CoRT: Counterfactual Replay for Token-Level Rubric-Guided Policy Optimization

Anonymous authors
Paper under double-blind review

Abstract

Rubric-based reinforcement learning enriches language model training by evaluating model outputs against explicit criteria. Yet in GRPO-style pipelines, these structured judgments are reduced to a scalar response-level reward and converted into a response-level advantage, which is broadcast uniformly to all generated tokens. This leaves no explicit mechanism for allocating credit within a response, even when different criteria are grounded in different spans, formatting decisions, or semantic choices. We propose CoRT, a token-level credit weighting method for rubric-conditioned GRPO. Instead of training an auxiliary token scoring model, CoRT uses counterfactual replay to rescore the same sampled response under the original rubric-conditioned prompt and a matched criteria-free prompt. The resulting tokenwise log-likelihood contrasts serve as a proxy for dependence on the rubric context. CoRT maps these contrasts to bounded, response-normalized weights and uses them to redistribute the signed GRPO advantage across tokens, without introducing an auxiliary scorer or changing the response-level reward. Experiments across instruction-tuned models and reward granularities show that CoRT improves over matched response-level GRPO in the vast majority of comparisons, with an average gain of 4.4 percentage points. The method remains competitive with learned token-level credit baselines while avoiding a separate relevance-learning stage. These results suggest that policy-internal counterfactual likelihood contrasts provide an effective training signal for within-response credit allocation while retaining the simplicity and stability of GRPO.

1 Introduction

Reinforcement learning (RL) has become a central component of large language model post-training, enabling models to optimize behaviors that are difficult to capture with supervised fine-tuning alone (Ouyang et al., 2022; Jaech et al., 2024; Guo et al., 2025). This is especially important for open-domain instruction following, where user requests often combine semantic goals with diverse surface and behavioral constraints, and many tasks do not have a single verifiable answer. In many recent RL pipelines, a verifier or reward model assigns a scalar score to each sampled response, and policy optimization methods such as GRPO (Shao et al., 2024) update the model using response-level advantages. This formulation is simple and scalable, but it treats a response as a single unit of credit assignment, even when different tokens may contribute to different requirements.

Rubric-based RL provides a more structured source of supervision for these settings. Rather than judging a response only by an overall score, a rubric decomposes response behavior into explicit criteria, such as correctness, formatting, or safety (Gunjal et al., 2025; Li et al., 2026). However, in GRPO-style training, these structured judgments are aggregated into a single advantage before optimization and applied uniformly to all generated tokens. The supervision is structured at the criterion level, while the policy update remains uniform at the token level, leaving potential attribution information encoded in the rubric underused. A natural way to address this mismatch is to move from response-level reward assignment to token-level credit assignment. In principle, token-level signals can differentiate tokens critical to fulfilling rubric criteria from incidental filler tokens within the final response. One route is to learn an auxiliary token-level relevance or scoring model, but this introduces a separate training stage and an additional component that must be integrated into the RL pipeline. This raises a simpler question: *Can token-level credit be derived directly from policy-internal signals, without training a separate relevance estimator?*

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.

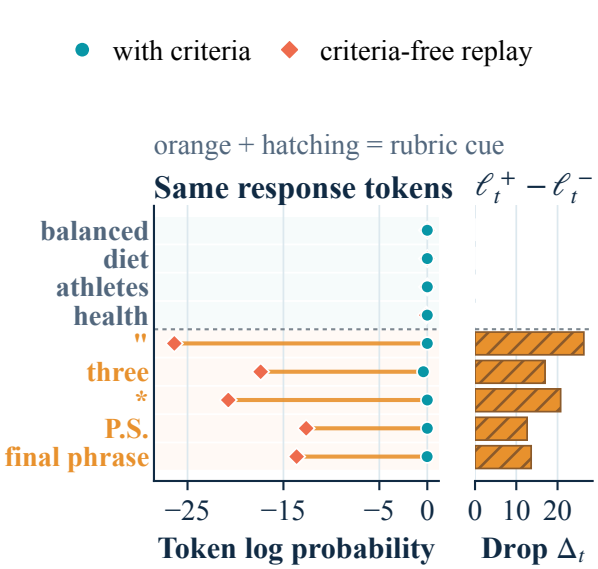

Figure 1: Real replay example. The same response is rescored with criteria and under a criteria-free replay. Generic content tokens show little log-probability change, while rubric-controlled tokens (orange) drop more after criteria removal.

Fixed-response replay exposes such a signal. If the same sampled response is rescored after removing the rubric criteria, tokens whose likelihood depends on these criteria tend to change more than generic content tokens. Figure 1 illustrates this effect on a real example. Because the response tokens are held fixed, the contrast isolates how the policy's likelihood for the observed trajectory changes under the criteria-free prompt. The highlighted tokens are not labeled by an external annotator; they are exposed by the policy's own response to this local counterfactual prompt change.

Motivated by this observation, we propose CoRT (**Co**unterfactual **R**eplay for **T**oken-level credit weighting), a token-level credit weighting method for rubric-conditioned GRPO. Here, counterfactual replay denotes a fixed-response intervention inspired by replay and counterfactual data augmentation in RL (Andrychowicz et al., 2017; Pitis et al., 2020): sampled tokens are held fixed and rescored after removing rubric criteria, yielding policy-internal likelihood contrasts rather than relabeled trajectories or new data. CoRT converts this contrast into bounded, response-normalized credit weights. These weights redistribute the original response-level GRPO advantage across tokens, so that rubric-dependent tokens receive more of the signed update signal without changing its direction. Unlike methods that train a token-level relevance model, CoRT requires no separate relevance-learning stage and does not change the response-level reward. By modifying only the within-response credit allocation while preserving response-level supervision, CoRT improves training effectiveness while preserving the simplicity of the GRPO.

2 Related Work

Instruction following and rubric feedback. Instruction following has been widely studied as a central goal of language model post training, with prior work improving model alignment through curated instruction data, instruction synthesis, and preference optimization (Zhou et al., 2023a; Xu et al., 2024; Lou et al., 2024; Rafailov et al., 2023). Recent benchmarks increasingly emphasize explicit constraints and fine-grained instruction diversity (Zhou et al., 2023b; Pyatkin et al., 2026; Gu et al., 2024), as well as multidimensional criteria and judge-based assessment (Ye et al., 2025; He et al., 2025). This evaluation paradigm has further motivated rubric and verifier based training methods, which extend RLHF beyond generic preference labels by providing more structured and inspectable supervision (Peng et al., 2025; Sun et al., 2024; He et al., 2024; Zhang et al., 2025). However, in many RL formulations, such structured feedback is ultimately reduced to a response level scalar reward or advantage. As a result, the criterion level information provided by rubrics or verifiers is not directly used to determine which parts of a response contribute to satisfying or violating specific requirements. Motivated by this gap, our work studies how criterion level rubric feedback can be used for within response credit assignment, allowing structured supervision to affect token level policy updates rather than only response level scoring.

