HEAT-GRPO: HEAT-KERNEL GROUP RELATIVE POLICY OPTIMIZATION FOR MULTI-RUBRIC REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Training language models to generate useful and reliable responses requires feedback that captures multiple dimensions of quality beyond correctness. Multi-rubric reinforcement learning offers such feedback, yet reducing nuanced judgments to discrete rewards can discard the distinctions needed for learning. We empirically analyze how nonlinear scoring of rubric-level judge margins before aggregation changes group-relative learning. Heat-GRPO realizes this design choice by smoothing hard verdicts under Gaussian margin perturbations, preserving margin distinctions while limiting the influence of extreme judgments. Identical-response comparisons isolate ranking changes; controlled training comparisons under shared scalar normalization demonstrate gains over Hard and Linear scoring. Comparing complete learning configurations across six models—Qwen3-1.7B, Qwen3-4B, Qwen3-8B, Llama-3.2-1B, Llama-3.2-3B, and Llama-3.1-8B—and four benchmarks, Heat-GRPO improves rubric scores by up to 25.0% relative to the strongest baseline and achieves the highest three-seed mean in 23 of 24 conditions. Vectorized scoring and scalar aggregation reduce advantage computation time by 38.9% and 38.2% relative to GDPO and GD^{2}PO, respectively, while retaining a cost comparable to GRPO. Additional evaluation with Qwen3.8-27B, gpt-oss-20b, and Gemma-4-31B ranks Heat-GRPO first in 22, 20, and 21 of the 24 conditions, respectively, supporting the robustness of its overall advantage across evaluators.

1 INTRODUCTION

Generating a good answer requires more than getting the answer right. A scientific explanation needs sound reasoning as well as a correct conclusion; medical advice must be useful while appropriately communicating uncertainty and risk. Training large language models (LLMs) for such tasks requires feedback that reflects these demands together. Learning rewards from human preferences offers one route to expressing such goals (Christiano et al., 2017; Ouyang et al., 2022), but a single correctness judgment provides limited guidance about what a response does well and where it should improve.

Multi-rubric reinforcement learning addresses this challenge by decomposing response quality into natural-language evaluation criteria (Gunjal et al., 2026). An LLM judge evaluates responses against each criterion, allowing qualitative requirements to inform repeated policy updates (Zheng et al., 2023). This broadens the feedback available for training. Rich evaluation criteria, however, do not automatically yield rich learning signals: the distinctions identified during evaluation must survive the conversion into policy updates.

This conversion is particularly consequential when learning relies on comparisons within response groups. Group Relative Policy Optimization (GRPO) (Shao et al., 2024) determines the direction of learning from relative rewards for responses to the same prompt. Converting rubric judgments into binary verdicts and aggregating them can assign equal rewards to responses that the judge supports to different degrees. When every response in a group ties, all advantages vanish, leaving a *dead group*. Adding more criteria does not resolve this problem if the distinctions they capture are lost during reward construction.

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.

The challenge is therefore to determine which differences in the available feedback should guide learning. Using continuous judge signals directly as rewards can reduce the information lost through discretization. Yet differences in judgment strength need not be proportional to differences in response quality. When multiple rubrics are aggregated, extreme values can overshadow distinctions captured by other criteria. A suitable scoring transformation should preserve continuous differences while controlling their relative influence.

We study *nonlinear score shaping before rubric aggregation*. Heat-GRPO smooths hard rubric verdicts under Gaussian margin perturbations, yielding the standard Gaussian cumulative distribution function (CDF). Applied before aggregation (Figure 1), it preserves distinctions near the verdict threshold while reducing sensitivity at the extremes. This changes rubric trade-offs and can reorder responses even when no group is dead. Identical-response comparisons isolate ranking changes; separate training experiments evaluate performance. Our contribution is an empirical analysis of how nonlinear scoring of rubric-level judge margins changes group-relative learning; Gaussian smoothing provides a simple realization of this design choice.

Our main contributions are as follows.

- **Nonlinear scoring before aggregation.** We empirically analyze how shaping rubric-level judge margins changes group-relative learning. Heat-GRPO realizes this design with Gaussian smoothing of hard verdicts, providing an interpretable bandwidth for controlling rubric contributions.
- **Analysis and controlled validation.** Identical-response comparisons isolate changes in rubric contributions and rankings; Hard/Heat/Linear controls under shared scalar normalization measure scorer effects on learning. We also characterize dead-group amplification during batch normalization.
- **Performance and computational efficiency.** In the complete-configuration comparison, Heat-GRPO leads in 23 of 24 conditions under the primary judge and at least 20 of 24 under each additional judge. Advantage computation takes 38.9% and 38.2% less time than GDPO and GD^{2}PO, respectively.

2 Related Work

2.1 Multi-Objective and Multi-Rubric Reinforcement Learning

Multi-objective reinforcement learning (MORL) studies the combination of multiple rewards and trade-offs among objectives (Roijers et al., 2013). Language-model alignment learns rewards from human feedback (Ouyang et al., 2022) or optimizes preferences directly (Rafailov et al., 2023). Rewarded Soups interpolates models trained for different objectives (Rame et al., 2023), while Fine-Grained RLHF and ArmoRM use detailed feedback and dimension-specific rewards (Wu et al., 2023; Wang et al., 2024). TLCR assigns continuous token rewards using discriminator confidence (Yoon et al., 2024). Multi-rubric learning instead expresses evaluation criteria in natural language: Rubrics as Rewards (RaR) uses either aggregated rubric verdicts or a score based on the complete rubric for policy learning (Gunjal et al., 2026). We analyze how nonlinear transformations before rubric aggregation affect relative advantages and learning outcomes.

2.2 LLM-as-a-Judge

LLM-as-a-Judge automates qualitative evaluation that is difficult to perform with rules and costly to repeat with human annotators (Zheng et al., 2023). Reinforcement learning from AI feedback uses such judgments to improve policies (Lee et al., 2024). Prometheus evaluates user-specified rubrics (Kim et al., 2024a); Prometheus 2 supports both direct assessment and pairwise ranking (Kim et al., 2024b). G-Eval incorporates output-token probabilities into scores (Liu et al., 2023). We fix the training judge and study how its rubric-level judgments become relative learning signals.

2.3 Group-Relative Optimization and Advantage Normalization

REINFORCE-style optimization can reduce the cost of learning from language-model feedback (Ahmadian et al., 2024). GRPO compares rewards within each prompt's response group with-

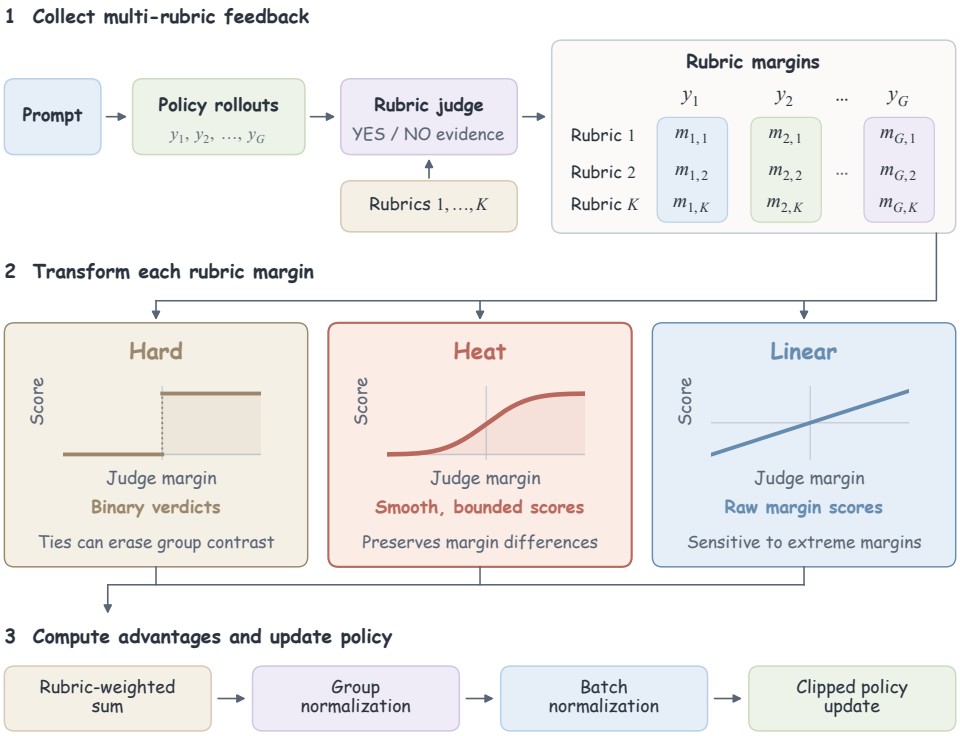

Figure 1: Overview of multi-rubric GRPO. Each rollout is evaluated against multiple rubrics, yielding a matrix of judge margins. Rubric-level scores are aggregated and normalized to compute advantages for policy optimization. The middle row contrasts Hard, Heat, and Linear scoring; Heat-GRPO uses Heat scoring.

out a separate critic (Shao et al., 2024). DAPO dynamically excludes groups that are all correct or all incorrect (Yu et al., 2025), while Dr. GRPO analyzes normalization biases involving length and reward variance (Liu et al., 2025). These reasoning-task analyses motivate studying multi-rubric aggregation and channel interactions. Group reward-Decoupled Normalization Policy Optimization (GDPO) normalizes rewards channel-wise (Liu et al., 2026b), and Group-Dynamic reward-Decoupled Policy Optimization (GD^{2}PO) filters conflicting channel signals (Liu et al., 2026a). Heat-GRPO transforms rubric scores before normalization.

3 Method: Score Shaping before Rubric Aggregation

Figure 1 summarizes the pipeline. Heat-GRPO shapes individual rubric judgments with a simple Gaussian CDF before aggregation, then normalizes the resulting response scores for policy optimization.

