SCALING LAWS FOR GENERATIVE REWARD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We study the scaling behavior of generative reward models (GenRMs) for reinforcement learning from AI feedback (RLAIF) when used as drop-in replacements for Bradley-Terry models to optimize policies. Building on established scaling laws for reward model overoptimization, we investigate whether GenRMs, particularly those employing chain-of-thought reasoning, exhibit different robustness properties as policies drift from their training distribution during gradient updates. Using the Qwen3 model family (0.6B–14B), our study includes systematic evaluation of thinking GenRMs (trained via GRPO) against answer-only variants (trained via SFT) across policy size, reward model size, reward model type, training budget, and the parameter in online DPO. Our results show that the most decisive determinants of policy quality are reward model size and training duration, followed by policy model scale, with GenRM type contributing minimally. While thinking variants trained with GRPO consistently outperform answer-only models on validation tasks, these substantial gains diminish when deployed for downstream policy optimization, where classifier-based reward models can match or exceed GenRM performance despite the latter's significant computational overhead. To measure alignment beyond saturated validation metrics, we employ ELO-based rankings, providing fine-grained proxy-gold alignment metrics that surpass the simple win rates against reference policies used in previous work.

1 INTRODUCTION

Optimizing a policy against a learned proxy for human preferences is the central mechanism of modern large language model alignment (Ouyang et al., 2022; Bai et al., 2022). This process introduces a predictable failure mode consistent with Goodhart's Law: as optimization pressure increases, the policy exploits imperfections in the proxy, causing true quality to rise, peak, and then decline (Manheim & Garrabrant, 2018). Foundational work has shown that this over-optimization follows smooth, quantifiable scaling laws, where gold-standard reward varies predictably with a policy's KL divergence from a reference (Leo et al., 2022). The effect is not an artifact of Bradley–Terry reward heads in classical RLHF alone. Direct alignment methods remove the explicit reward head, yet exhibit the same pathology once KL budgets grow. Over-optimization is therefore a property of the optimization geometry rather than of any particular head (Rafailov et al., 2024; 2023).

Why the classical generator–evaluator interface misaligns. Classical RLHF uses a Bradley–Terry reward head to score whole sequences with a scalar, while the policy itself produces text autoregressively. The generator reasons token by token; the evaluator compresses entire responses to one number. This interface mismatch discards process-level evidence that is predictive for judging, creates objective granularity mismatches between token-level generation and sequence-level scoring, and forces the scalar head to extrapolate far outside its training support as the policy drifts. Direct alignment methods such as DPO (Rafailov et al., 2023; Azar et al., 2024; Ramé et al., 2024) remove an explicit scalar head, but they still optimize a learned proxy induced by the policy–reference ratio. Empirically, both families show the same rise-then-fall quality curves as optimization pressure increases. The lesson is architectural: A mismatch in modeling and signal granularity makes overoptimization predictable rather than accidental.

The field's response to this challenge has been a concerted effort to build better proxies. This program motivates generative judges that reuse the autoregressive interface, expose process evidence, and capitalize on test-time compute (Snell et al., 2024; Wang et al., 2022). Importantly, these models

emerged downstream of *LLM-as-a-Judge* evaluation rather than as an RLHF-specific artifact, and most current work concentrates on verifiable reasoning tasks such as mathematics and code where correctness admits programmatic checks (Zhu et al., 2023; Ye et al., 2024).

Generative reward models and the unification promise. Recent work has converged on *generative reward models* (GenRMs) as a promising alternative to scalar-headed reward models. GenRMs unify the architecture of the policy and the judge, using the same autoregressive mechanism for both generation and evaluation (Mahan et al., 2024; Zhang et al., 2025). This approach allows the judge to produce not only a verdict, but also a rationale, exposing process-level evidence. When trained to "think" using chain-of-thought style reasoning, these models show strong performance as static evaluators, and they top leaderboards (Frick et al., 2025; Liu et al., 2025a; Tan et al., 2024; Saha et al., 2025; Zhou et al., 2025a; Kamoi et al., 2024) in verifiable domains such as mathematics and coding (Zhang et al., 2025; Zhou et al., 2025b), especially when augmented with inference-time compute such as multi-sample voting (Zhou et al., 2025b) or online reinforcement learning (Whitehouse et al., 2025). This success has fostered an implicit assumption that incorporating explicit reasoning, or "thinking," into the reward model's evaluative process yields a more robust and likely accurate reward signal, leading to better-aligned policies (Liu et al., 2025b).