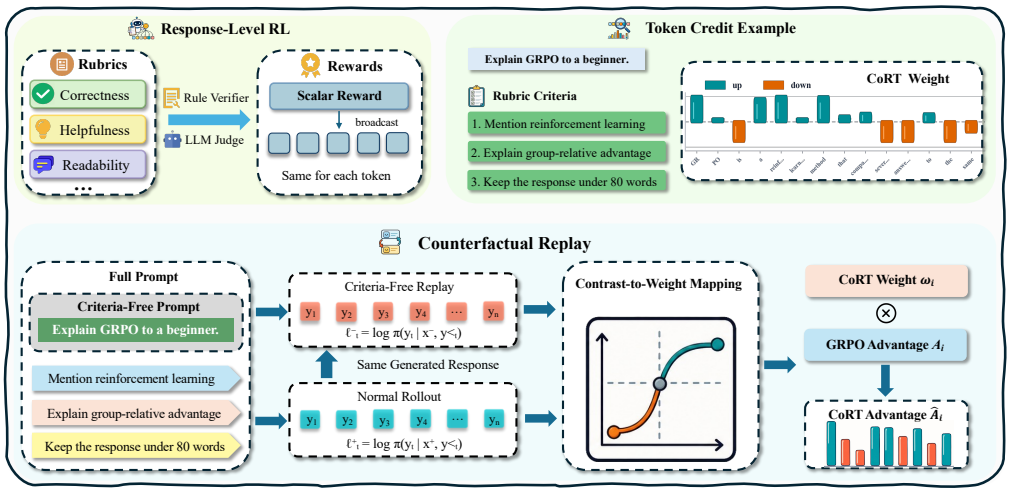

Figure 2: Overview of CoRT. Response level rubric rewards are first converted into a scalar GRPO advantage, which is normally broadcast uniformly to all generated tokens. CoRT replays the same response under a criteria-free counterfactual prompt, contrasts token log probabilities against the normal rollout, maps the contrast into token weights, and uses these weights to redistribute the response level advantage across tokens.

Policy optimization for language models. PPO is built around a clipped policy-gradient surrogate and remains widely used for language-model alignment (Schulman et al., 2017; Ouyang et al., 2022). GRPO removes the value model by normalizing rewards within a response group, and has become a common basis for scalable RL training of LLMs (Shao et al., 2024; Guo et al., 2025). Recent variants such as DAPO and GSPO refine large scale RL training through choices in normalization, clipping, and sampling (Yu et al., 2026; Zheng et al., 2025), while other work studies group normalization, multi reward settings, and credit assignment for reasoning (Parthasarathi et al., 2025; Liu et al., 2026). These GRPO-style methods primarily refine how response-level feedback is normalized, sampled, and optimized, but they do not specify how rubric criteria should be attributed to tokens within a response. CoRT redistributes the group-relative advantage across tokens while leaving the reward interface and normalization unchanged.

Fine grained credit assignment. Fine-grained supervision aims to alleviate the sparsity of outcome-only rewards by assigning training signals to smaller units of behavior. Process reward methods score intermediate reasoning steps (Lightman et al., 2024; Wang et al., 2024), while agentic RL studies sparse long-horizon credit assignment through implicit step rewards, hindsight credit assignment, and hierarchical advantage estimation (Liu et al., 2025; Tan et al., 2026; Peng et al., 2026). In instruction following, Rubrics-to-Tokens (RTT) learns a Token-Level Relevance Discriminator to convert response-level rubric outcomes into token-level rewards (Xu et al., 2026). As the closest prior work, RTT targets the same uniform-credit problem in rubric-based RL, but training its discriminator requires a nontrivial data-generation pipeline for token-level relevance supervision. CoRT keeps the goal of fine-grained credit assignment, but replaces discriminator relevance with a policy-internal likelihood contrast. This connects CoRT to context intervention and attribution methods that test whether outputs depend on specific parts of the input context (Cohen-Wang et al., 2024; Sarti et al., 2024; Shilpika et al., 2026), as well as broader counterfactual views of intervention (Pearl, 2009; Wachter et al., 2017). Rather than using such contrasts only to explain completed generations, CoRT uses them to redistribute the GRPO advantage during policy training.

3 Rubric Signals in Group-Relative RL

Let x be an instruction and c a set of rubric criteria. We denote the full rubric prompt by $x^+ = (x, c)$ and the criteria-free prompt by $x^- = x$. For each prompt group, the old policy samples G responses $y_i \sim \pi_{\theta_{\text{old}}}(\cdot \mid x^+)$. A rubric verifier assigns each response a scalar outcome reward r_i, and GRPO

Algorithm 1 CoRT pipeline for one prompt group

Require: Full prompt x^+, criteria-free prompt x^-, responses $\{y_i\}_{i=1}^G$, rewards $\{r_i\}_{i=1}^G$

1: Compute group-relative advantages A_i from rewards.
2: Obtain full-prompt log probabilities $\ell_{i,t}^+$ on sampled responses.
3: Score the same response token sequences under x^- to obtain $\ell_{i,t}^-$.
4: Compute counterfactual contrasts $\Delta_{i,t} = \ell_{i,t}^+ - \ell_{i,t}^-$.
5: Map contrasts to bounded replay-margin scores $s_{i,t}$.
6: Apply scheduled response normalization to obtain $w_{i,t}$.
7: Optimize the clipped policy objective with $\widehat{A}_{i,t} = \mathrm{sg}(w_{i,t})A_i$.

converts it into a group-relative advantage

$$A_i = \frac{r_i - \mu}{\sigma + \epsilon_r},$$

where μ and σ are the mean and standard deviation of rewards in the group (Shao et al., 2024).

Although the policy loss is evaluated tokenwise, standard GRPO uses the same group-relative advantage A_i for every generated token in response y_i. For token position t, define the policy ratio

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(y_{i,t} \mid x^+, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x^+, y_{i,<t})}.$$

Following the PPO-style clipped surrogate (Schulman et al., 2017), let

$$\bar{\rho}_{i,t}(\theta) = \mathrm{clip}(\rho_{i,t}(\theta), 1 - \epsilon_{\text{clip}}, 1 + \epsilon_{\text{clip}}).$$

The GRPO loss can be written as

$$\mathcal{L}_{\text{GRPO}} = -\mathbb{E}_{i,t}\left[\min(\rho_{i,t}A_i, \bar{\rho}_{i,t}A_i)\right],$$

where the expectation is over sampled responses and their generated tokens. This objective is agnostic to which tokens express the rubric criteria: the same response-level advantage coefficient is applied across the response.