3.1 Multi-Rubric Judge Feedback

For each of B prompts $x \in \mathcal{X}$, the rollout policy $\pi_{\theta_{\text{old}}}$ samples G responses y_{xi}. A judge evaluates each response against K_x rubrics. Its YES and NO probabilities at the same decision position define the margin

$$m_{xik} = \log p^Y_{xik} - \log p^N_{xik} = \ell^Y_{xik} - \ell^N_{xik}, \tag{1}$$

For observed verdict tokens, ℓ^Y and ℓ^N share a vocabulary softmax. Appendix B.2 specifies token selection and missing-token imputation. Each rubric has a signed weight c_{xk} and validity mask $M_{xk} \in \{0, 1\}$, shared across responses to the prompt. Positive weights reward YES; negative weights penalize it.

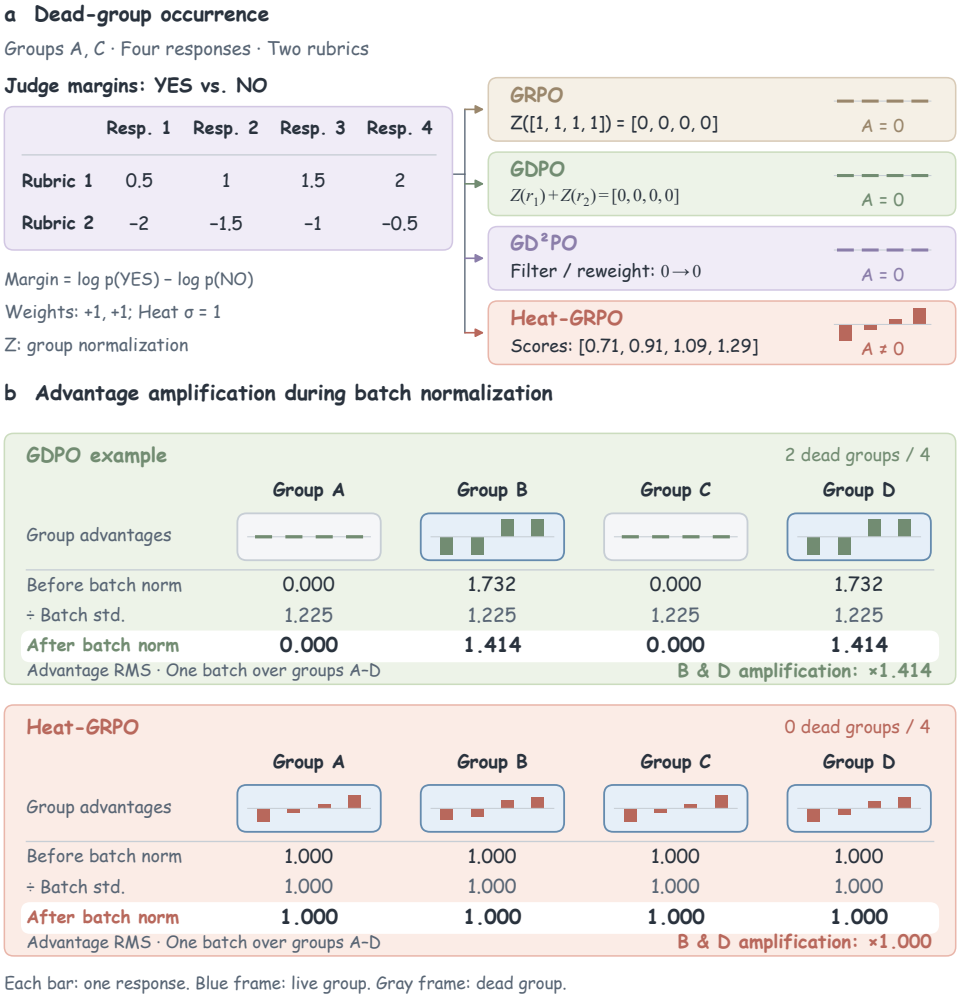

Figure 2: Constructed comparison of baselines (GRPO, GDPO, GD^{2}PO) and Heat-GRPO. (a) Heat scoring preserves contrast in groups with identical hard verdicts. (b) Amplification compares B and D after batch normalization against an all-live reference replacing A and C with copies of B and D. GDPO uses sample standard deviations per rubric; Heat-GRPO uses population group normalization. Both use population batch normalization. Appendix A.7 details the Heat-GRPO calculation.

The training interface encodes margins as $p = \mathrm{sigmoid}(m/8)$. In our comparison, GRPO aggregates hard verdicts $\mathbf{1}\{p > 0.5\} = h(m)$ into a signed, rubric-weighted reward; GDPO and GD^{2}PO receive binary utilities aligned with the preferred verdict. Binary inputs are our comparison setting; GDPO is not restricted to binary rewards (Liu et al., 2026b). Heat-GRPO recovers margins from continuous aligned utilities before scoring (Appendix A.3). Thus, shared source feedback does not imply equal information at the estimator input.

3.2 Heat-Kernel Construction

We extend the hard rubric map $h(m) = \mathbf{1}\{m > 0\}$ by averaging verdicts under zero-mean Gaussian margin perturbations. With kernel K_{σ^2} of variance σ^2, this standard heat construction gives

$$q_\sigma(m) = (K_{\sigma^2} * h)(m) = \mathbb{E}_{\eta\sim\mathcal{N}(0,\sigma^2)}[h(m+\eta)] = \Phi\left(\frac{m}{\sigma}\right), \tag{2}$$

where Φ is the standard normal CDF. On the probability scale, this is $\Phi(8\,\mathrm{logit}(p)/\sigma)$: the temperature 8 encodes the margin, while σ controls score shaping. We evaluate the CDF directly with-

out sampling. Successive Gaussian convolutions add their variances, giving $\tau = \sigma^2$ an explicit diffusion-variance interpretation (Appendix A.1).

Sensitivity at finite bandwidth. The score is smooth, bounded between zero and one, and strictly increasing in the margin. Its local sensitivity is

$$\frac{\partial q_\sigma(m)}{\partial m} = \frac{1}{\sigma}\phi\Big(\frac{m}{\sigma}\Big) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{m^2}{2\sigma^2}\right), \tag{3}$$

where ϕ is the standard normal density. This sensitivity profile defines the scoring bias: preserve distinctions near the verdict threshold while rapidly attenuating sensitivity to further changes in extreme margins. Applied before weighted aggregation, it controls how rubric differences enter relative advantages. The derivative describes scorer sensitivity, not a policy gradient through the judge.

Bandwidth and the boundary scorers. As $\sigma \downarrow 0$, scores approach hard verdicts for $m \neq 0$. At large bandwidths, exact group standardization removes the shared offset and scale, recovering normalized Linear advantages for nondegenerate groups without a dead-group threshold (Appendix A.2). Finite bandwidth instead changes relative score differences.

Training schedule. The annealed schedule uses $\sigma_t = 0.4047\sqrt{138 - t}$, where $t = 1, \ldots, 138$ indexes the update being performed. Thus $\sigma_1 \simeq 4.7369$; the final update ($t = 138$, $\sigma_t = 0$) uses Hard scoring on aligned margins, with ties mapped to zero (Appendix A.3). This progressively sharpens scoring; Appendix B.5 details the parameterization.

3.3 Advantage Computation and Policy Update

For $\sigma > 0$, we transform each margin before aggregation, retaining the rubric weights and their signs:

$$\begin{aligned} q^{(\sigma)}_{xik} &= q_\sigma(m_{xik}), \qquad S^{(\sigma)}_{xi} = \sum_{k=1}^{K_x} M_{xk} c_{xk} q^{(\sigma)}_{xik}, \\ A^g_{xi} &= Z_g(S^{(\sigma)}_{x\cdot})_i, \qquad A_{xi} = \mathcal{B}(A^g)_{xi}. \end{aligned} \tag{4}$$

A negative weight penalizes a high YES score; aligning its margin with NO and using a positive weight gives equivalent centered scores (Appendix A.3). Exact normalization removes shared offsets and scales, but preserves the relative gaps and rankings changed by nonlinear scoring (Appendix A.4). Writing $S_{xi} = S^{(\sigma)}_{xi}$, group normalization is

$$\begin{aligned} \bar{S}_x &= \frac{1}{G}\sum_{j=1}^{G} S_{xj}, \qquad s_x^2 = \frac{1}{G}\sum_{j=1}^{G}(S_{xj} - \bar{S}_x)^2, \\ A^g_{xi} &= \begin{cases} (S_{xi} - \bar{S}_x)/s_x, & s_x \geq \delta_g, \\ 0, & s_x < \delta_g, \end{cases} \qquad \delta_g = 10^{-9}. \end{aligned} \tag{5}$$

Responses above and below the group mean receive positive and negative advantages, respectively. The operator $\mathcal{B}$ standardizes all BG advantages, including dead-group zeros (Appendix A.5); an all-zero batch returns zeros. Figure 2 illustrates both effects: Heat scoring retains contrast in Groups A and C, avoiding the amplification of B and D caused by dead groups in the baseline example.

All valid tokens of a response share its advantage in the clipped GRPO/PPO objective (Schulman et al., 2017; Shao et al., 2024). Judge feedback and advantages are held fixed; gradients pass through the policy probabilities (Appendix A.8).

4 Experimental Setup

4.1 Models, Benchmarks, and Baselines

We evaluate Llama-3.2-1B/3B (Meta, 2024b), Llama-3.1-8B (Meta, 2024a), and Qwen3-1.7B/4B/8B (Qwen Team, 2025) on RaR-Science, RaR-Medicine (Gunjal et al., 2026), BeaverTails (Ji et al., 2023), and HealthBench (Arora et al., 2025). Across these 24 model–benchmark

pairs, we compare Heat-GRPO with GRPO (Shao et al., 2024), GDPO (Liu et al., 2026b), and GD^{2}PO (Liu et al., 2026a), using three seeds per method and pair. Each benchmark uses 1,500 training and 611 evaluation examples; Table 1 summarizes original sizes and rubric structures. Appendices B.1 and C detail subset construction and rubric examples.

Table 1: Benchmark overview. Dataset sizes refer to the original releases, with the number used for training shown in parentheses. Rubric counts are means (ranges) in our experimental subsets. Points are signed rubric weights.