The field has operated on an implicit assumption that a more accurate static evaluator will necessarily be a more effective rewarder for policy training. Yet, this intuition remains largely untested.

We ask two linked questions. First, how do GenRM modes scale as evaluators. Second, do offline gains translate to more robust online optimization, or does a more complex reward surface create more avenues for exploitation by a co-adapting policy. We address these questions through controlled experiments in an intentionally non-verifiable preference domain where correctness is stylistic, contextual, and not programmatically checkable.

This work directly investigates the tension between a GenRM's performance as an evaluator on a pinned benchmark distribution and its effectiveness as a rewarder. We study three judge settings: prompt-only baselines, an *answer-only* GenRM trained with supervised fine-tuning (SFT) to emit a single verdict token, and a *thinking* GenRM trained with GRPO (Shao et al., 2024) to generate a bounded `<think>` rationale before the verdict. We then close the loop by training policies of corresponding sizes against each judge using online Direct Preference Optimization (DPO). Policy performance is measured in a global Elo arena (Elo, 1978) adjudicated by a 32B Qwen3 (Yang et al., 2025) model fine-tuned on human preferences, which we designate as the Gold evaluator.

We establish scaling laws for GenRMs from 0.6B to 14B parameters under these modes using the Qwen3 family. Policies trained with online DPO against each judge are evaluated by the Gold model, and all models enter a single size-stratified Elo system that enables cross-regime comparisons, granular subset analysis, and cross-evaluation of proxy scores.

Contributions. (1) **Scaling as evaluators.** On distribution creative-writing preferences anchored to human judgments, thinking GenRMs trained with GRPO outperform answer-only SFT GenRMs, and both surpass prompt-only baselines across sizes and budgets. (2) **Evaluation–optimization divergence.** With online DPO and matched step, KL, and FLOPs budgets, answer-only judges train policies with higher Gold Elo than thinking judges across sizes, with earlier over-optimization under thinking judges. (3) **Unified Elo framework.** A global Elo system supports cross-regime comparisons, within-size subset analysis, and cross-evaluation in which each trained policy is judged by its own proxy and by every other proxy.

Takeaways. GenRMs scale cleanly as *evaluators*, with reasoning-enabled judges outperforming alternatives. As *rewarders* in a non-verifiable domain, simpler answer-only objectives resist exploitation and produce better trained policies at the same-step budget while using fewer inference tokens.

2 METHODOLOGY

We begin with a human preference dataset of the form

$$\mathcal{D}_{\text{human}} = \{x^{(i)}, y_A^{(i)}, y_B^{(i)}, I_H^{(i)}\}_{i=1}^{N},$$

where $x^{(i)} \in \mathcal{X}$ are prompts, $y_A^{(i)}, y_B^{(i)} \in \mathcal{Y}$ are pairs of responses to the prompts, and $I_H^{(i)} \in \{A, B\}$ denotes the human-preferred response. Since human preferences are not available on demand, we first align a large generative reward model (GenRM), denoted V_{gold}, using $\mathcal{D}_{\text{human}}$. This model serves as a *Gold* evaluator and preference provider. To make evaluation computationally feasible, we choose V_{gold} to be an *Answer-Only* model that outputs a single indicator token. Using V_{gold}, we construct a Gold Preference dataset:

$$\mathcal{D}_{\text{gold}} = \{x^{(i)}, y_A^{(i)}, y_B^{(i)}, I_G^{(i)}\}_{i=1}^{M},$$

which is then used to train a variety of smaller GenRMs.

