y') = \ell\right]. \tag{17}$$

The law of total expectation gives the exact decomposition

$$D_{\mathrm{sem},t} = \sum_{k=1}^{K} \sum_{\ell=1}^{K} p_k^{(t)} p_\ell^{(t)} d_{k\ell}^{(t)}. \tag{18}$$

In particular, $d_{kk}^{(t)}$ need not be zero: responses following the same strategy can still differ semantically. To isolate changes in strategy frequencies, the analysis below holds the conditional response distributions fixed over the interval considered, so that both $d_{k\ell}$ and the strategy-level expected rewards $\mu_k = \mathbb{E}[r(Y) \mid Z(Y) = k]$ remain constant.

A local policy-update model. We analyze a KL-regularized update over strategy probabilities. This provides a tractable model of reward-driven redistribution across modes; it is not an exact identity for clipped, token-level GRPO. Let $\alpha > 0$ denote the update size, let $\lambda \geq 0$ control a fixed-reference KL penalty, and assume that both $p^{(t)}$ and p^{ref} have positive entries. The update is

$$p^{(t+1)} = \arg\max_{q \in \mathcal{P}_K} \left\{ \sum_{k=1}^{K} q_k \mu_k - \frac{1}{\alpha} D_{\mathrm{KL}}(q \| p^{(t)}) - \lambda D_{\mathrm{KL}}(q \| p^{\mathrm{ref}}) \right\}, \tag{19}$$

where $\mathcal{P}_K = \{q : q_k \geq 0,\ \sum_k q_k = 1\}$. Introducing a multiplier ν for the normalization constraint, the first-order condition is

$$\mu_k - \frac{1}{\alpha}\left(\log \frac{q_k}{p_k^{(t)}} + 1\right) - \lambda\left(\log \frac{q_k}{p_k^{\mathrm{ref}}} + 1\right) + \nu = 0. \tag{20}$$

Solving and normalizing yields

$$p_k^{(t+1)} = \frac{\left(p_k^{(t)}\right)^{\frac{1}{1+\alpha\lambda}} \left(p_k^{\mathrm{ref}}\right)^{\frac{\alpha\lambda}{1+\alpha\lambda}} \exp\left(\frac{\alpha\mu_k}{1+\alpha\lambda}\right)}{\sum_{\ell=1}^{K} \left(p_\ell^{(t)}\right)^{\frac{1}{1+\alpha\lambda}} \left(p_\ell^{\mathrm{ref}}\right)^{\frac{\alpha\lambda}{1+\alpha\lambda}} \exp\left(\frac{\alpha\mu_\ell}{1+\alpha\lambda}\right)}. \tag{21}$$

Accumulation of a persistent preference. Consider two valid strategies with probabilities p_t and $1 - p_t$. Suppose the fixed rubric assigns them a persistent expected reward difference $\gamma = \mu_1 - \mu_2 > 0$. Let $\ell_t = \log(p_t/(1-p_t))$ and define ℓ_{ref} analogously. Taking the ratio in Equation equation 21 gives

$$\ell_{t+1} = \frac{\ell_t + \alpha\gamma + \alpha\lambda\ell_{\mathrm{ref}}}{1 + \alpha\lambda}. \tag{22}$$

Without a fixed-reference penalty, $\lambda = 0$, repeated updates satisfy

$$\ell_t = \ell_0 + t\alpha\gamma, \qquad p_t = \frac{1}{1 + \exp(-\ell_0 - t\alpha\gamma)} \longrightarrow 1. \tag{23}$$

The local KL term limits each individual update, but it does not prevent a persistent reward difference from accumulating across updates. The strategy entropy consequently satisfies

$$H_t = -p_t \log p_t - (1 - p_t) \log(1 - p_t) \longrightarrow 0. \tag{24}$$

A fixed-reference penalty changes this limit. For $\lambda > 0$, write $a = (1 + \alpha\lambda)^{-1}$. Solving Equation equation 22 gives

$$\ell_t = a^t \ell_0 + (1 - a^t) \left(\ell_{\text{ref}} + \frac{\gamma}{\lambda} \right), \qquad p_t \longrightarrow \frac{1}{1 + \exp\left(-\ell_{\text{ref}} - \gamma/\lambda\right)}. \tag{25}$$

Thus, reference regularization limits concentration. A large reward difference relative to λ can still strongly suppress the less-rewarded strategy, but a finite positive reference penalty does not imply complete mode collapse.