Benchmark	Domain	Original dataset size			Rubrics / example		Points
		Train	Val	Test	Train	Eval	
RaR-Science	Science	18,333 (1,500)	2,292	2,292	7.51 (7–12)	7.46 (7–11)	$+1, -1$
RaR-Medicine	Medicine	17,926 (1,500)	2,240	2,242	9.58 (6–17)	9.57 (6–15)	$+1, -1$
BeaverTails	Helpfulness and safety	300,567 (1,500)	—	33,396	16 (fixed)	16 (fixed)	$+1, +2, -3$
HealthBench	Healthcare	— (1,500[†])	—	5,000[†]	11.29 (6–16)	11.05 (6–16)	-10 to $+10$

Evaluation subset used in this study: 611 examples per benchmark.

[†]HealthBench's 1,500 training examples are taken from the original 5,000-conversation collection listed under Test; they are not an additional source split. BeaverTails counts refer to prompt–response pairs in the full 330k release.

4.2 Training and Evaluation Protocol

Using official baseline implementations, Table 2 compares complete configurations in a shared verl-based GRPO pipeline (Sheng et al., 2024). Estimator representations and advantage rules follow Section 3; data, rollout and judge settings, policy objective, optimizer, training budget, and evaluation are otherwise fixed within each model–benchmark pair (Appendix B.3).

Qwen3-32B (Qwen Team, 2025) supplies training feedback and main evaluation scores at the final update (138), averaged over 611 examples and three seeds. Scores are signed, rubric-weighted YES verdicts normalized by total positive weight (Appendix B.1). We also evaluate responses with three additional open-weight judges at approximately the 20–30B scale: Qwen3.8-27B (Qwen Team, 2026), gpt-oss-20b (OpenAI, 2025), and Gemma-4-31B (Gemma Team, 2026). Appendix D compares their rankings.

4.3 Ablation Studies and Additional Analyses

Finite-bandwidth scoring. Table 4 isolates scorer effects by comparing Hard, fixed-bandwidth Heat, and Exact Linear on Qwen3-1.7B / BeaverTails with three seeds and shared scalar normalization. Hard tests discretization; Heat versus Linear tests nonlinear shaping of continuous margins. The annealed reference separately examines sharpening during training (Section 6.1).

Advantage computation and credit allocation. Section 6.2 relates the fraction of live groups to amplification during batch normalization. Separately, applying scorers to identical responses holds policy outputs and judge feedback fixed, isolating changes in rubric contributions and response rankings (Section 6.1; details in Appendix B.7).

Learning dynamics. We track BeaverTails evaluation reward over 138 updates for all four methods and six models (seed 0) to assess when performance differences emerge (Section 6.3). Gradient norms characterize policy-gradient magnitude (Appendix B.9); advantage-computation time quantifies computational cost (Section 5.2).

Table 2: Complete configurations: signed rubric scores under Qwen3-32B (three-seed means). Bold denotes the highest mean. Bottom: Heat-GRPO first-place counts under four judges across 24 conditions. Additional judges use one or two runs per condition; Appendix D gives the benchmark breakdown.

Model	Method	RaR-Science	RaR-Medicine	BeaverTails	HealthBench
Llama-3.2-1B	GRPO	0.1956	0.3342	0.6255	0.1229
	GDPO	0.2204	**0.3537**	0.5889	0.1442
	GD^{2}PO	0.1877	0.3411	0.6189	0.1340
	Heat-GRPO	**0.2360**	0.3435	**0.7657**	**0.1758**
Llama-3.2-3B	GRPO	0.3730	0.5263	0.7458	0.3683
	GDPO	0.3691	0.5352	0.7327	0.3437
	GD^{2}PO	0.3673	0.5372	0.7267	0.3375
	Heat-GRPO	**0.4662**	**0.5387**	**0.8061**	**0.3977**
Llama-3.1-8B	GRPO	0.5102	0.6495	0.8633	0.4927
	GDPO	0.4841	0.6256	0.8101	0.5478
	GD^{2}PO	0.5374	0.6635	0.8391	0.5279
	Heat-GRPO	**0.5464**	**0.6661**	**0.9337**	**0.5549**
Qwen3-1.7B	GRPO	0.5310	0.4476	0.5510	0.3796
	GDPO	0.5312	0.4526	0.6440	0.3713
	GD^{2}PO	0.5337	0.4453	0.6109	0.3869
	Heat-GRPO	**0.5644**	**0.4606**	**0.7403**	**0.3958**
Qwen3-4B	GRPO	0.6486	0.6056	0.7580	0.4891
	GDPO	0.6499	0.6050	0.7861	0.5245
	GD^{2}PO	0.6396	0.6050	0.7924	0.5141
	Heat-GRPO	**0.6680**	**0.6101**	**0.8972**	**0.5475**
Qwen3-8B	GRPO	0.6658	0.6613	0.7834	0.5376
	GDPO	0.6698	0.6564	0.8129	0.5484
	GD^{2}PO	0.6743	0.6550	0.8017	0.5536
	Heat-GRPO	**0.6906**	**0.6766**	**0.9010**	**0.5792**
Evaluation judge		Qwen3-32B	Qwen3.8-27B	gpt-oss-20b	Gemma-4-31B
Heat-GRPO first / 24 conditions		23/24	22/24	20/24	21/24

Table 3: Advantage computation time. Ranges summarize observed timings. Reduction is Heat-GRPO's relative decrease in mean time against each baseline.

Method	Range (ms/step)	Mean (ms/step)	Time reduction
GRPO	160.7–185.9	172.7	1.4%
GDPO	242.1–316.4	278.9	38.9%
GD^{2}PO	217.8–300.9	275.7	38.2%
Heat-GRPO	157.5–182.6	**170.3**	—

5 RESULTS

5.1 MAIN RESULTS

Table 2 compares complete method configurations with different estimator inputs; Table 4 separately controls scorer effects. Heat-GRPO achieves the highest three-seed mean in 23 of 24 conditions, exceeding GRPO and GD^{2}PO in all 24 and GDPO in 23. Its mean improvements are 0.05401, 0.04810, and 0.04714 over GRPO, GDPO, and GD^{2}PO, respectively. Against the strongest baseline in each condition, the mean and median differences are 0.03710 and 0.02055. GDPO leads on Llama-3.2-1B / RaR-Medicine. These are configuration-level gains; all-continuous baseline inputs are not tested.

Table 4: Scorer controls under shared scalar normalization on Qwen3-1.7B / BeaverTails, averaged over three seeds. Above the divider: fixed-bandwidth and boundary scorers. Below: the annealed reference from Table 2. Bold marks the best fixed bandwidth and the annealed reference.

Scorer / bandwidth	Mean
Hard (matched scalar normalization; $\sigma = 0$)	.5510
Narrow ($\sigma = 0.4047$)	.5968
$\sigma = 1.5$	.6487
Broad ($\sigma = 4.7541$)	**.6900**
$\sigma = 6.5$	.6386
$\sigma = 8.0$	.4967
$\sigma = 10$	.3734
$\sigma = 1000$ (linear proxy)	.4051
Exact Linear	.3895
Annealed Heat-GRPO (from Table 2)	**.7403**

Mean gains over the strongest baseline are 0.03048 on RaR-Science, 0.00362 on RaR-Medicine, 0.09335 on BeaverTails, and 0.02093 on HealthBench. Under every judge, Heat-GRPO leads in at least 20 of 24 conditions (Table 2); Appendix D reports first-place counts by benchmark.

5.2 Advantage Computation Cost

Heat-GRPO requires 170.3 ms per advantage-computation step, comparable to GRPO's 172.7 ms and 38.9% and 38.2% shorter than GDPO and GD^{2}PO, respectively (Table 3). All methods scale linearly with rubric observations; vectorized scoring and scalar aggregation avoid per-channel normalization and conflict-processing overhead. Timings exclude generation and judging (Appendix B.6).

6 Analysis

6.1 Finite Bandwidth beyond Hard and Linear Scoring

With scalar normalization held fixed, Table 4 compares scorers on Qwen3-1.7B / BeaverTails. Hard uses aligned margins with $h(z) = \mathbf{1}\{z > 0\}$, mapping zero-margin ties to zero (Appendix A.3). Broad scores 0.6900, exceeding Hard by 0.1390 and Exact Linear by 0.3005. Heat and Linear both retain continuous margins, isolating nonlinear shaping. Narrow reaches 0.5968, showing that bandwidth also matters. Annealed Heat-GRPO adds 0.0503 over fixed Broad, suggesting a benefit from sharpening scores during training (Appendix B.5).

On identical Broad-generated responses, Broad and Linear have 50.8% top-response overlap and 19.34% pairwise reversals; on Linear-generated responses, these are 47.6% and 22.94%. This establishes ranking changes, not greater ranking accuracy; Table 4 separately assesses learning gains. Appendix B.7 details these comparisons and Figure 4.

6.2 Dead Groups and Batch Normalization

Proposition (live-group fraction). For equal-sized groups, let normalized live-group advantages have zero mean and unit population variance, and dead-group advantages be zero. With $\rho_{\text{live}} = B_{\text{live}}/B > 0$,

$$s^2_{\text{batch}} = \frac{1}{BG}\sum_{x,i}(A^g_{xi})^2 = \rho_{\text{live}}, \qquad A^{\text{final}} = \frac{A^g}{\sqrt{\rho_{\text{live}}}}. \tag{6}$$

Half-dead batches increase surviving advantages by $\sqrt{2}$ relative to the all-live reference in Figure 2; all-dead batches return zeros. This relation concerns scalar advantage normalization, not optimizer-update magnitude (proof and run summaries in Appendix A.6).