Advantage redistribution. Our goal is not to call a new verifier or change the scalar outcome reward, but to redistribute the same response-level advantage within each sampled response. Let T_i be the length of response y_i. We require token weights $w_{i,t}$ to satisfy

$$\frac{1}{T_i}\sum_{t=1}^{T_i} w_{i,t} = 1,$$

so that the average shaped advantage remains

$$\frac{1}{T_i}\sum_{t=1}^{T_i} \widehat{A}_{i,t} = \frac{1}{T_i}\sum_{t=1}^{T_i} w_{i,t}A_i = A_i.$$

Under this constraint, token weights reallocate the signed advantage across tokens while preserving the response-level reward and group-relative normalization. CoRT realizes this by deriving $w_{i,t}$ from counterfactual replay and response normalization, replacing the uniform broadcast of A_i with rubric-dependent token weights.

4 CoRT

CoRT converts a sparse response-level GRPO advantage into dense token-level credit for the same sampled response. Figure 2 provides an overview of the pipeline, while Algorithm 1 gives the corresponding training procedure. For each prompt group, the standard GRPO path first converts scalar rubric rewards into response-level advantages. CoRT keeps the sampled responses and rewards fixed, then adds a counterfactual replay pass that scores the same response tokens under the criteria-free

prompt. Comparing these scores with the factual full-prompt scores yields tokenwise log-probability contrasts, which indicate which tokens depend more strongly on the rubric context. These contrasts are mapped to bounded token weights and normalized within each response before being multiplied with the original advantage. In this way, the replay signal determines how the update is distributed across tokens, while the signed GRPO advantage still determines whether those tokens are reinforced or suppressed. Thus, the method changes only within-response credit allocation while leaving the rollout, verifier reward, and response-level update direction unchanged.

4.1 Counterfactual Scoring

For each sampled response y_i, CoRT scores the same observed token trajectory under two contexts: the original prompt with rubric criteria, x_i^+, and the counterfactual prompt without these criteria, x_i^-. It computes

$$\ell_{i,t}^+ = \log \pi_{\bar{\theta}}(y_{i,t} \mid x_i^+, y_{i,<t}),$$
$$\ell_{i,t}^- = \log \pi_{\bar{\theta}}(y_{i,t} \mid x_i^-, y_{i,<t}).$$

Here $\bar{\theta}$ denotes the frozen scoring policy for the current batch. In implementation, $\ell_{i,t}^+$ is available from the normal rollout log probabilities, while $\ell_{i,t}^-$ is computed by replaying the same response tokens under the counterfactual prompt x_i^- and evaluating their per token log probabilities. The replay contrast $\Delta_{i,t} = \ell_{i,t}^+ - \ell_{i,t}^-$ compares the log probability assigned to the same next token under the criteria-conditioned and criteria-free contexts. Since the prefix and target token are fixed, positive contrasts indicate stronger dependence on the rubric criteria relative to the criteria-free baseline, while small or negative contrasts indicate weak or reversed dependence. We use this contrast as a proxy for rubric dependence, rather than as a calibrated measure of token importance.

4.2 Replay-Margin Token Credit

The replay contrasts $\Delta_{i,t}$ are unbounded and can be heavy-tailed or model-scale dependent. We therefore map them to a bounded score before constructing token credit weights:

$$s_{i,t} = \mathrm{sigmoid}(\tau(\Delta_{i,t} - b)) - \frac{1}{2},$$
$$\tilde{w}_{i,t} = 1 + \eta \lambda_k s_{i,t}.$$

Here b centers the replay contrast and τ controls the sharpness of the sigmoid transform. The resulting score $s_{i,t}$ lies in $(-\frac{1}{2}, \frac{1}{2})$ and becomes zero when $\Delta_{i,t} = b$. The schedule coefficient $\lambda_k \in [0, 1]$ controls how strongly replay weighting is activated at trainer step k, while η controls the overall weighting strength, giving provisional weights in $(1 - \frac{\eta\lambda_k}{2}, 1 + \frac{\eta\lambda_k}{2})$.

4.3 Scheduled Response Normalization

CoRT does not use the replay weights at full strength from the first update. Let k denote the trainer step and define the normalized progress u_k after warmup as

$$u_k = \mathrm{clip}\left(\frac{k - k_0}{K}, 0, 1\right),$$
$$\lambda_k = 3u_k^2 - 2u_k^3.$$

where k_0 is the warmup offset and K is the ramp length. This is the standard cubic Hermite SmoothStep schedule (Perlin, 1985), satisfying $\lambda_k = 0$ before the ramp, $\lambda_k = 1$ after the ramp, and zero slope at both endpoints. The schedule keeps early updates close to GRPO and smoothly introduces token-level weighting only after the rollout and reward statistics become more stable.

The resulting provisional weights are normalized within each response:

$$\bar{w}_i = \frac{1}{T_i} \sum_{t=1}^{T_i} \tilde{w}_{i,t}, \qquad w_{i,t} = \frac{\tilde{w}_{i,t}}{\bar{w}_i}.$$

By construction, $\frac{1}{T_i} \sum_{t=1}^{T_i} w_{i,t} = 1$ and $\frac{1}{T_i} \sum_{t=1}^{T_i} w_{i,t} A_i = A_i$. This preserves the GRPO advantage scale while redistributing signed credit across response tokens.

4.4 POLICY OBJECTIVE

The token-shaped advantage is

$$\widehat{A}_{i,t} = \mathrm{sg}(w_{i,t})A_i,$$

where $\mathrm{sg}(\cdot)$ stops gradients through the replay-derived weight. CoRT uses the same clipped surrogate form as GRPO, but replaces the uniform response-level advantage A_i with the token-dependent advantage $\widehat{A}_{i,t}$:

$$g_{i,t}(a) = \min(\rho_{i,t}a, \bar{\rho}_{i,t}a)\,.$$

The resulting policy loss is

$$\mathcal{L}_{CoRT} = -\mathbb{E}_{i,t}\Big[g_{i,t}(\widehat{A}_{i,t})\Big]\,,$$

where the expectation is over sampled responses and their generated tokens. Thus, $w_{i,t}$ controls the magnitude of the signed update at each token, while the original group-relative advantage A_i determines whether the token is reinforced or suppressed during the policy update.

Computation. CoRT adds one additional forward scoring pass for each sampled response under the criteria-free prompt x_i^-. This pass reuses the generated response tokens and computes their per-token log probabilities, without extra generation, verifier calls, rejection sampling, or token-level relevance labels. Since the factual log probabilities under x_i^+ are already available from rollout, the additional computation is limited to counterfactual scoring and response normalization for advantage construction. Thus, CoRT remains a lightweight modification to GRPO compared with methods that require a separate relevance-learning stage.

5 EXPERIMENTAL SETUP

Training data. Training uses HIR-16k (Zhang et al., 2025). Each example provides a prompt without the instruction list and the corresponding instruction list, allowing us to construct the two contexts required by CoRT. We obtain x^+ by appending the instruction list to the end of the prompt and use it throughout the main RL loop. We use the original prompt as x^- only for replay scoring.