Consequences for the measured diversity. To make the connection to Equation equation 15 explicit, suppose the two modes have equal within-mode expected distance $d_{11} = d_{22} = d_{\text{in}}$ and a larger cross-mode distance $d_{12} = d_{21} = d_{\text{out}} > d_{\text{in}}$. Equation equation 18 becomes

$$\begin{aligned} D_{\text{sem},t} &= \left[p_t^2 + (1 - p_t)^2 \right] d_{\text{in}} + 2p_t(1 - p_t) d_{\text{out}} \\ &= d_{\text{in}} + 2p_t(1 - p_t) \left(d_{\text{out}} - d_{\text{in}} \right). \end{aligned} \tag{26}$$

If $p_t \to 1$, then $D_{\text{sem},t} \to d_{\text{in}}$: the contribution from variation between strategies disappears, while within-strategy variation remains. With a positive reference penalty, the limiting diversity is obtained by substituting the nondegenerate probability from Equation equation 25 into Equation equation 26.

The direction of change follows from

$$\frac{\partial D_{\text{sem}}}{\partial p} = 2(1 - 2p) \left(d_{\text{out}} - d_{\text{in}} \right), \qquad \frac{\partial H}{\partial p} = \log \frac{1 - p}{p}. \tag{27}$$

Once the favored strategy has probability $p > 1/2$, further concentration decreases both strategy entropy and semantic diversity. Meanwhile, its expected reward increases because

$$\bar{r}(p) = p\mu_1 + (1 - p)\mu_2, \qquad \frac{\partial \bar{r}}{\partial p} = \gamma > 0. \tag{28}$$

Increasing mean reward can therefore coexist with declining diversity. If the favored strategy initially has probability below $1/2$, diversity may first increase as the mixture becomes more balanced and decline only afterward. The derivation establishes a mechanism for diversity contraction under persistent reward preferences; it does not require diversity to decrease at every training step.

A.5 Statistical Significance

We assess statistical significance using paired bootstrap over instance-level evaluation scores. For each backbone and benchmark, we compute the score difference between RubricEvo and the strongest baseline on every shared test instance, and construct a 95% confidence interval from 10,000 bootstrap resamples of these paired differences. As shown in Table 6, all confidence intervals exclude zero, indicating that the improvements of RubricEvo are statistically significant across all evaluated backbones and benchmarks.

A.6 Details of Datasets

A.6.1 Data Selection

Writing. We collect writing prompts from the training split of LongWriter-6k (Bai et al., 2025). Starting from 6,000 raw instances, we extract the first user request and remove empty or duplicate queries, resulting in 5,846 unique candidates. We review these candidates for linguistic quality, self-containment, task clarity, and suitability for open-ended evaluation. Both English and Chinese prompts are retained, while malformed, overly constrained, or weakly evaluable requests are discarded.

Table 6: Statistical significance of the improvements achieved by RUBRICEVO over the strongest baseline. Each entry reports the mean paired improvement Δ and the half-width of its 95% confidence interval, estimated from 10,000 paired-bootstrap resamples.

Backbone	HealthBench	LMB-EN	WritingBench	Creative Writing	WildBench
Qwen3-1.7B	$+3.71 \pm 0.63$	$+2.23 \pm 0.41$	$+2.38 \pm 0.71$	$+0.67 \pm 0.38$	$+1.04 \pm 0.56$
Qwen3-4B	$+2.62 \pm 0.52$	$+1.60 \pm 0.47$	$+2.14 \pm 0.58$	$+1.71 \pm 0.44$	$+0.51 \pm 0.29$
Qwen3-8B	$+2.96 \pm 0.69$	$+1.14 \pm 0.36$	$+1.93 \pm 0.64$	$+1.08 \pm 0.51$	$+1.06 \pm 0.43$

Chat. We source general conversational prompts from the Chat split of RubricHub (Li et al., 2026). After normalization and deduplication, its 9,812 source instances yield 9,801 unique candidates. We retain self-contained, open-ended prompts that support substantive and potentially diverse responses, while excluding retrieval-dependent, template-like, unsafe, or weakly evaluable examples.

Medical. We construct the medical subset from the training split of RaR-Medicine (Gunjal et al., 2026). From its 17,926 source instances, we remove duplicate questions, malformed examples, closed-form multiple-choice problems, prompts containing answer leakage, and tasks that can be answered only with an option label or a single word. We retain self-contained prompts requiring explanatory, advisory, or clinically grounded open-ended responses.

A.6.2 RUBRIC GRAPH ANNOTATION

For Chat and Medical, we transform the query-specific flat rubrics provided by the source datasets into rubric graphs. Since LongWriter-6k does not provide rubrics, we directly annotate a rubric graph for each selected writing prompt. We distinguish immutable core criteria that capture shared requirements from mutable frontier criteria that differentiate strong responses, and annotate criterion dependencies, alternative strategy branches, and permissible evolution operations. Each node contains its criterion description and scoring guidance, while the graph specifies which criteria are universally applicable, conditionally activated, or eligible for subsequent online evolution.