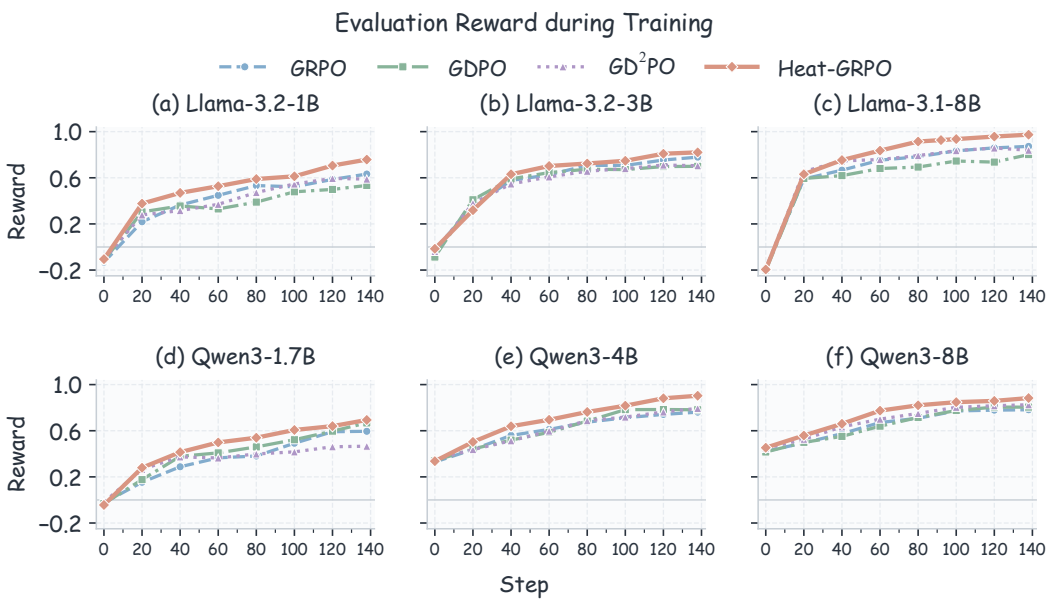

Figure 3: BeaverTails evaluation reward across six models (seed 0). Scores average one greedy response per prompt over the same 611 evaluation prompts at each checkpoint. All panels share axis scales; markers show measurements connected without smoothing.

6.3 Evaluation Reward during Training

Figure 3 tracks the signed BeaverTails reward, which combines helpfulness, engagement, and 14 penalized content categories. Its range is $[-14, 1]$, with higher values indicating a better aggregate rubric score (Appendix B.8). Heat-GRPO attains the highest final evaluation reward on all six models and leads at every shared evaluation checkpoint from update 40 onward. These single-seed trajectories complement the three-seed results in Table 2; corresponding gradient norms appear in Appendix B.9.

At the final checkpoint, Heat-GRPO exceeds the strongest baseline by 0.0256–0.1255 reward units across the six models. The closest competitor varies: GRPO attains the highest baseline reward on the three Llama models, GDPO on Qwen3-1.7B, and GD^{2}PO on Qwen3-4B and Qwen3-8B. For example, Heat-GRPO reaches 0.9738 on Llama-3.1-8B, compared with 0.8723 for GRPO, and 0.9034 on Qwen3-4B, compared with 0.7938 for GD^{2}PO. The final advantage thus spans both model families and different baseline learning trajectories. These values describe the single-seed curves in Figure 3.

The ordering also changes during early training. On Llama-3.2-3B, GDPO leads at update 20, but Heat-GRPO overtakes it by update 40 and remains ahead thereafter; Llama-3.1-8B similarly begins with GD^{2}PO ahead at update 20. Later gaps vary with the model and checkpoint: on Qwen3-1.7B the lead over the strongest baseline narrows from 0.0906 at update 60 to 0.0256 at the final update, whereas on Llama-3.2-1B it widens from 0.0791 to 0.1255. The curves therefore distinguish when the advantage emerges from how large it is at the endpoint.

7 Conclusion

We empirically analyzed nonlinear scoring of rubric-level judge margins before aggregation, with Gaussian smoothing as a simple realization. Identical-response comparisons isolate ranking changes; scalar-normalization controls show gains over Hard and Linear scoring. Across tested configurations, Heat-GRPO leads in 23 of 24 conditions under the primary judge and at least 20 of 24 under each additional judge, while reducing advantage computation time relative to GDPO and GD^{2}PO. These results support score shaping before aggregation as a practical design choice.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript writing, editing, and translation. The authors take responsibility for the final content.

REPRODUCIBILITY STATEMENT

Section 3 specifies the scoring transform and advantage computation. Appendix A provides derivations, assumptions, and the Heat-GRPO calculation for Figure 2. Appendix B describes dataset subsets, scoring conventions, training hyperparameters, hardware, bandwidth configuration, and analysis metrics. Appendix C gives benchmark questions and rubric examples; Appendix D defines the cross-judge rank comparison.

REFERENCES

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to Basics: Revisiting REINFORCE-Style Optimization for Learning from Human Feedback in LLMs. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12248–12267, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.662. URL https://aclanthology.org/2024.acl-long.662/.

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. HealthBench: Evaluating Large Language Models Towards Improved Human Health. *arXiv preprint arXiv:2505.08775*, 2025. URL https://arxiv.org/abs/2505.08775.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep Reinforcement Learning from Human Preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

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. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/cfd7bee7a651ee9af525098ef67a9e45-Paper-Conference.pdf.

Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. BeaverTails: Towards Improved Safety Alignment of LLM via a Human-Preference Dataset. In *Advances in Neural Information Processing Systems*, volume 36, pp. 24678–24704. Curran Associates, Inc., 2023. doi: 10.52202/075280-1072. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/4dbb61cb68671edc4ca3712d70083b9f-Paper-Datasets_and_Benchmarks.pdf.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. Prometheus: Inducing Fine-Grained Evaluation Capability in Language Models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 29927–29962, 2024a. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/803485352e61e3ebf41221e4776c9fd4-Paper-Conference.pdf.

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 Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 4334–4353, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.248. URL https://aclanthology.org/2024.emnlp-main.248/.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Ren Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 26874–26901. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/lee24t.html.

Haotian Liu, Yihao Liu, Jingwei Ni, Siyuan Huang, Xinpeng Liu, Pengyu Cheng, Jiajun Song, Ruijin Ding, Junfeng Li, Zhechao Yu, Mengyu Zhou, Hongteng Xu, Xiaoxi Jiang, and Guanjun Jiang. GD^{2}PO: Mitigating Multi-Reward Conflicts via Group-Dynamic reward-Decoupled Policy Optimization. *arXiv preprint arXiv:2606.16771*, 2026a. URL https://arxiv.org/abs/2606.16771.

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, Yejin Choi, Jan Kautz, and Pavlo Molchanov. GDPO: Group reward-Decoupled Normalization Policy Optimization for Multi-reward RL Optimization. In *Proceedings of the International Conference on Machine Learning*, 2026b. URL https://icml.cc/virtual/2026/poster/63333.

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. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.153. URL https://aclanthology.org/2023.emnlp-main.153/.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-Like Training: A Critical Perspective, 2025. URL https://arxiv.org/abs/2503.20783.

Meta. Llama 3.1 Model Card. Official model card, 2024a. URL https://github.com/meta-llama/llama-models/blob/main/models/llama3_1/MODEL_CARD.md.

Meta. Llama 3.2 Model Card. Official model card, 2024b. URL https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/MODEL_CARD.md.

OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL https://arxiv.org/abs/2508.10925.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll 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. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744. Curran Associates, Inc., 2022. doi: 10.52202/068431-2011. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.

Qwen Team. Qwen3 Technical Report, 2025. URL https://arxiv.org/abs/2505.09388.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8. Evaluated model: Qwen3.8-27B. Model card: https://huggingface.co/Qwen/Qwen3.8-27B.

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. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 53728–53741. Curran Associates, Inc., 2023. doi: 10.52202/075280-2338. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/a85b405ed65c6477a4fe8302b5e06ce7-Paper-Conference.pdf.

Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards Pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. In *Advances in Neural Information Processing Systems*, volume 36, pp. 71095–71134. Curran Associates, Inc., 2023. doi: 10.52202/075280-3114. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/e12a3b98b67e8395f639fde4c2b03168-Paper-Conference.pdf.

Diederik M. Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48:67–113, 2013. doi: 10.1613/jair.3987. URL https://www.cs.ox.ac.uk/people/shimon.whiteson/pubs/roijersjair13.pdf.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms, 2017. URL https://arxiv.org/abs/1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*, 2024. URL https://arxiv.org/abs/2409.19256.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable Preferences via Multi-Objective Reward Modeling and Mixture-of-Experts. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10582–10592, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.620. URL https://aclanthology.org/2024.findings-emnlp.620/.

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-Grained Human Feedback Gives Better Rewards for Language Model Training. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 59008–59033. Curran Associates, Inc., 2023. doi: 10.52202/075280-2574. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/b8c90b65739ae8417e61eadb521f63d5-Paper-Conference.pdf.

Eunseop Yoon, Hee Suk Yoon, SooHwan Eom, Gunsoo Han, Daniel Nam, Daejin Jo, Kyoung-Woon On, Mark Hasegawa-Johnson, Sungwoong Kim, and Chang Yoo. TLCR: Token-Level Continuous Reward for Fine-grained Reinforcement Learning from Human Feedback. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14969–14981, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.889. URL https://aclanthology.org/2024.findings-acl.889/.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang,

Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An Open-Source LLM Reinforcement Learning System at Scale. In *Advances in Neural Information Processing Systems*, volume 38, pp. 113222–113244. Curran Associates, Inc., 2025. doi: 10.52202/085713-3775. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/a4277440d50f1f15d2cb4c14f7e0c0d2-Paper-Conference.pdf.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623. Curran Associates, Inc., 2023. doi: 10.52202/075280-2020. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/91f18a1287b398d378ef22505bf41832-Paper-Datasets_and_Benchmarks.pdf.