Reward modes. Following RTT (Xu et al., 2026), we evaluate two rubric reward granularities: Constraint Satisfaction Rate (CSR) and All-or-Nothing (AON). Let M be the number of positive criteria for a response, and let $z_j \in \{0, 1\}$ indicate whether the j-th criterion is satisfied. CSR assigns the fraction of satisfied criteria, while AON assigns reward one only when all positive criteria are satisfied:

$$r_{\mathrm{CSR}} = \frac{1}{M}\sum_{j=1}^{M} z_j, \qquad r_{\mathrm{AON}} = \mathbf{1}\left[\sum_{j=1}^{M} z_j = M\right].$$

These rewards are used for RL training only; validation uses the official metrics of each benchmark.

Models and optimization. The main matrix contains Qwen3-4B-Instruct and Qwen2.5-7B-Instruct, each trained under CSR and AON (Yang et al., 2025a;b). All runs use a prompt batch size of 64, sample 8 responses per prompt, set maximum prompt and response lengths to 2048 and 4096, and use rollout temperature 1.0, top-p 0.99, and top-k 100. The actor learning rate is 10^{-6} with warmup ratio 0.03 and weight decay 0.1. We do not use an explicit reward-KL term or an auxiliary actor-KL loss. The lower and upper clipping bounds are 0.2 and 0.27, respectively.

CoRT instantiation. Unless otherwise stated, CoRT uses a zero-centered replay-margin transform with unit temperature, $\eta = 0.5$, a 100-step SmoothStep ramp, and response-mean normalization. Before response normalization, this bounds the active token multiplier in $(0.75, 1.25)$. The replay-derived weights are applied to positive and negative group-relative advantages, so the method changes token credit rather than the reward definition.

Evaluation. Validation uses IFBench, IFEval, MultiDimIF, and AdvancedIF (Pyatkin et al., 2026; Zhou et al., 2023b; Ye et al., 2025; He et al., 2025). IFBench and IFEval report strict prompt-level and instruction-level accuracy. MultiDimIF reports accuracy over multi-dimensional instruction constraints. AdvancedIF follows its judge-based protocol with `o3-mini`. For all benchmarks, all methods are evaluated with 5 sampled responses per prompt, temperature 0.7, top-p 0.8, and top-k 20. Following the validation protocol of RTT (Xu et al., 2026), we report the best validation score within the first 500 trainer steps for IFBench, IFEval, and MultiDimIF under this shared protocol. We also report the corresponding step-500 CoRT results in Table 4. AdvancedIF uses the same 5-sample decoding protocol and is evaluated at step 500.

6 Results

Table 1 reports the main validation results, organized by model family and training reward. Within each CSR or AON group, GRPO, RTT, and CoRT are compared under the same reward setting. Shaded rows denote CoRT, and bold and underlined values mark the best and second-best results within each model block. SFT, DPO, and RTT denote supervised fine-tuning, direct preference optimization, and Rubrics-to-Tokens, respectively. CSR and AON denote Constraint Satisfaction Rate and All-or-Nothing rewards.

Overall, CoRT performs strongly and consistently across the main evaluation matrix. Under the same reward family, it improves over response-level GRPO in most comparisons across models, reward settings, and benchmarks. Compared with RTT, CoRT is competitive and often stronger, while avoiding a learned token-level relevance discriminator.

Main comparison. The most controlled comparison is within each reward family, where GRPO and CoRT use the same rollout prompts, scalar rewards, and group-relative normalization. Across CSR and AON settings, CoRT improves over response-level GRPO in most comparisons across models and benchmarks. The gains also hold on AdvancedIF under the fixed-step judge-based protocol, where CoRT improves over matched GRPO across the reported metrics. These results show that the same response-level rubric signal can become more effective when its advantage is redistributed across tokens, without changing the verifier reward or group-relative normalization.

Comparison with learned token relevance. RTT directly targets the response-to-token credit gap by training a token-level relevance discriminator (Xu et al., 2026). CoRT explores a different tradeoff by deriving token weights from policy-internal counterfactual likelihood contrasts rather than learning a separate relevance model. Across the main comparisons, CoRT is competitive with and often stronger than RTT under the same evaluation protocol, while avoiding a separate relevance-learning stage. This suggests that explicit token relevance training is not always necessary for token-level credit assignment in rubric-conditioned GRPO.

Qwen3-14B results. Table 2 evaluates whether CoRT extends to a larger model scale. Under CSR, CoRT improves all reported metrics; under AON, it improves IFBench and MultiDimIF while trailing GRPO on IFEval. These results show that replay-based credit weighting remains effective at 14B scale, with metric-specific variation under sparse rewards.

Compatibility with policy optimization objectives. Because CoRT redistributes token credit after the response-level advantage is computed, it can be combined with different policy optimization objectives. We integrate CoRT into DAPO (Yu et al., 2026) and GSPO (Zheng et al., 2025) under the CSR reward on Qwen3-4B-Instruct-2507. As shown in Table 3, adding CoRT improves all reported metrics over DAPO and improves four of the five metrics over GSPO, with only a small decrease on MultiDimIF. This suggests that replay weighting can complement both objectives.

7 Training Diagnostics

On Qwen2.5-7B under CSR, we compare the full recipe with variants removing response normalization, the SmoothStep ramp, or both. The full recipe keeps the mean token weight close to one, length clipping low, and gradient norms and entropy stable. It also maintains the strongest average

Method	IFEval		IFBench		MultiDimIF	AdvancedIF	
	Prompt	Instruction	Prompt	Instruction	Accuracy	Overall	Rubric
Qwen3-4B-Instruct							
Instruct	83.40	88.13	30.95	32.83	56.08	44.31	79.17
SFT	83.73	88.85	29.59	31.64	56.50	45.07	79.23
DPO	82.62	87.77	29.93	32.84	55.83	45.71	79.33
CSR reward							
GRPO	84.29	89.21	32.31	33.43	74.38	46.02	79.81
+RTT	85.03	<u>90.17</u>	34.01	36.12	76.33	<u>48.39</u>	80.77
+CoRT	**86.06**	**90.48**	34.49	36.24	<u>80.48</u>	**49.18**	**81.14**
	+1.77	+1.27	+2.18	+2.81	+6.10	+3.16	+1.33
AON reward							
GRPO	82.99	88.61	32.99	35.52	74.25	47.29	79.69
+RTT	<u>85.21</u>	88.73	<u>34.69</u>	**37.61**	76.75	47.48	<u>80.79</u>
+CoRT	84.66	89.50	**35.10**	<u>37.37</u>	**81.60**	48.03	80.10
	+1.67	+0.89	+2.11	+1.85	+7.35	+0.74	+0.41
Qwen2.5-7B-Instruct							
Instruct	69.68	77.93	25.85	27.16	51.08	29.48	71.31
SFT	71.90	79.38	28.23	30.15	52.83	29.71	69.61
DPO	71.16	78.90	27.55	28.96	51.08	30.82	70.90
CSR reward							
GRPO	78.56	84.41	28.23	29.85	66.67	30.91	72.00
+RTT	81.15	86.33	<u>32.31</u>	<u>34.63</u>	69.67	33.74	**73.72**
+CoRT	81.92	**87.51**	**34.63**	**36.30**	<u>78.52</u>	**34.18**	73.28
	+3.36	+3.10	+6.40	+6.45	+11.85	+3.27	+1.28
AON reward							
GRPO	79.67	85.13	31.97	33.13	69.41	32.10	72.33
+RTT	**82.26**	87.29	<u>32.31</u>	33.43	71.83	32.51	72.70
+CoRT	<u>82.00</u>	<u>87.36</u>	29.66	31.22	**79.63**	<u>33.75</u>	<u>73.57</u>
	+2.33	+2.23	-2.31	-1.91	+10.22	+1.65	+1.24