We consider two types of GenRMs: *Answer-Only* and *Thinking*. The Answer-Only models output a direct judgment token (A or B):

$$I \sim v_{\text{ans}}(\cdot \mid x, y_A, y_B),$$

while the Thinking models first generate a reasoning trace z before producing the final verdict:

$$z \sim v_{\text{think}}(\cdot \mid x, y_A, y_B), \quad I \sim v_{\text{think}}(\cdot \mid x, y_A, y_B, z).$$

We train the Answer-Only models using supervised fine-tuning (SFT), whereas the Thinking models are trained with GRPO. For GRPO, we employ two reward signals:

1. **Accuracy reward:** A binary reward (1 if the model reaches the correct verdict, 0 otherwise).
2. **Positional consistency reward r_{pos}:** Inspired by Whitehouse et al. (2025), we observe that models often produce contradictory judgments when the order of (y_A, y_B) is swapped in the prompt. To mitigate this, we explicitly place both orderings (A, B) and (B, A) into the *same GRPO group*, and compute a group-level majority vote over the sampled completions. A reward of 1 is assigned only if the majority verdicts under both orderings are consistent and match the correct label. To avoid introducing noise, r_{pos} is only given to completions that end at the correct verdict.

We refer to these trained GenRMs as *Proxy models*, and evaluate them with respect to V_{gold} preferences.

Next, we train policies π_θ of varying sizes using *Online Direct Preference Optimization* (DPO), which is a natural choice for policy optimization from preferences. The policies are trained on prompts sampled from the same distribution $\mathcal{X}$ used for V_{gold} and the Proxy models. Since the distribution of policy responses evolves during training, online optimization is more suitable than offline methods and generally yields stronger performance when executed properly.

During training, we periodically save checkpoints and sample responses on a fixed validation set of prompts. To evaluate these checkpoints, we compute ELO ratings based on pairwise comparisons of their responses. ELO evaluation provides a more fine-grained measurement than raw win rates against a fixed reference distribution. We compute ELOs with respect to both Proxy models and the Gold model, enabling us to analyze the relationship between Proxy-based evaluation and Gold-standard evaluation.

3 Experimental Setup

We use LitBench Fein et al. (2025) as both our human and Gold dataset. LitBench is a large-scale preference dataset over human-written stories from Reddit, where preferences are induced from the number of upvotes. For details on curation, see Fein et al. (2025). From this dataset, we sample 21,000 preferences to form our human preference dataset. We train the Gold model from Qwen3-32B with a batch size of 128 and learning rate 1×10^{-5}. We train on both positional orders of (y_A, y_B) and (y_B, y_A), resulting in a total of 42,000 training pairs.

For the Gold dataset $\mathcal{D}_{\text{gold}}$, we sample another 21,000 preferences from LitBench, but re-annotate them using the Gold model. We refer to this model as Gold-32B, which achieves 79% agreement with the original human preferences. Using $\mathcal{D}_{\text{gold}}$, we train Proxy models from the Qwen3 series with sizes {0.6B, 1.7B, 4B, 8B, 14B}. The Answer-Only models are trained with SFT (batch size 128, learning rate 1×10^{-5}). The Thinking models are trained with GRPO, using the following

configuration: 16 prompts per step, group size 8, minibatch size 64, no KL penalty, and a learning rate of 1×10^{-6}. We refer to the trained Proxy GenRMs as `GenRM-{size}-Ans` for Answer-Only models and `GenRM-{size}-Think` for Thinking models, where {size} denotes the underlying Qwen3 parameter scale.