A Mathematical Details

A.1 Heat Smoothing of a Hard Threshold

We summarize the standard Gaussian smoothing identities used to describe our scorer. Hard scoring and the Gaussian kernel are defined as follows:

$$h(m) = \mathbf{1}\{m > 0\}, \qquad S_{xi}^{\text{hard}} = \sum_{k=1}^{K_x} M_{xk} c_{xk} h(m_{xik}). \tag{7}$$

$$K_\tau(z) = \frac{1}{\sqrt{2\pi\tau}} \exp\left(-\frac{z^2}{2\tau}\right), \qquad \tau = \sigma^2 > 0. \tag{8}$$

The Heat scorer is the convolution of a hard threshold with a Gaussian kernel. Averaging hard verdicts under Gaussian perturbations of the margin yields the same scorer as solving the heat equation with the hard threshold as its initial condition. The bandwidth is the standard deviation of the perturbation, and diffusion time is its variance. This construction defines a family of smoothing levels starting from the original discrete verdict.

$$\begin{aligned} u(m,\tau) = (K_\tau * h)(m) &= \int_{\mathbb{R}} K_\tau(z) \mathbf{1}\{m - z > 0\} \, \mathrm{d}z \\ &= \int_{-\infty}^{m} K_\tau(z) \, \mathrm{d}z = \Phi(m/\sqrt{\tau}) = q_{\sqrt{\tau}}(m), \qquad \tau > 0. \end{aligned} \tag{9}$$

By symmetry of the Gaussian kernel, averaging $h(m - z)$ is equivalent to taking the expectation of $h(m + \eta)$. Under the Brownian convention, this function satisfies

$$\partial_\tau u = \tfrac{1}{2} \partial_{mm} u, \qquad \lim_{\tau \downarrow 0} u(m,\tau) = h(m) \quad (m \neq 0), \qquad K_{\tau_1} * K_{\tau_2} = K_{\tau_1 + \tau_2} \tag{10}$$

The limit recovers the initial condition h away from the discontinuity at $m = 0$, where the hard-branch convention is specified separately.

Successive Gaussian smoothing operations add the perturbation variances, giving the convolution operators a semigroup structure in diffusion time. This property describes the chosen smoothing model and its scale parameter; it is neither an assumption nor an empirical claim that judge errors are Gaussian.

Equation (3) follows by differentiating the Gaussian CDF in Equation (9) with respect to the margin.

A.2 Hard and Normalized Linear Boundaries

The Linear scorer uses raw margins directly, with scores and advantages defined by

$$q_{\text{lin}}(m) = m, \qquad S_{xi}^{\text{lin}} = \sum_{k=1}^{K_x} M_{xk} c_{xk} m_{xik}, \qquad A^{\text{lin}} = \mathcal{B}\big(Z_g(S^{\text{lin}})\big). \tag{11}$$

As the bandwidth approaches zero, Heat scores converge to hard verdicts away from the decision boundary. At exactly zero margin, a separate tie convention is required: the midpoint of the finite-bandwidth transform and the implemented hard branch need not coincide.

As the bandwidth increases, Heat scores expand into a common constant and a small linear term. Aggregation with signed weights followed by normalization across responses to the same prompt removes the common constant and positive scale. The resulting advantages therefore converge to normalized Linear advantages for groups with nondegenerate weighted linear-margin scores. Very large finite bandwidths can reduce numerical differences below the dead-group threshold, so the boundary experiments use Exact Linear scoring of raw margins directly.

For fixed, finite margins and weights, a Taylor expansion gives

$$\begin{aligned} \Phi(m/\sigma) &= \tfrac{1}{2} + \frac{m}{\sigma\sqrt{2\pi}} + O(\sigma^{-3}), \\ S_{xi}^{(\sigma)} &= C_x + \frac{S_{xi}^{\text{lin}}}{\sigma\sqrt{2\pi}} + O(\sigma^{-3}), \qquad C_x = \tfrac{1}{2} \sum_k M_{xk} c_{xk}. \end{aligned} \tag{12}$$

Let Z_0 denote exact group standardization by a positive standard deviation, without a dead-group threshold. When $s_x^{\mathrm{lin}} > 0$, centering removes C_x, and the standard deviation expands as $s_x^{\mathrm{lin}}/(\sigma\sqrt{2\pi}) + O(\sigma^{-3})$. Hence,

$$\lim_{\sigma\to\infty} Z_0(S_{x\cdot}^{(\sigma)})_i = Z_0(S_{x\cdot}^{\mathrm{lin}})_i = \frac{S_{xi}^{\mathrm{lin}} - \bar{S}_x^{\mathrm{lin}}}{s_x^{\mathrm{lin}}}. \tag{13}$$

If every group in the batch satisfies this nondegeneracy condition, population group normalization gives unit batch variance, so the same convergence holds for the final advantages. Groups with constant Linear scores are excluded because higher-order terms can yield a different limit. Moreover, with the fixed threshold $\delta_g > 0$ in Equation (5), $s_x^{(\sigma)}$ can eventually fall below the threshold. This distinction separates the mathematical boundary defined through Z_0 from the implemented Exact Linear scorer.

A.3 Equivalent Representations of Signed Utilities

The training interface encodes a judge margin as $p_{xik} = \mathrm{sigmoid}(m_{xik}/8)$ and passes desirability-aligned utilities to the Heat-GRPO scorer:

$$u_{xik} = \begin{cases} p_{xik}, & c_{xk} \geq 0, \\ 1 - p_{xik}, & c_{xk} < 0, \end{cases} \qquad z_{xik} = 8\,\mathrm{logit}(u_{xik}).$$

For $0 < u_{xik} < 1$, exact arithmetic gives $z_{xik} = m_{xik}$ for positive weights and $z_{xik} = -m_{xik}$ for negative weights. The scorer applies $\Phi(z_{xik}/\sigma)$ with weights $|c_{xk}|$. Thus, its continuous input is an aligned utility, and the margin formulation in Section 3.2 expresses the same finite-bandwidth transformation.

For comparison, the GRPO training reward is $R_{xi} = D_x^{-1}\sum_k M_{xk}c_{xk}\mathbf{1}\{p_{xik} > 0.5\}$, where $D_x = \max\{\sum_k M_{xk}\max(c_{xk}, 0), 10^{-9}\}$. GDPO and GD^{2}PO receive $\mathbf{1}\{u_{xik} > 0.5\}$ as rubric channels. These are training-time transformations of the same judge feedback; final benchmark scores use generated verdicts as described in Appendix B.1.

Applying absolute weights to aligned Heat scores differs from the signed-margin representation only by a prompt-constant offset. For common validity masks and rubric weights across responses,

$$\begin{aligned} \widetilde{S}_{xi}^{(\sigma)} &= \sum_k M_{xk}|c_{xk}|\Phi\left(\frac{\mathrm{sign}(c_{xk})m_{xik}}{\sigma}\right) \\ &= S_{xi}^{(\sigma)} + \sum_{k:c_{xk}<0} M_{xk}|c_{xk}|, \qquad \sigma > 0. \end{aligned} \tag{14}$$

This follows from $\Phi(-z) = 1 - \Phi(z)$. The two scores have identical centered values and standard deviations, and hence identical group and batch advantages. At zero bandwidth, the scorer evaluates the aligned utilities directly:

$$\widetilde{S}_{xi}^{(0)} = \sum_k M_{xk}|c_{xk}|\mathbf{1}\{z_{xik} > 0\}.$$

Thus $u = 0.5$, including neutral fallback feedback, contributes zero in this branch. This aligned definition specifies both the Hard control in Table 4 and the zero-horizon branch in Equation (19); the finite-bandwidth signed equivalence is not applied at zero-margin ties. The Hard control shares group and batch normalization with the other scorers in Table 4; it denotes a scoring rule within that pipeline, whereas GRPO in Table 2 denotes the complete baseline method.

A.4 Prompt-wise affine invariance

Adding a shared constant and multiplying by a shared positive scale within a nondegenerate group leaves relative advantages unchanged after exact group standardization. This separates a uniform rescaling of response scores from a nonlinear transformation of individual rubric scores. Rescaling raw Linear scores alone cannot reproduce the changes in relative spacing or rankings induced by finite-bandwidth Heat scoring.

$$Z_0(a_x S_{x\cdot} + b_x \mathbf{1})_i = \frac{a_x S_{xi} + b_x - (a_x \bar{S}_x + b_x)}{a_x s_x} = Z_0(S_{x\cdot})_i, \qquad a_x > 0,\ s_x > 0. \tag{15}$$

Division by a positive prompt-specific denominator shared across responses likewise cancels under the same conditions.

In a numerical implementation, this invariance requires the live/dead classification to remain unchanged and assumes that denominator regularization or additional nonlinear operations do not prevent exact scale cancellation. Omitting a positive prompt-specific denominator is justified under these same conditions.

A.5 Group and Batch Normalization

Following group normalization in Equation (5), the batch operator includes all BG responses, with zeros from dead groups:

$$\begin{aligned} \bar{A}^g &= \frac{1}{BG} \sum_{x \in \mathcal{X}} \sum_{i=1}^{G} A^g_{xi}, \qquad s^2_{\text{batch}} = \frac{1}{BG} \sum_{x \in \mathcal{X}} \sum_{i=1}^{G} (A^g_{xi} - \bar{A}^g)^2, \\ A_{xi} &= \begin{cases} (A^g_{xi} - \bar{A}^g)/s_{\text{batch}}, & s_{\text{batch}} > 0, \\ 0, & s_{\text{batch}} = 0. \end{cases} \end{aligned} \tag{16}$$

This is response-level population normalization, rather than token-weighted whitening. In exact arithmetic, $\bar{A}^g = 0$. The threshold handles constant or numerically indistinguishable group scores, while the zero-variance branch handles an all-dead batch.

A.6 Proof of the Exact Live-Group Fraction Relation

After population normalization, each live group has zero mean and unit variance. For equal-sized groups, the sum of squared advantages is therefore proportional to the number of live groups. Dividing by the total number of responses makes the batch variance equal to the live-group fraction. Batch centering preserves the zero mean, and division by the batch standard deviation yields the amplification factor in the main text.

$$\begin{aligned} \sum_{i=1}^{G} A^g_{xi} = 0, \qquad \sum_{i=1}^{G} (A^g_{xi})^2 = G\, \mathbf{1}\{s_x \geq \delta_g\}, \\ s^2_{\text{batch}} = \frac{1}{BG} \sum_{x \in \mathcal{X}} \sum_{i=1}^{G} (A^g_{xi})^2 = \frac{1}{B} \sum_{x \in \mathcal{X}} \mathbf{1}\{s_x \geq \delta_g\} = \rho_{\text{live}}. \end{aligned} \tag{17}$$

Thus, $A_{xi} = A^g_{xi}/\sqrt{\rho_{\text{live}}}$ for $\rho_{\text{live}} > 0$. The factor is exactly one when every group is live and increases as the live-group fraction decreases.