Table 1: Main validation results on instruction following benchmarks. Rows are grouped by model family and reward setting. Baseline rows, including Instruct, SFT, DPO, GRPO, and RTT, are taken from RTT (Xu et al., 2026); CoRT rows are our runs under the same evaluation protocol. Deltas are vs. matched GRPO. Shaded rows indicate CoRT; bold and underlined values mark the best and second-best results within each model block.

Method	IFEval		IFBench		MultiDim.
	Prompt	Inst.	Prompt	Inst.	Acc.
CSR reward					
GRPO	87.87	92.34	39.93	42.53	82.59
+CoRT	**89.17**	**93.28**	**40.68**	**43.11**	**83.48**
AON reward					
GRPO	**90.13**	**93.94**	35.84	37.92	83.43
+CoRT	89.80	93.48	**38.50**	**41.09**	**84.08**

Table 2: Qwen3-14B results.

validation and training reward around step 500, although some ablations score higher on IFBench. These results support complementary roles for response normalization and scheduled activation in this setting. Full diagnostics and continuation results appear in Appendix A.

Method	IFEval		IFBench		MultiDim.
	Prompt	Inst.	Prompt	Inst.	Acc.
Qwen3-4B-Instruct					
DAPO	87.10	88.90	40.27	45.79	83.41
+CoRT	**87.95**	**90.20**	**41.30**	**47.02**	**84.06**
GSPO	86.14	91.10	35.97	38.74	**82.26**
+CoRT	**86.54**	**91.20**	**36.11**	**39.42**	81.96

Table 3: CoRT with DAPO and GSPO (CSR).

Preliminary computation cost. In a five-step Qwen3-4B CSR pilot with 64 prompts and eight responses per prompt, mean step times over steps 2–5 were 154.93 s for CoRT and 149.84 s for GRPO (+3.40%). The corresponding mean response lengths were 332.6 and 333.5 tokens, well below the pilot's 1,024-token response cap. This is a preliminary timing comparison; its training configuration differs from the formal experiments.

8 Conclusion

We presented CoRT, a lightweight credit assignment method for rubric-based instruction following RL. The central observation is that response-level criteria rewards are optimized through tokenwise policy updates, yet GRPO assigns the same response advantage to all generated tokens. CoRT addresses this within-response mismatch by using counterfactual replay to rescore sampled responses after removing the criteria, and by converting the resulting likelihood contrast into normalized token credit weights. These weights redistribute the signed GRPO advantage toward tokens that depend more strongly on the criteria context, while keeping the original rollout distribution, verifier, scalar reward, and clipped GRPO surrogate structure intact without training a separate token relevance model. The bounded replay-margin transform, response normalization, and scheduled activation further make this redistribution stable in practice.

Across instruction following benchmarks and reward granularities, CoRT improves over response-level GRPO in most comparisons and remains competitive with learned token relevance methods. Its consistent gains when combined with DAPO and GSPO further show that replay-derived token credit is not specific to standard GRPO and can be integrated with different policy optimization algorithms. The training diagnostics further show that response normalization and scheduled activation help keep token weights on the GRPO advantage scale and reduce unstable training behavior. These results support the view that structured criteria can be useful not only for scoring responses, but also for guiding where credit is assigned within a response. Since CoRT estimates criteria reliance along observed trajectories, its signal can be weaker when criteria have little effect on token likelihoods, when they are redundant with the original prompt, or when the response-level reward is noisy. Future work can explore finer-grained interventions, criterion-specific allocation, and longer-horizon credit across turns, actions, and tool calls.

AI Use Statement

Generative AI tools were used in an assistive capacity, primarily to support literature research, manuscript polishing, and the writing of portions of the code. The authors take responsibility for the final manuscript, code, and reported results.

REFERENCES

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in Neural Information Processing Systems*, 30, 2017.

Benjamin Cohen-Wang, Harshay Shah, Kristian Georgiev, and Aleksander Mądry. Contextcite: Attributing model generation to context. *Advances in Neural Information Processing Systems*, 37:95764–95807, 2024.

Zihui Gu, Xingwu Sun, Fengzong Lian, Zhanhui Kang, Chengzhong Xu, and Ju Fan. DINGO: Towards diverse and fine-grained instruction-following evaluation. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence*, volume 38, pp. 18108–18116, 2024.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean M. Hendryx. Rubrics as rewards: Reinforcement learning beyond verifiable domains. In *NeurIPS 2025 Workshop on Efficient Reasoning*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Qianyu He, Jie Zeng, Qianxi He, Jiaqing Liang, and Yanghua Xiao. From complex to simple: Enhancing multi-constraint complex instruction following ability of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10864–10882, 2024.

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. *arXiv preprint arXiv:2511.10507*, 2025.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Sunzhu Li, Jiale Zhao, Miteto Wei, Huimin Ren, Yang Zhou, Jingwen Yang, Shunyu Liu, Kaike Zhang, and Wei Chen. Rubrichub: A comprehensive and highly discriminative rubric dataset via automated coarse-to-fine generation. *arXiv preprint arXiv:2601.08430*, 2026.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, et al. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *arXiv preprint arXiv:2601.05242*, 2026.

Xiaoqian Liu, Ke Wang, Yuchuan Wu, Fei Huang, Yongbin Li, Junge Zhang, and Jianbin Jiao. Agentic reinforcement learning with implicit step rewards. *arXiv preprint arXiv:2509.19199*, 2025.

Renze Lou, Kai Zhang, Jian Xie, Yuxuan Sun, Jihyun Ahn, Hanzi Xu, Yu Su, and Wenpeng Yin. Muffin: Curating multi-faceted instructions for improving instruction following. In *International Conference on Learning Representations*, volume 2024, pp. 21464–21493, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Prasanna Parthasarathi, Mathieu Reymond, Boxing Chen, Yufei Cui, and Sarath Chandar. Grpo-λ: Credit assignment improves llm reasoning. *arXiv preprint arXiv:2510.00194*, 2025.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2 edition, 2009.