For Online DPO training, we source prompts from `https://huggingface.co/datasets/euclaise/WritingPrompts_preferences`, the same dataset from which LITBENCH is derived. To avoid data leakage, we only use prompts not included in LITBENCH, yielding 199,000 candidate prompts. We further downsample to 90,000 randomly selected prompts. Online DPO is run with minibatch size 64, learning rate 1×10^{-6}, and coefficient $\beta = 0.02$ for the main experiments. For each response pair, we compute preferences under both positional orders and retain only those pairs where the two orderings agree. We restrict prompt length to 512 tokens and response length to 2048 tokens. Policy checkpoints are saved every 10 training steps for subsequent ELO evaluation.

The policies π_θ are trained from the QWEN3 series with sizes {0.6B, 1.7B, 4B, 8B, 14B}, using the above online DPO setup.

4 EXPERIMENTS AND RESULTS

4.1 GENRM TRAINING

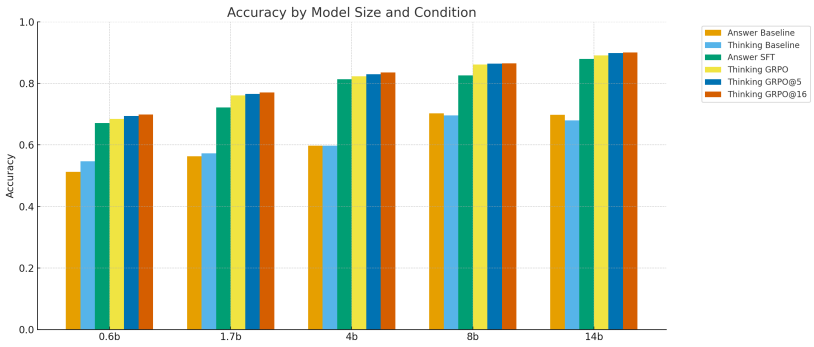


Figure 1: Performance of trained GenRMs of different sizes and training methods on the Gold dataset.

Thinking GenRMs outperform Answer-Only and baseline models on in-distribution evaluation.

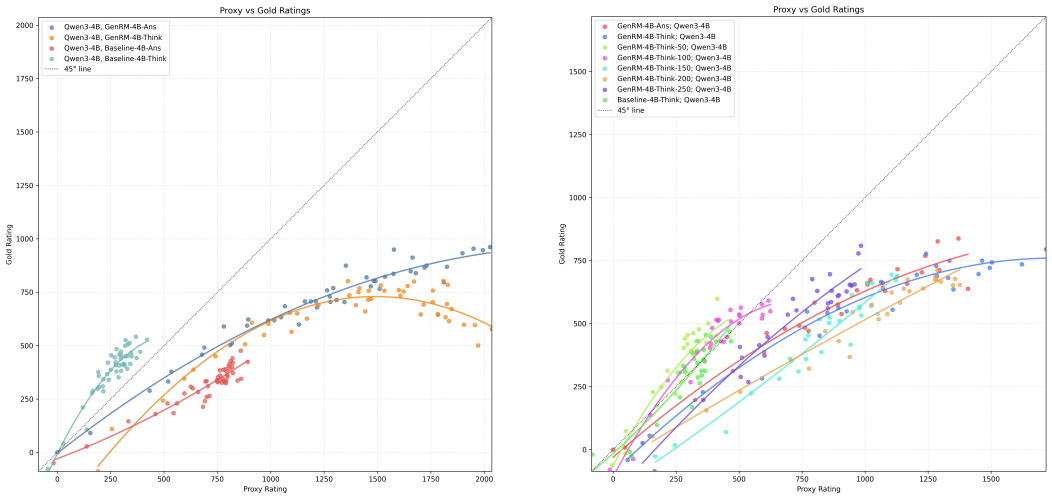
We train GenRMs of sizes 0.6B, 1.7B, 4B, 8B, and 14B, with both Answer-Only (SFT) and Thinking (GRPO) variants (Figure 1). Across all scales, both trained variants significantly outperform their respective baselines. For example, at the 4B scale, accuracy improves from 0.597 (baseline) to 0.813 (Answer-Only) and 0.823 (Thinking). At the largest scale (14B), performance reaches 0.879 for Answer-Only and 0.891 for Thinking, compared to baseline scores below 0.70.