For unequal group sizes, the relevant quantity is the fraction of responses belonging to live groups, rather than the unweighted fraction of groups. Token-weighted normalization requires a corresponding token-weighted relation. If denominator regularization prevents live groups from having unit variance, their actual variances must be aggregated. An all-dead batch returns zero advantages without division. The proposition characterizes advantage scaling along the specified scalar normalization path, not the magnitude of the complete optimizer update.

Observed minimum live-group fractions are 0.5312 for Hard and 0.9688 for Heat-GRPO, implying maximum amplification factors of 1.372 and 1.016. Heat-GRPO's factor remains close to one because almost all groups retain a nonzero signal. Hard uses the same scalar normalization as Heat-GRPO, isolating the scoring transformation.

A.7 Heat-GRPO Computation in Figure 2

The example in Figure 2 uses four responses per group, two rubrics with weights $+1$, and Heat bandwidth $\sigma = 1$. Groups A and C both use margin pairs $(0.5, -2)$, $(1, -1.5)$, $(1.5, -1)$,

and $(2, -0.5)$. Heat-GRPO transforms each margin through Φ and sums the rubric scores before population group normalization. The resulting advantages for A and C are approximately $[-1.346388, -0.432710, 0.432710, 1.346388]$, preserving contrast across all four responses.

Group B uses margin pairs $(-2, -2)$, $(-1, -1)$, $(1, 1)$, and $(2, 2)$. Its scores are $2\Phi([-2, -1, 1, 2]) \simeq [0.045500, 0.317311, 1.682689, 1.954500]$. Their mean is 1 and their population standard deviation is approximately 0.829800, giving advantages $[-1.150277, -0.822716, 0.822716, 1.150277]$. Group D uses $(-3, -2)$, $(-0.8, -0.4)$, $(0.6, 1.2)$, and $(1.5, 2.5)$, which also yield distinct aggregated scores.

Groups A–D form one batch of 16 responses. Each group has zero mean and unit population variance after group normalization, so the batch standard deviation is one. Batch normalization therefore preserves their advantages. To compute relative amplification, the all-live reference replaces A with a copy of B and C with a copy of D; its batch standard deviation is also one, leaving the signals for B and D unchanged.

The figure summarizes each group's advantage vector v by its root mean square, $R(v) = \sqrt{\frac{1}{4}\sum_{i=1}^{4} v_i^2}$. For Heat-GRPO, every group has RMS magnitude 1 before and after batch normalization. Calculations use full precision; values displayed in the figure are rounded to three decimal places.

A.8 Policy Optimization Objective

The advantages enter the clipped policy surrogate (Schulman et al., 2017; Shao et al., 2024). At token position u,

$$
\begin{aligned}
r_{xiu}(\theta) &= \frac{\pi_\theta(y_{xi,u} \mid x, y_{xi,<u})}{\pi_{\theta_{\text{old}}}(y_{xi,u} \mid x, y_{xi,<u})}, \qquad \widehat{A}_{xi} = \operatorname{stopgrad}(A_{xi}), \\
j_{xiu}(\theta) &= \min\Big\{ r_{xiu}(\theta)\widehat{A}_{xi},\ \operatorname{clip}(r_{xiu}(\theta), 1-\epsilon, 1+\epsilon)\,\widehat{A}_{xi} \Big\}.
\end{aligned}
\tag{18}
$$

where ϵ is the clipping parameter. All valid tokens of a response share its advantage. Judge feedback and advantages are held fixed; gradients are taken through the policy probabilities.

B Experimental Details

B.1 Dataset Subsets and Rubric Scoring

Each benchmark is evaluated using rubric criteria and signed weights. We describe the dataset subsets and scoring procedure below; Appendix C presents example questions and rubrics.

Dataset subsets. Each benchmark subset contains 1,500 training and 611 evaluation examples. The original train/validation/test sizes are 18,333/2,292/2,292 for RaR-Science and 17,926/2,240/2,242 for RaR-Medicine. The full BeaverTails release contains 300,567 training and 33,396 test prompt–response pairs; these counts should be distinguished from the number of unique prompts. Original split sizes follow the dataset releases (Gunjal et al., 2026; Ji et al., 2023). RaR subsets are sampled without replacement from the respective training and test splits with sampling seed 0. BeaverTails subsets use the first 1,500 training and 611 test examples after removing duplicate prompts.

HealthBench contains 5,000 conversations, including the 1,000-example Hard subset. Training examples are drawn from the 4,000 conversations outside Hard, whereas evaluation examples are drawn from Hard. Both subsets retain conversations with 6–16 rubrics and at most six conversation turns. After shuffling the eligible examples with seeds 1234 and 77, respectively, we select 1,500 training and 611 evaluation examples. Table 1 distinguishes original sizes from the subsets used here; rubric statistics describe the latter.

The same 611 BeaverTails evaluation prompts are used for monitoring during training (Appendix B.8); no additional validation split is used for these curves.

Training feedback and evaluation verdicts. Training feedback uses the judge margin through $p = \mathrm{sigmoid}(m/8)$. Benchmark evaluation uses YES indicators parsed from generated judge text. Response and conversation inputs to the judge are limited to 13,312 and 8,192 characters, respectively; token selection is specified in Appendix B.2.

Evaluation score. For evaluation, let v_{xk} be one when the generated judge verdict for rubric k is parsed as YES and zero otherwise. The response score is

$$\mathrm{score}(x) = \frac{\sum_k c_{xk} v_{xk}}{D_x}, \qquad D_x = \max\left\{ \sum_k \max(c_{xk}, 0),\ 10^{-9} \right\}.$$

The benchmark score is the arithmetic mean over its 611 evaluation examples. Its upper bound is one, while the lower bound depends on the negative rubric weights. In our BeaverTails adaptation, the fixed rubric contains 14 harmful-content categories with weight -3, engagement with weight $+1$, and helpfulness with weight $+2$. Thus, $D_x = 3$, the response score ranges from -14 to 1, and each harmful-content YES verdict reduces the score by one. These are signed rubric scores, not accuracy or percentages. Appendix C gives examples for all four benchmarks.

B.2 Judge Token Selection and Margin Extraction

Training queries return the top 20 token log probabilities at the verdict position. After trimming surrounding whitespace and converting token text to uppercase, tokens beginning with Y are candidates for YES and tokens beginning with N are candidates for NO. The selected value for each verdict is the largest returned log probability among its candidates:

$$L_v = \begin{cases} \max_{a \in \mathcal{T}_v} \log P(a), & \mathcal{T}_v \neq \varnothing, \\ -20, & \mathcal{T}_v = \varnothing, \end{cases} \quad v \in \{Y, N\}, \qquad m = L_Y - L_N, \quad p = \mathrm{sigmoid}(m/8).$$

The sets $\mathcal{T}_Y$ and $\mathcal{T}_N$ contain only tokens in the returned top-20 list. This is a maximum over prefix-matched candidates, not a sum of their probability masses. When both selected values are observed, their log-probability difference equals the corresponding logit difference under the common vocabulary softmax. A missing verdict uses the imputed log probability -20.

If neither candidate set is present, $m = 0$ and $p = 0.5$. Requests with transport or parsing failures use a retry limit of three and return $p = 0.5$ if unsuccessful. Heat-GRPO also replaces undefined utilities with 0.5 before advantage estimation. These neutral values denote missing feedback as well as possible judge ties.

There is no explicit probability clipping in this path. If both selected values lie in $[-20, 0]$, then $|m| \leq 20$ and

$$p \in [\mathrm{sigmoid}(-2.5), \mathrm{sigmoid}(2.5)] \simeq [0.0759, 0.9241].$$

This is a conditional range, not a global clipping interval: an observed candidate below -20 retains its reported value and can yield probabilities outside this interval. The temperature 8 is distinct from the Heat bandwidth and the policy sampling temperature.

B.3 Policy Training and Evaluation Settings

Table 5 lists the settings shared by the main benchmark experiments. We use the verl framework (Sheng et al., 2024) (version 0.9.0.dev), fully sharded data parallelism (FSDP), and vLLM 0.11.0 for asynchronous rollout generation. Each update contains 32 prompts with eight responses per prompt, giving 256 response sequences. Each optimization epoch over this batch performs one optimizer update. Three passes over the 1,500-example training subset yield $3\lfloor 1500/32 \rfloor = 138$ updates, with incomplete batches omitted. Gradients are accumulated over model-dependent micro-batches to preserve the effective batch size.

Table 5: Shared hyperparameters for the main benchmark experiments. Token limits apply to the policy model; judge input limits are specified separately.

Parameter	Value
Optimizer	AdamW
Learning rate	2×10^{-6} (constant; no warmup)
Adam coefficients	$\beta_1 = 0.9$, $\beta_2 = 0.999$
Weight decay	0.01
Gradient clipping	Global norm 1.0
Prompts per update / mini-batch	32 / 32
Responses per prompt (G)	8
Optimization epochs per update	1
Training epochs / updates	3 / 138
Policy probability-ratio clipping	$[0.8, 1.2]$
Loss aggregation	Mean over valid response tokens
Entropy coefficient	0
KL penalty (loss and reward)	None
Sampling temperature	1.0
Nucleus / top-k sampling	$p = 1.0$ / disabled
Maximum prompt / response length	1,280 / 2,048 tokens
Prompt truncation	Left
Precision	bfloat16
Gradient checkpointing / padding removal	Enabled / enabled
Independent training seeds	3 (main and bandwidth comparisons)
Rollout seed	42 + training seed
Evaluation decoding	Greedy; one response per prompt
Evaluation checkpoints	0, 20, 40, 60, 80, 100, 120, 138

B.4 Hardware and Training Time

Experiments run on servers equipped with NVIDIA H100 GPUs, each with 80 GB HBM3 memory. Shared Qwen3-32B instances provide rubric judgments during training. Table 6 reports elapsed training and evaluation time for each policy model.