A.6.3 QUALITY CONTROL AND STATISTICS

Rubric graphs are annotated by `GPT-5.5` following a unified, human-authored protocol. During the initial annotation pass, 15.8% of the Writing candidates, 35.6% of the Chat candidates, and 35.4% of the Medical candidates are rejected for failing the semantic or structural requirements of the protocol. We further apply automatic validation for schema completeness, scoring-scale validity, duplicated criteria, invalid node types, cyclic structures, and dangling dependencies.

For final quality assessment, we randomly sample 300 annotations from each domain for stratified review. Human reviewers retain 89% of the Writing annotations and 92% of the Chat annotations. For Medical, a context-isolated instance of `GPT-5.5`, with no access to the original annotation context, independently performs the review and retains 84%. Rejected annotations are revised or replaced before inclusion. The resulting dataset contains 9,000 query–rubric-graph pairs, with 3,000 instances in each domain.

A.7 IMPLEMENTATION DETAILS.

The principal training settings and hyperparameters are summarized in Table 7.

For Qwen3-8B, the combination of ZeRO-3, activation checkpointing, smaller per-device batches, gradient accumulation, micro-batched rollouts, and timely release of temporary tensors enables full-parameter GRPO training on four A100 GPUs.

Table 7: Implementation details and training hyperparameters used in our experiments.

Hyperparameter	Setting
General Configuration	
Backbones	Qwen3-1.7B, Qwen3-4B, and Qwen3-8B, with thinking mode disabled.
Hardware	$4\times$ NVIDIA A100 GPUs with 80 GB memory per GPU.
Training framework	TRL 0.19.1 with DeepSpeed distributed training.
Training algorithm	Full-parameter GRPO with batch-normalized policy optimization (`loss_type=bnpo`).
Numerical precision	BF16.
Random seed	2026.
Optimization	
Optimizer	AdamW.
Learning rate	2×10^{-6}.
Learning-rate scheduler	Constant.
KL coefficient	$\beta = 0.01$.
GRPO clipping ratio	0.2.
Maximum gradient norm	1.0.
Responses per prompt	4.
Effective prompts per step	8, corresponding to 32 sampled responses per optimizer step.
Per-device batch size	4 for Qwen3-1.7B/4B and 2 for Qwen3-8B.
Gradient accumulation	2 steps for Qwen3-1.7B/4B and 4 steps for Qwen3-8B.
Maximum prompt length	4,096 tokens.
Maximum response length	6,144 tokens.
Sampling configuration	Temperature 1.0 and top-$p = 1.0$.
Distributed Training and Memory Management	
Qwen3-1.7B/4B	DeepSpeed ZeRO-2, which partitions optimizer states and gradients across the four GPUs while replicating model parameters.
Qwen3-8B	DeepSpeed ZeRO-3, which partitions model parameters, gradients, and optimizer states across the four GPUs.
Activation checkpointing	Enabled for Qwen3-8B to avoid retaining all intermediate activations during the forward pass.
Attention cache	Disabled during policy optimization to reduce persistent GPU memory consumption.
Rollout processing	Policy responses are generated and scored in memory-bounded micro-batches rather than retaining all rollout intermediates simultaneously.
Retained training tensors	Only token IDs, attention masks, old and reference log-probabilities, and scalar rewards required by the GRPO objective are retained for policy optimization.
Temporary tensor release	Decoded responses, generation outputs, intermediate logits, hidden states, and temporary reward tensors are released immediately after the corresponding rollout or scoring stage.
Memory cleanup	Python garbage collection and CUDA-cache cleanup are performed between rollout generation, reward computation, and policy optimization.
Checkpoint writing	Checkpoints are consolidated and written by the main process to avoid redundant model copies on different ranks.
Checkpoint interval	Every 50 optimizer steps.
Checkpoint contents	Model parameters, optimizer and scheduler states, random-number generator states, and the current rubric graphs.
Evaluation Models	
Reward evaluator	GPT-4.1-mini with temperature zero.
Rubric evolver	GPT-4o with temperature zero.
Semantic encoder	BGE-M3.

A.8 PROMPT

At each evolution event, the external evolver receives the complete active rubric graph, graph-health statistics, the trigger type, and representative near-tie responses. It may modify one active muta-

Table 8: General system instructions for rubric evolution.