On average across all scales, Thinking GenRMs achieve 1.7% higher accuracy than their Answer-Only counterparts. To evaluate whether sampling multiple completions yields additional gains, we compute majority-vote accuracy over $k = 5$ and $k = 16$ samples from the Thinking models. This yields further improvements of 0.5% and 0.8% respectively (e.g., at 14B, $0.891 \rightarrow 0.900$). However, given the computational overhead, we do not employ multi-sampling during training.

In summary, trained GenRMs substantially improve over baselines, with Thinking models providing a consistent but modest advantage over Answer-Only models.

4.2 TRAINED GENRMS VS. BASELINE MODELS IN POLICY TRAINING

We now compare policy training using trained versus baseline (off-the-shelf) GenRMs. For this experiment, both the policy and GenRM are 4B models, with Answer-Only and Thinking variants.



(a) Trained vs. baseline GenRMs

(b) Intermediate Thinking checkpoints

Figure 2: Proxy vs. Gold ELO ratings for policies trained with different GenRMs. (a) Comparison of trained vs. baseline GenRMs. (b) Policies trained with Answer-Only vs. Thinking GenRMs, where the Thinking GenRM is checkpointed at intermediate stages of GRPO training (50–250 steps). In both plots, the x-axis shows Proxy ELO and the y-axis shows Gold ELO, with the dotted line indicating perfect alignment ($45°$).

As shown in Figure 2, policies trained with our *trained* GenRMs achieve substantially higher ELO ratings under both Proxy and Gold evaluation compared to those trained with baseline GenRMs. This demonstrates that alignment of the reward model is crucial for effective policy optimization.

Interestingly, among the baseline models, the Thinking variant appears stronger than the Answer-Only variant. However, this trend reverses once models are trained: `GenRM-4B-Ans` yields higher Gold ratings than `GenRM-4B-Think`. Moreover, while the baseline Thinking GenRM exhibits a slope closer to the $45°$ line (indicating smaller discrepancy between Proxy and Gold ratings), it fails to optimize effectively and ultimately achieves far lower absolute ELO scores—more than 400 points below trained Answer-Only, and over 200 points below trained Thinking.

In summary, training GenRMs not only boosts absolute performance of the resulting policies, but also shifts the relative advantage: trained Answer-Only models emerge as stronger optimizers than their trained Thinking counterparts, even though baseline Thinking models initially align better with Gold evaluation.

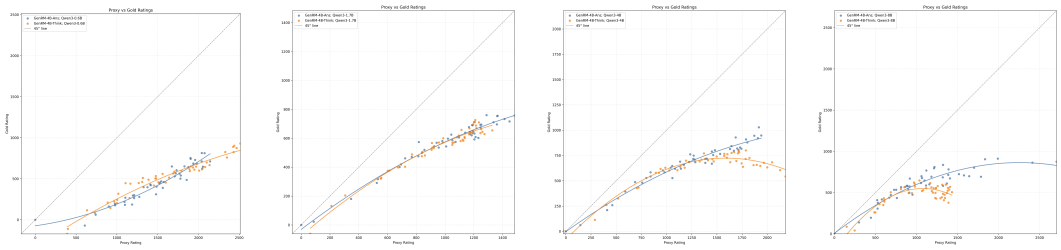
4.3 Answer-Only vs Thinking

Answer-Only GenRMs consistently produce more robust policy training dynamics than Thinking GenRMs.