Hao Peng, Yunjia Qi, Xiaozhi Wang, Bin Xu, Lei Hou, and Juanzi Li. Verif: Verification engineering for reinforcement learning in instruction following. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 30312–30327, 2025.

Jiangweizhi Peng, Yuanxin Liu, Ruida Zhou, Charles Fleming, Zhaoran Wang, Alfredo Garcia, and Mingyi Hong. Hiper: Hierarchical reinforcement learning with explicit credit assignment for large language model agents. *arXiv preprint arXiv:2602.16165*, 2026.

Ken Perlin. An image synthesizer. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 287–296, 1985.

Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. *Advances in Neural Information Processing Systems*, 33:3976–3990, 2020.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hanna Hajishirzi. Generalizing verifiable instruction following. *Advances in Neural Information Processing Systems*, 38, 2026.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Gabriele Sarti, Grzegorz Chrupała, Malvina Nissim, and Arianna Bisazza. Quantifying the plausibility of context reliance in neural machine translation. In *International Conference on Learning Representations*, volume 2024, pp. 25526–25554, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

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

Shilpika Shilpika, Carlo Graziani, Bethany Lusch, Venkatram Vishwanath, and Michael E Papka. Probabilistic attribution for large language models. *arXiv preprint arXiv:2605.21726*, 2026.

Haoran Sun, Lixin Liu, Junjie Li, Fengyu Wang, Baohua Dong, Ran Lin, and Ruohui Huang. Conifer: Improving complex constrained instruction-following ability of large language models. *arXiv preprint arXiv:2404.02823*, 2024.

Hui-Ze Tan, Xiao-Wen Yang, Hao Chen, Jie-Jing Shao, Yi Wen, Yuteng Shen, Weihong Luo, Xiku Du, Lan-Zhe Guo, and Yu-Feng Li. Hindsight credit assignment for long-horizon llm agents. *arXiv preprint arXiv:2603.08754*, 2026.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *arXiv preprint arXiv:1711.00399*, 2017.

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

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9426–9439, 2024.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. Wizardlm: Empowering large pre-trained language models to follow complex instructions. In *International Conference on Learning Representations*, volume 2024, pp. 30745–30766, 2024.

Tianze Xu, Yanzhao Zheng, Pengrui Lu, Lyumanshan Ye, Yong Wu, Zhentao Zhang, Yuanqiang Yu, Chao Ma, Jihuai Zhu, Pengfei Liu, Baohua Dong, Hangcheng Zhu, Ruohui Huang, and Gang Yu. Rubrics to tokens: Bridging response-level rubrics and token-level rewards in instruction following tasks. *arXiv preprint arXiv:2604.02795*, 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*, 2025a.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, et al. Qwen2.5 technical report, 2025b.

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

Junjie Ye, Caishuang Huang, Zhuohan Chen, Wenjie Fu, Chenyuan Yang, Leyi Yang, Yilong Wu, Peng Wang, Meng Zhou, Xiaolong Yang, Gui Tao, Qi Zhang, Jianping Shi, and Xuanjing Huang. A multi-dimensional constraint framework for evaluating and improving instruction following in large language models. *arXiv preprint arXiv:2505.07591*, 2025.

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

Kongcheng Zhang, Qi Yao, Shunyu Liu, Wenjian Zhang, Min Cen, Yang Zhou, Wenkai Fang, Yiru Zhao, Baisheng Lai, and Mingli Song. Replay failures as successes: Sample-efficient reinforcement learning for instruction following. *arXiv preprint arXiv:2512.23457*, 2025.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021, 2023a.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023b.

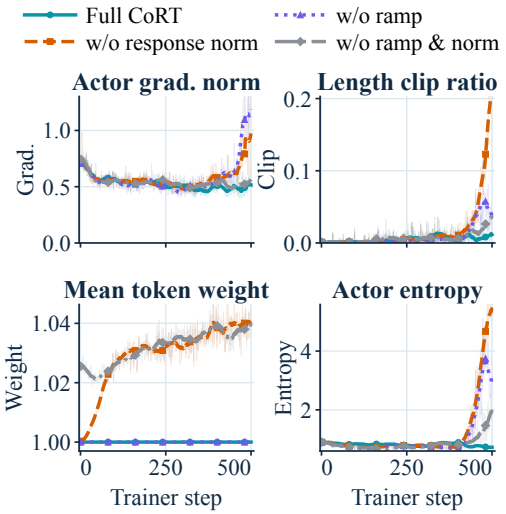

Figure 3: Qwen2.5-7B CSR diagnostics for CoRT integration variants over the first 500 trainer steps. Curves use 21-step smoothing.

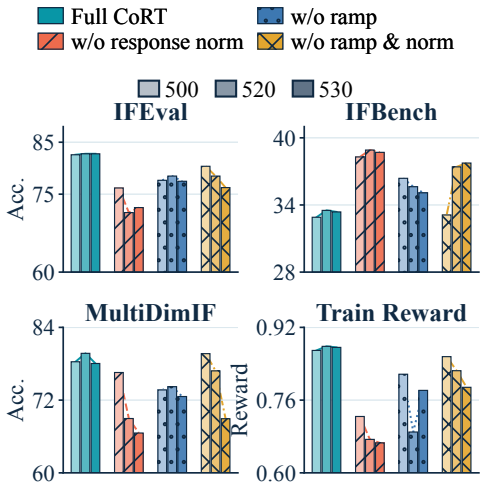

Figure 4: Post-500 continuation for Qwen2.5-7B CSR integration ablations. Validation reports internal `mean@5` accuracy at steps 500, 520, and 530; training reward is a 10-step trailing mean ending at each checkpoint.

A Integration Diagnostics

Figure 3 examines whether the integration components in CoRT stabilize training dynamics as intended. We use Qwen2.5-7B under the CSR reward and compare the full recipe with variants that remove response normalization, remove the SmoothStep ramp, or both. This ablation separates scale control from scheduled activation and tests whether these components keep replay-derived token weighting close to the GRPO update scale throughout the full training run.

With the full recipe, the mean token weight stays close to one, length clipping remains near zero, and the actor gradient norm and entropy stay stable. Removing response normalization causes the mean token weight to drift upward, approaching 1.04 near the end of training. This scale drift coincides with a sharp late increase in length clipping, gradient norm, and entropy, suggesting that unnormalized replay weights can act as a response-level multiplier. Removing the ramp keeps the mean token weight controlled by response normalization, but the abrupt use of replay-derived weights leads to a late gradient and entropy spike with higher length clipping than the full recipe. The variant without both controls also shows elevated mean weights and higher length clipping than the full recipe. These patterns indicate that the two controls address different stability issues: response normalization controls coefficient scale, while scheduled activation introduces token weights gradually.