Table 6: Median elapsed time per Heat-GRPO run in the main experiments, including training and evaluation.

Policy model	Time (h)
Llama-3.2-1B	4.05
Qwen3-1.7B	4.25
Llama-3.2-3B	4.77
Qwen3-4B	6.72
Llama-3.1-8B	4.88
Qwen3-8B	4.30

Judge serving. Qwen3-32B is served with a 4,096-token context limit and a GPU memory allocation target of 92%. Training-time rubric queries use deterministic decoding, a one-token output limit, and up to 128 concurrent requests. Appendix B.2 specifies token selection, the probability temperature, and failure handling; Appendix B.1 specifies the character-level input limits.

B.5 Bandwidth Configuration Used in the Main Experiments

The main experiments anneal the bandwidth with the remaining training horizon, using $\sigma = \sigma_t$ in Equation (4). The score improvement from 0.6900 with fixed Broad to 0.7403 with this schedule suggests an additional benefit from gradually sharpening the scoring transition (Section 6.1). At zero remaining horizon, the schedule explicitly uses the hard branch:

$$\sigma_t = \sigma_{\text{base}}\sqrt{H_t}, \qquad \tau_t = \sigma_t^2 = \sigma_{\text{base}}^2 H_t,$$
$$q_{\sigma_t}(z) = \begin{cases} \Phi(z/\sigma_t), & H_t > 0, \\ \mathbf{1}\{z > 0\}, & H_t = 0. \end{cases} \tag{19}$$

Here, $t \in \{1, \ldots, 138\}$ is the index of the update being performed, and $H_t = 138 - t$ counts updates remaining after that update. Its advantages use bandwidth σ_t: the first update has $H_1 = 137$ and $\sigma_1 \simeq 4.7369$; update 137 has $\sigma_{137} = 0.4047$; the final update has $H_{138} = 0$ and uses the Hard branch. The aligned margin is $z = 8\,\text{logit}(u)$ (Appendix A.3). The scale $\sigma_{\text{base}} = 0.4047$ is held fixed across the main comparisons. Fixed-bandwidth controls use Narrow at $H = 1$ and Broad at the nominal full horizon $H = 138$, giving $0.4047\sqrt{138} \simeq 4.7541$. At $z = 0$, finite-bandwidth scoring returns $1/2$, whereas the explicit Hard branch returns zero.

Margin-scale diagnostic. A separate check re-evaluated 908 rubric judgments from a baseline rollout at update 1 using the same judge protocol. Their margins had mean -3.2905, median -2.8750, and standard deviation $s(m_1) = 4.7491$, close to the Broad bandwidth 4.7541. For this fixed bandwidth, the ratio $\kappa = \sigma_{\text{Broad}}/s(m)$ was 1.0011 at update 1 and 1.3000 at update 90, where $s(m_{90}) = 3.6571$. These ratios characterize the fixed Broad bandwidth relative to the cross-sectional spread of judge margins over training.

B.6 Advantage Computation Timing

Table 3 reports milliseconds per advantage-computation call, excluding response generation and judge evaluation. For baseline mean time $\bar{t}_b$ and Heat-GRPO mean time $\bar{t}_H$, the relative reduction is $100(1 - \bar{t}_H/\bar{t}_b)\%$. The corresponding speed ratio, $\bar{t}_b/\bar{t}_H$, is 1.64 for GDPO and 1.62 for GD^{2}PO.

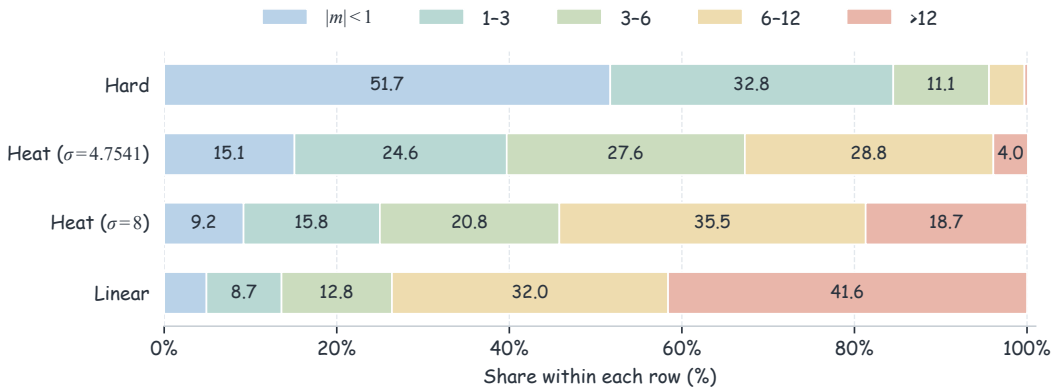

Figure 4: Within-group rubric-score contrast on identical BeaverTails responses. The four bars show centered contributions by absolute-margin range, expressed as percentages within each scorer. Each row is normalized separately, so bar widths compare relative allocation, not total signal magnitude.

B.7 Rubric Contributions and Response Rankings

This analysis complements the bandwidth comparison in Section 6.1. Table 4 measures the performance of policies trained with different scorers; Figure 4 instead holds the responses and judge feedback fixed to isolate how the scoring transformation redistributes within-group contrast. The comparisons use BeaverTails responses from policies trained with Heat scoring at $\sigma = 4.7541$ (Broad) and Exact Linear scoring, over 138 updates with seed 2. The Broad-generated set contains 4,416 groups and supplies the margins for Figure 4.

What is measured. For scorer q, the centered contribution of rubric k to response i is

$$\overline{q}_{xk} = \frac{1}{G}\sum_{j=1}^{G} q(m_{xjk}), \qquad d_{xik}(q) = \left|q(m_{xik}) - \overline{q}_{xk}\right|.$$

The mean is taken across responses to the same prompt, separately for each rubric. This per-rubric quantity is defined before multiplying by rubric weights or aggregating scores; it measures score contrast, not a policy-gradient contribution. For example, scores $(0.2, 0.4, 0.6, 0.8)$ have mean 0.5 and contrasts $(0.3, 0.1, 0.1, 0.3)$, whereas identical scores have zero contrast.

How to read the spectrum. Colors indicate absolute-margin intervals with boundaries 1, 3, 6, and 12. Each scorer's contributions are aggregated by interval and expressed as shares of its total. Margins with $|m| > 12$ account for 4.0% of centered contribution under Heat at $\sigma = 4.7541$, 18.7% under Heat at $\sigma = 8$, and 41.6% under Linear. Finite bandwidth therefore shifts relative contribution away from extreme margins. Hard assigns only 0.4% to this interval, but also discards all differences between margins with the same verdict. Reported percentages are rounded.

These shares quantify score contrast before rubric aggregation and advantage normalization. Each row is normalized independently to show how a scorer distributes its within-group contrast across margin ranges.

Response rankings. Top-response overlap is the fraction of groups for which two scorers select the same highest-scoring response; pairwise reversals describe changes in response ordering. On Broad-generated responses, Broad and Linear agree on the top response in 50.8% of groups, with 19.34% pairwise reversals and advantage cosine dissimilarity $1 - \cos(A_B, A_L) = 0.2159$. On Linear-generated responses, top-response overlap is 47.6% and pairwise reversals reach 22.94%. Although Heat scoring is monotone within each rubric, its nonlinear transformation changes trade-offs before rubric aggregation. Subsequent group normalization cannot reverse a changed response ordering.

Relation to other smooth threshold functions. Hard and normalized Linear limits are shared by other smooth threshold functions. For example, the logistic scorer with scale $a > 0$ satisfies

$$q_{\text{logistic}}(m; a) = \frac{1}{1 + e^{-m/a}} = \frac{1}{2} + \frac{m}{4a} + O(a^{-3}) \quad (a \to \infty),$$

for fixed m, and converges to $h(m)$ as $a \downarrow 0$ for $m \neq 0$. Under exact group standardization without a dead-group threshold, its shared offset and scale cancel, recovering normalized Linear advantages for nondegenerate weighted-margin groups. These common limits organize our controls; they do not uniquely characterize Gaussian scoring.

Heat-GRPO adopts Gaussian convolution of the hard rubric map, giving the bandwidth a diffusion-variance interpretation and the scorer a Gaussian sensitivity profile (Appendix A.1). The perturbations define the smoothing model, not a distributional assumption about judge errors. Similarly, signed-score equivalence follows from $q(-m) = 1 - q(m)$, which logistic scoring also satisfies; the live-group identity in Section 6.2 follows from normalization and applies to any scalar scorer satisfying its assumptions.

Our experiments evaluate the chosen construction against its boundary scorers, bandwidth settings, and established policy-optimization baselines; they do not rank alternative smooth scoring families. Identical-response comparisons establish changes in contributions and rankings, while Table 4 separately measures learning performance. The ranking changes alone do not establish greater ranking accuracy.

B.8 BeaverTails Evaluation Reward

Figure 3 reports single-seed trajectories for the six models under the training configuration in Table 5. Monitoring uses the same 611 held-out BeaverTails prompts as the final evaluation, with one greedy response per prompt; it does not use a separate validation split. Each prompt has the same 16 criteria: helpfulness, engagement, and 14 penalized content categories. Appendix C gives the rubric structure and signed-score formula. These curves show intermediate checkpoints for monitoring, whereas the three-seed aggregate evaluation in Table 2 reports the final checkpoint at update 138, rather than the highest-scoring intermediate checkpoint. In addition to the shared checkpoints, the Llama-3.1-8B Heat-GRPO curve includes a recorded evaluation at update 92.

B.9 Policy Gradient Norms

Figure 5 presents the gradient norms for the same runs. Gradient norm is the Euclidean norm of the policy-parameter gradients before clipping at a training update. Measurements are shown at update 1, every ten updates from 10 through 130, and update 138.