Component	Instruction
Role	You are the external Rubric Evolver in RUBRICEVO. The policy model is separate from you. Modify the evaluation topology, not the policy or optimizer.
Scope	Use only an atomic REPLACE or BRANCH operation on one active mutable rubric. Never modify a Principle Rubric or rewrite the complete rubric.
Required context	Before proposing a patch, inspect the query, the target rubric, every other active rubric, the trigger type, and the supplied near-tie responses.
Semantic preservation	Preserve every essential requirement, hard constraint, safety condition, and factual obligation expressed by the target rubric. Evolution may sharpen discrimination but must not delete required coverage, relax a necessary condition, or introduce an unrelated aspect.
Full-graph consistency	Check the complete active graph. A new rubric or branch child must not duplicate, contradict, or merely paraphrase a Principle Rubric, another active rubric, or another child in the same branch.
Evidence grounding	Base every modification on concrete differences among the supplied responses. Do not invent distinctions absent from the evidence merely to produce a candidate.
Anti-shortcut rule	Do not reward verbosity, formatting tricks, unnecessary detail, a prescribed writing style, or superficial lexical differences. Evaluate semantic execution quality and task fulfillment.
REPLACE semantics	REPLACE retires one saturated mutable rubric and introduces one strictly sharper rubric. The replacement must refine rather than redefine the target and must retain its essential scope and obligations.
REPLACE evidence	The rationale must identify the preserved obligations, the concrete response pair that should be separated, and why the new scoring bands produce the intended ordering. A broad but necessary rubric must not be narrowed to one optional subcase.

ble rubric through an atomic REPLACE or BRANCH operation. Tables 8–10 present the complete instruction template and output requirements used in our implementation.

Table 9: Validity requirements for a BRANCH operation.

Requirement	Instruction
Branch semantics	BRANCH retires one mutable rubric and introduces two OR-connected children. Because only the higher-scoring child contributes to the reward, the children must represent genuinely substitutable and quality-comparable strategies for satisfying the same original evaluation aspect.
Substitution	A complete, high-quality response must be able to follow either child without satisfying the other and without becoming incomplete, incorrect, unsafe, or inconsistent with the query.
Non-complementarity	If a strong response should normally satisfy both children, they are complementary requirements and must not be represented as alternative branches.
Non-dominance	Neither child may be merely more comprehensive, detailed, cautious, specific, or higher quality than the other. Such alternatives represent different quality levels rather than different strategies.
Correctness	Do not branch competing factual answers, diagnoses, legal conclusions, or other outcomes when the query admits a single correct conclusion. Different explanations of the same correct conclusion may be branched only when both independently realize the same evaluation aspect.
Coverage preservation	Each child, independently at a score of 9–10, must preserve every essential obligation of the retired rubric. Max-pooling must not create a route around a required constraint.
Comparability	Both children must evaluate the same scope at comparable difficulty on the same 1–10 scale. Each child evaluates execution quality within its strategy rather than rewarding the presence or identity of that strategy.
Response evidence	The supplied responses must contain a quality-matched pair instantiating the two strategies. The rationale must explain why they deserve similar rewards and identify a within-strategy pair whose execution quality should remain ordered.
Fallback rule	If any requirement above fails, do not propose BRANCH for that target; use a semantics-preserving REPLACE. In particular, do not branch complementary dimensions, separate checklist items, a minimal versus comprehensive answer, or a correct versus incorrect conclusion.

Table 10: Dynamic inputs and required output schema for the rubric evolver.

Field	Specification
Graph state	The complete active rubric graph, including Principle Rubrics, mutable rubrics, existing alternative branches, and relevant graph metadata.
Health evidence	Reward-resolution and diversity statistics associated with the current audit window.
Trigger type	A resolution-only trigger permits only REPLACE. Diversity or joint triggers permit BRANCH only when all branch-validity requirements are satisfied; otherwise, the evolver must use REPLACE.
Response evidence	Representative high-quality near-tie responses selected from the current policy.
Candidate request	Return up to K substantially different and semantically valid candidates targeting an active mutable rubric with low group-level discrimination.
Common fields	Every candidate specifies an operation, a target-rubric identifier, and an evidence-based rationale describing the predicted response ordering.
REPLACE fields	A replacement candidate contains one rubric with a short name, precise description, weight, provenance, and complete scoring guidance.
BRANCH fields	A branch candidate contains exactly two child rubrics satisfying all requirements in Table 9.
Scoring schema	Every new rubric contains all five scoring bands: 1–2, 3–4, 5–6, 7–8, and 9–10. The bands must be monotonic, mutually coherent, grounded in observable response properties, and sufficiently specific to distinguish nearby strong responses. The top band must remain attainable without unnecessary length or stylistic conformity.
Required rationale	Every rationale identifies the supporting evidence, the preserved semantic contract, and why the selected operation is appropriate. A branch rationale additionally states that its children are substitutable rather than complementary and that each child independently preserves the target rubric's required coverage.
Serialization	Return one JSON object conforming to the supplied output schema and no additional text.