Figure 4 shows the post-500 validation results. At steps 500, 520, and 530, the full recipe has the highest average across IFEval, IFBench, and MultiDimIF and the highest 10-step trailing training reward among the tested variants. Some ablations achieve higher IFBench accuracy. The values are reported in Table 10.

B Additional Results

B.1 Fixed Step-500 CoRT Checkpoints

The main table selects the best official validation checkpoint within the first 500 trainer steps for IFBench, IFEval, and MultiDimIF. Table 4 reports the corresponding CoRT results at exactly step 500.

Method	IFEval		IFBench		MultiDimIF
	Prompt	Instruction	Prompt	Instruction	Accuracy
Qwen3-4B					
CoRT-CSR	86.06	90.48	34.49	36.24	80.48
CoRT-AON	84.66	89.50	32.31	34.44	81.60
Qwen2.5-7B					
CoRT-CSR	81.77	87.36	34.63	36.30	78.52
CoRT-AON	81.81	87.29	29.32	30.27	78.62

Table 4: CoRT results at exactly step 500 for the main experiment matrix.

B.2 General Capability Checks

Table 5 reports mean@5 evaluation on Math500, GPQA-Diamond, and MMLU-Pro for the original instruction-tuned models and the step-500 CoRT checkpoints. These evaluations use the same original instruction-tuned checkpoint for the CSR and AON comparisons; only the CoRT checkpoint changes with the reward granularity. This provides a complementary check on whether instruction-following optimization changes the model's general math and knowledge capabilities. Across both model families, CoRT largely preserves this general capability profile.

Method	Math500	GPQA	MMLU-Pro	Avg.
Qwen3-4B				
Instruct	86.28	57.98	70.16	71.47
	86.52	59.09	69.62	71.74
CoRT-CSR	+0.24	+1.11	-0.54	+0.27
	86.36	57.78	70.18	71.44
CoRT-AON	+0.08	-0.20	+0.02	-0.03
Qwen2.5-7B				
Instruct	70.64	33.23	55.48	53.12
	69.32	33.84	54.34	52.50
CoRT-CSR	-1.32	+0.61	-1.14	-0.62
	70.20	33.64	55.27	53.04
CoRT-AON	-0.44	+0.41	-0.21	-0.08

Table 5: General capability checks under mean@5 evaluation. Deltas are relative to the original instruction-tuned model within each family.

B.3 Relation to Self-OPD

On-policy self-distillation and CoRT share the goal of making sparse response-level supervision produce denser token-level training signals on trajectories sampled from the current policy. Self-Distilled RLVR derives token-level policy differences from a privileged context while keeping the RLVR reward as the update direction (Yang et al., 2026). TRACE further routes self-OPD signals to critical spans, reducing updates on redundant tokens and limiting privileged-information leakage (Wang et al., 2026). CoRT differs in the source and role of the dense signal. It does not use a privileged answer trace, span annotation, or external target distribution. Instead, it keeps the sampled response fixed, removes the rubric criteria from the prompt, and treats the criteria-free log probability as a background likelihood baseline. The resulting full-vs.-criteria-free contrast is then multiplied by the signed GRPO advantage: successful responses reinforce tokens whose likelihood depends on the rubric context, while unsuccessful responses suppress such tokens more strongly. Thus, CoRT is closer to signed counterfactual contrast shaping than to self-distillation, even though both approaches use on-policy token-level modulation.

B.4 Replay Case Studies

We include several HIR replay examples to clarify what the replay contrast measures on actual generated responses. These examples serve three purposes. First, standard replay cases show whether high-contrast tokens correspond to visible criteria-controlled spans. Second, matched context con-

trols test whether the signal is explained only by removing prompt text or by changing the length of the prompt. Third, criterion-wise removal checks whether removing a specific criterion produces localized changes on the tokens that realize that criterion. Together, these examples illustrate how the replay signal responds to changes in rubric context.

The first case uses a HIR prompt asking why a balanced diet matters for athletes. The criteria include several visible formatting requirements, including English output, at least five sentences, exactly three Markdown bullet points, a postscript beginning with `P.S.`, a fixed final phrase, and enclosing double quotation marks. We sample a response with the full criteria from the step-500 Qwen3-4B CoRT checkpoint, then replay it without them.

The text below shows the prompt, criteria, and response excerpt used in the replay analysis. We abbreviate non-critical explanatory content with ellipses, while preserving the visible criteria-controlled spans that appear in Table 6.

> **Prompt.** Provide a detailed response to the following query: "Explain the importance of maintaining a balanced diet for athletes."
>
> **Rubric criteria.** The response should be in English, contain at least five sentences, contain exactly three Markdown bullet points, add a postscript beginning with `P.S.`, finish with the exact phrase "Is there anything else I can help with?", and wrap the entire response with double quotation marks.
>
> **Generated response excerpt.** "Maintaining a balanced diet is crucial for athletes because it directly impacts their performance, recovery, and overall health. A well-balanced diet provides the necessary nutrients that fuel physical activity, support muscle repair, and enhance immune function. . . .
>
> Here are three key reasons why a balanced diet is important for athletes:
>
> * This is point 1. It ensures consistent energy supply during prolonged physical activity.
> * This is point 2. It supports muscle growth and repair through adequate protein intake.
> * This is point 3. It strengthens the immune system and promotes better overall health.
>
> . . .
>
> `P.S.` Remember that individual nutritional needs may vary based on sport, age, gender, and activity level, so athletes should consult with a sports nutritionist to tailor their diet effectively. Is there anything else I can help with?"

Table 6 reports representative high-contrast tokens from the same fixed response. The selected tokens align with the visible requirements in the criteria: the opening quotation mark realizes the wrapping constraint, `three` reflects the required number of bullet points, `*` marks the Markdown bullet structure, `P` and `.S` realize the required postscript prefix, and `Is` and `?"` correspond to the fixed final phrase and closing quote. In contrast, these tokens are not simply generic content words about nutrition or athletes. This case supports replay contrast as a proxy for criteria dependence.

Token	Rubric cue	Δ_t	w_t
`"`	opening quotation mark	26.37	1.19
`three`	exact number of bullet points	16.96	1.19
`*`	Markdown bullet marker	20.75	1.19
`P`	start of postscript	12.62	1.19
`.S`	postscript continuation	7.47	1.19
`Is`	start of fixed final phrase	13.61	1.19
`?"`	final phrase and closing quote	3.17	1.17

Table 6: Replay contrasts for the balanced-diet HIR case (Qwen3-4B CoRT, step 500).

We next use a pandemic-economy prompt as a control-friendly case. Unlike the balanced-diet example, this prompt contains several separable local requirements, including section headers, Markdown bullets, highlighted spans, and the keyword `pandemic`. This makes it suitable for the matched-context and criterion-wise removal controls below. Table 7 shows the basic replay pattern: high-contrast tokens appear on the required output elements.