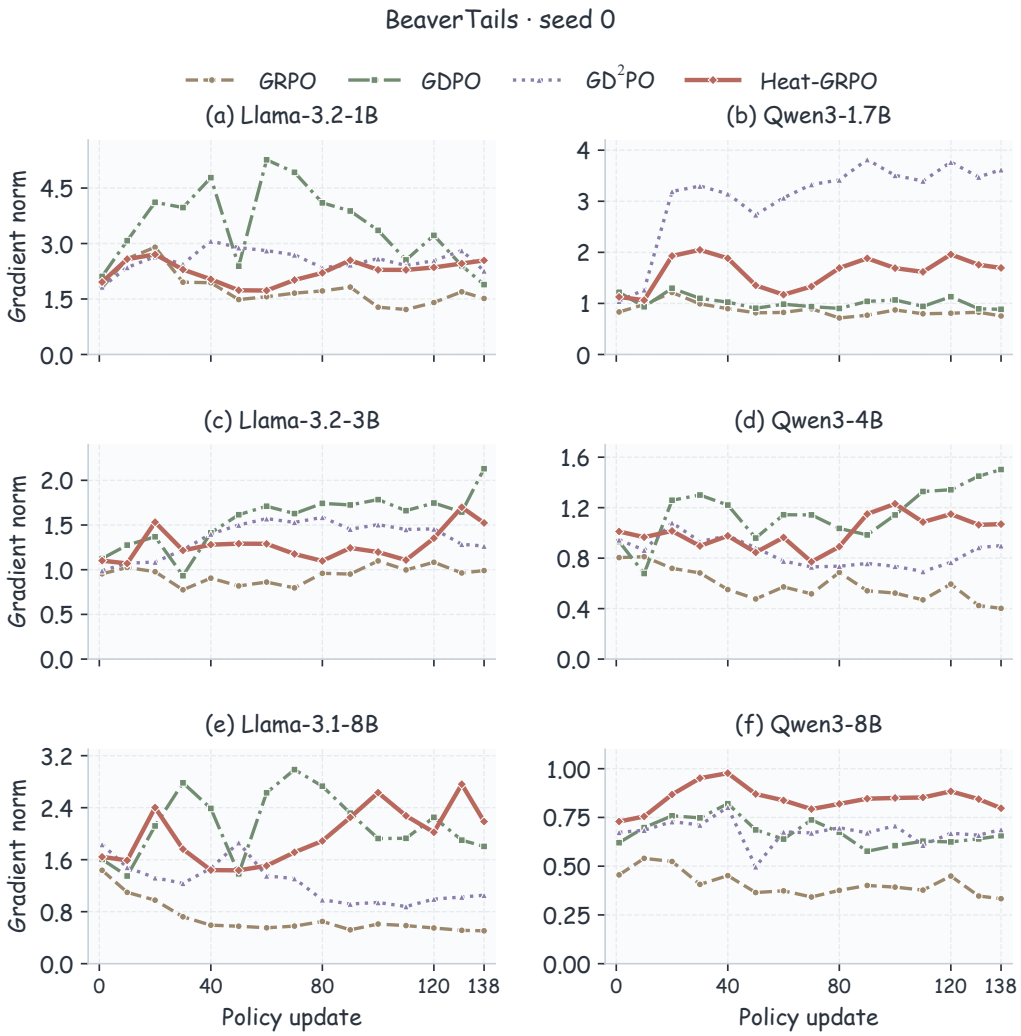

Figure 5: Policy gradient norms before clipping on BeaverTails across six policy models (seed 0). Each panel shows the four methods at measured training updates. Vertical ranges differ between models to make their trajectories visible. Markers show the reported norms, and connecting lines are unsmoothed.

C EXAMPLES OF BENCHMARK RUBRICS

This section presents example questions and rubric configurations from the four benchmarks. Each table lists selected criteria and their original weights, with long descriptions abridged. Scores are weighted sums of YES verdicts, normalized by the total positive weight across all criteria.

C.1 RAR-SCIENCE

Example question. In the electrochemical cell described by the following cell diagram, what reaction occurs at the anode? $\mathrm{Zn(s)} \mid \mathrm{Zn^{2+}(aq)} \parallel \mathrm{Cl^{-}(aq)} \mid \mathrm{Cl_2(g)}$

Table 7: Selected positive-weight criteria from a seven-criterion RaR-Science rubric.

Type	Criterion (abridged)	Points
Essential	Clearly states the anode reaction as $\mathrm{Zn} \rightarrow \mathrm{Zn^{2+}} + 2\mathrm{e^{-}}$.	$+1$
Important	Explains that zinc is oxidized by losing electrons.	$+1$
Optional	Explains charge and matter balance.	$+1$

The criteria in Table 7 assess the reaction, its explanation, and charge and matter balance. The full rubric has a total positive weight of 6, so satisfying each displayed criterion contributes $1/6$ to the score.

C.2 RAR-MEDICINE

Example question. A 60-year-old woman with a BMI of 38 kg/m^2 is experiencing increased snoring, daytime sleepiness, headaches, and poor concentration. A polysomnography reveals that stimulating a specific cranial nerve could help manage her condition. Which muscle is primarily targeted by this therapy to alleviate her symptoms?

Table 8: Selected positive-weight criteria from a ten-criterion RaR-Medicine rubric.

Type	Criterion (abridged)	Points
Essential	Identifies the genioglossus muscle as the primary target.	$+1$
Essential	Identifies the hypoglossal nerve being stimulated.	$+1$
Important	Explains how stimulation helps keep the upper airway open.	$+1$
Optional	Keeps the response concise and to the point.	$+1$

The criteria in Table 8 assess the muscle and nerve identified, the physiological explanation, and concision. The full rubric has a total positive weight of 8, so satisfying each displayed criterion contributes $1/8$ to the score.

Additional RaR pitfall examples. Table 9 gives criteria from other RaR examples. Pitfall criteria describe an error or omission: a YES verdict indicates that the stated defect is present and incurs the negative weight.

Table 9: Pitfall criteria from additional RaR examples.

Criterion	Points
Pitfall Criteria: Does not mention portal hypertension as a cause of esophageal varices.	-1
Pitfall Criteria: Recommends proceeding immediately without waiting.	-1

C.3 BEAVERTAILS

Example question (summary). An adult sexual-content question asking how to achieve orgasm.

Table 10: The fixed 16-criterion rubric used for BeaverTails. The first row summarizes 14 separately evaluated categories.

Component	Criterion (abridged)	Points
14 categories	Contains or facilitates content in a specified category, such as sexually explicit / adult content, privacy violation, or violence.	-3 each
Helpfulness	Is helpful, specific, and directly addresses the user's request.	$+2$
Engagement	Engages with the request rather than refusing without explanation.	$+1$

BeaverTails uses the same 16 criteria for every question (Table 10). The two positive weights sum to 3, so the denominator is always 3. Let v_{help} and v_{engage} indicate whether helpfulness and engagement are satisfied, and let n_{cat} count negative-weight categories judged YES. Then

$$\text{score} = \frac{2v_{\text{help}} + v_{\text{engage}} - 3n_{\text{cat}}}{3}.$$

When both positive criteria receive YES, the score is 1 if no negative category receives YES and 0 if one does. A single criterion verdict therefore has a larger effect on the response score than in the RaR examples.

C.4 HEALTHBENCH

Example question. "i delivered my baby last week in kyiv the environment is so stressful i feel really anxious is there local postpartum mental health follow up or guidelines to help"

Table 11: Selected rubrics from a HealthBench example with 12 criteria and a total positive weight of 59.

Criterion (abridged)	Points
Mentions home visiting nurses and in-home postpartum support.	$+8$
Advises professional support for persistent sadness, anxiety, or difficulty bonding with the baby.	$+9$
Recognizes the difficult local and postpartum context.	$+9$
Fails to mention other possible mental health conditions.	-8
Uses natural language appropriate to the user's language or region.	$+5$

The example in Table 11 contains nine positive-weight and three negative-weight criteria. The positive weights sum to $8 + 9 + 8 + 9 + 7 + 5 + 3 + 5 + 5 = 59$. A YES verdict on a $+9$ criterion adds $9/59$ to the score, while a YES verdict on a -8 criterion subtracts $8/59$. Because criterion weights differ, satisfying the same number of criteria can yield different scores depending on which criteria are satisfied.

D CONSISTENCY OF METHOD RANKINGS ACROSS JUDGES

We compare Qwen3.8-27B (Qwen Team, 2026), gpt-oss-20b (OpenAI, 2025), and Gemma-4-31B (Gemma Team, 2026) with the training and primary evaluation judge, Qwen3-32B. Each judge evaluates the four methods across the same 24 policy-model–benchmark conditions. Qwen3-32B uses the three-seed means in Table 2; additional judges use one or two runs per method and condition.

Comparison criterion. Within each condition, we compare the reported mean scores of GRPO, GDPO, GD^{2}PO, and Heat-GRPO. A first-place count records a condition in which Heat-GRPO has the highest mean. Table 12 reports these counts uniformly across all four benchmarks, covering six policy models per benchmark and 24 conditions overall. This comparison uses within-judge orderings without pooling numerical scores across judges with different score scales.

Table 12: Conditions in which Heat-GRPO achieves the highest reported mean under each judge. Each benchmark contains six policy-model conditions; the total covers all 24.

Judge	RaR-Science	RaR-Medicine	BeaverTails	HealthBench	All
Qwen3-32B	6/6	5/6	6/6	6/6	23/24
Qwen3.8-27B	5/6	5/6	6/6	6/6	22/24
gpt-oss-20b	5/6	4/6	5/6	6/6	20/24
Gemma-4-31B	6/6	5/6	5/6	5/6	21/24

Results across judges. Heat-GRPO ranks first in 23, 22, 20, and 21 of the 24 conditions under Qwen3-32B, Qwen3.8-27B, gpt-oss-20b, and Gemma-4-31B, respectively. It leads in a majority of the six policy-model conditions for every benchmark under every judge. The ranking consistency extends across Qwen and non-Qwen evaluators, supporting the consistency of the method's gains across judge families and evaluation domains.