Token	Rubric cue	Δ_t	w_t
`Section`	start of required section label	34.88	1.17
`*`	highlighted-span marker	23.59	1.17
`highlight`	highlighted-span content	14.11	1.17
`-`	Markdown bullet marker	11.31	1.17
`point`	bullet-list template word	11.70	1.17
`pandemic`	required repeated keyword	4.35	1.17

Table 7: Replay contrasts for the pandemic HIR case (Qwen3-4B CoRT, step 500).

We next run matched context controls on the same pandemic response to test whether the signal is mainly an artifact of prompt removal. Table 8 compares the criteria-free replay with two alternatives: replacing the criteria with neutral filler of similar length, and replacing them with criteria from another HIR example. In all three settings, visible criteria-controlled tokens have larger average contrast than other tokens. The cue-token contrast also persists with neutral filler of similar length, so prompt shortening alone does not explain it.

Control context	Cue $\bar{\Delta}$	Other $\bar{\Delta}$	Representative high-contrast tokens
Criteria-free prompt	6.11	1.02	`Section`, `*`, `highlight`, `-`
Length-matched neutral filler	6.67	1.20	`Section`, `*`, `highlight`, `-`
Shuffled HIR criteria	5.76	0.97	`Section`, `-`, repeated section markers

Table 8: Matched context controls for the pandemic case. Cue tokens are visible criteria-controlled tokens such as section labels, Markdown markers, highlight markers, and the required keyword.

Criterion-wise removal provides a more local control. We rescore the same fixed response after removing one criterion at a time from the full criteria list. Table 9 shows that local criteria produce localized effects. Removing the bullet criterion emphasizes bullet markers and bullet template words; removing the highlight criterion emphasizes emphasis markers and `highlight`; and removing the section criterion emphasizes `Section`. In contrast, removing global criteria such as English language or minimum sentence count has little concentrated effect on visible cue tokens. These controls suggest that local criteria yield localized likelihood changes, whereas global or absence-style criteria have diffuse effects.

Removed criterion	Cue $\bar{\Delta}$	Other $\bar{\Delta}$	Highest affected tokens
At least five sentences	0.07	0.11	newline and generic opening tokens
English language	-0.01	-0.00	no concentrated rubric cue
Keyword `pandemic`	0.67	0.21	`pandemic` (6.36, 4.08), nearby temporal words
Exactly two bullets	2.29	0.37	`-` (35.87), `point` (18.74), `This` (12.13)
Highlighted spans	4.67	0.37	`*` (27.76, 23.33), `highlight` (14.24)
Two section labels	3.85	0.27	`Section` (43.25, 18.09)

Table 9: Criterion-wise removal controls on the pandemic case. Values are computed as full-prompt log probability minus the log probability after removing one criterion.

B.5 Post-500 Training and Validation Continuation

Table 10 reports the values behind Figure 4. Validation uses the internal `mean@5` fields from the Qwen2.5-7B CSR integration-ablation runs. IFEval, IFBench, and MultiDimIF are evaluated at selected trainer steps, and the average is the unweighted mean of these three accuracies. Training reward is a 10-step trailing mean ending at each selected step.

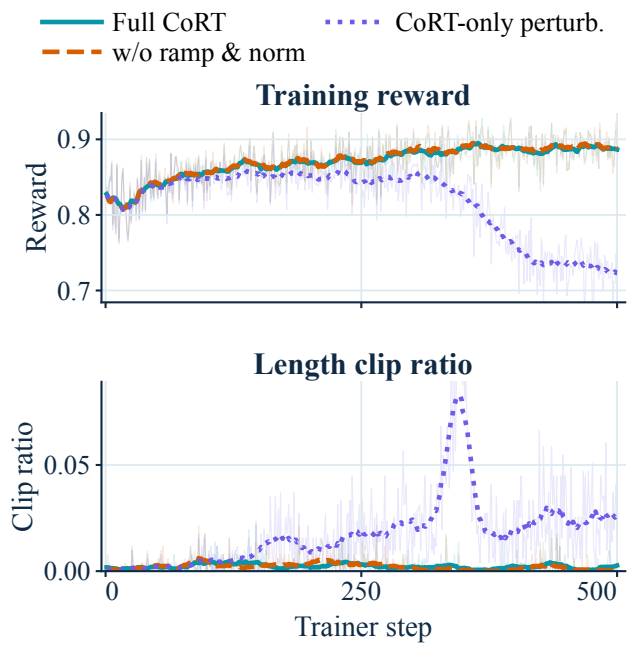

Figure 5: Qwen3-4B CSR failure-mode diagnostics for a CoRT-only perturbation variant over the first 500 trainer steps. Curves are smoothed over a 21-step window.

Variant	Step	IFEval	IFBench	MultiDimIF	Avg.	Train R.
Full	500	82.59	32.90	78.33	64.61	0.869
	520	82.77	33.52	79.72	65.34	0.878
	530	82.77	33.38	78.04	64.73	0.876
w/o norm	500	76.19	38.29	76.56	63.68	0.724
	520	71.46	38.91	68.96	59.78	0.674
	530	72.42	38.70	66.57	59.23	0.666
w/o ramp	500	77.67	36.38	73.68	62.58	0.817
	520	78.48	35.63	74.21	62.77	0.690
	530	77.49	35.09	72.59	61.72	0.781
w/o both	500	80.37	33.11	79.67	64.38	0.856
	520	78.48	37.41	76.82	64.24	0.825
	530	76.27	37.75	68.93	60.98	0.788

Table 10: Qwen2.5-7B CSR post-500 continuation for CoRT integration ablations. Validation is internal `mean@5`; training reward is a 10-step trailing mean.

B.6 Failure-Mode Diagnostics

Figures 3 and 5 summarize the diagnostic runs used to check CoRT's integration controls. In Qwen2.5-7B CSR runs, removing response normalization leads to elevated length clipping and a larger token-weight scale, while removing the activation ramp increases late-training gradient and entropy relative to the full recipe. In Qwen3-4B CSR runs, a CoRT-only perturbation variant that applies replay-derived perturbations without the standard GRPO advantage path shows lower training reward and larger length clipping; this run was stopped shortly after step 500.

These patterns clarify how the integration controls interact. Response normalization keeps the average token multiplier on the GRPO scale, so replay-derived weights redistribute credit within a response instead of changing the response-level update magnitude. The SmoothStep ramp delays the full strength of token weighting until the early rollout and reward statistics have stabilized, reducing abrupt changes in the policy-gradient multiplier. The GRPO advantage path supplies the signed reward direction: high-advantage responses reinforce rubric-sensitive tokens, while low-advantage

responses suppress them. When replay perturbations are used without this reward-conditioned direction, the token contrast can still change the policy, but the update is no longer anchored to group-relative outcome improvement. The failure-mode curves are consistent with this design choice, showing that CoRT works best as a credit allocator for an existing advantage rather than as an independent token-level training signal.