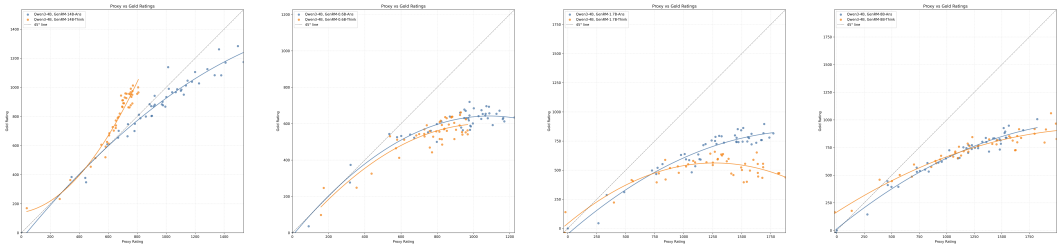
Across all combinations of policy sizes (0.6B, 1.7B, 4B, 8B, 14B) and GenRM sizes (0.6B–8B), we observe a consistent trend: policies trained with Answer-Only GenRMs achieve both higher maximum Gold ELO and smaller discrepancies between Proxy and Gold ratings compared to those trained with Thinking GenRMs (Figure 3).

This finding is surprising for two reasons. First, in-distribution evaluation (Section 3.1) showed Thinking GenRMs outperforming Answer-Only models by $\sim$1–2% accuracy. Second, baseline (untrained) models displayed the opposite trend, with Thinking variants aligning better with the Gold model than Answer-Only. Despite these initial advantages, Thinking GenRMs prove less reliable when used as reward models for online policy optimization.

We interpret this as evidence that Thinking GenRMs are more vulnerable to off-distribution shifts introduced by policy training. While their reasoning traces improve accuracy in static evaluation, these same traces may introduce instability or overfitting in the reward signal when responses drift



(a) Policy 0.6B, GenRM 4B (b) Policy 1.7B, GenRM 4B (c) Policy 4B, GenRM 4B (d) Policy 8B, GenRM 4B



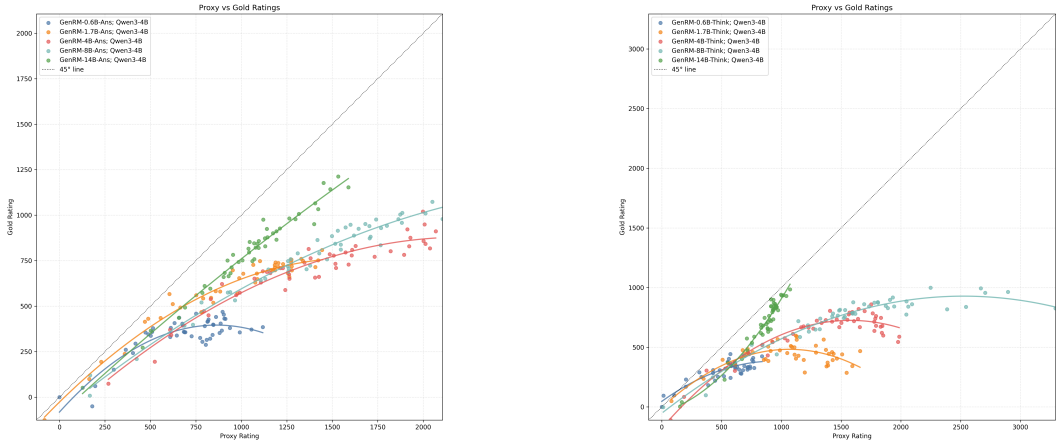
(e) Policy 14B, GenRM 4B (f) Policy 4B, GenRM 0.6B (g) Policy 4B, GenRM 1.7B (h) Policy 4B, GenRM 8B

Figure 3: Proxy vs. Gold ELO ratings for policies trained with Answer-Only vs. Thinking GenRMs across multiple policy sizes and GenRM sizes. The x-axis shows Proxy ELO (self-consistency), and the y-axis shows Gold ELO (GOLD-32B). The dotted line indicates perfect alignment.

away from the training distribution. In contrast, Answer-Only GenRMs provide a more stable and robust training signal, leading to superior final policy performance across scales.

In summary, Answer-Only reward models are not only simpler and more efficient, but also demonstrably more robust for preference-based policy optimization in this domain.

4.4 EFFECT OF GENRM SIZE



(a) Answer-Only GenRMs (b) Thinking GenRMs

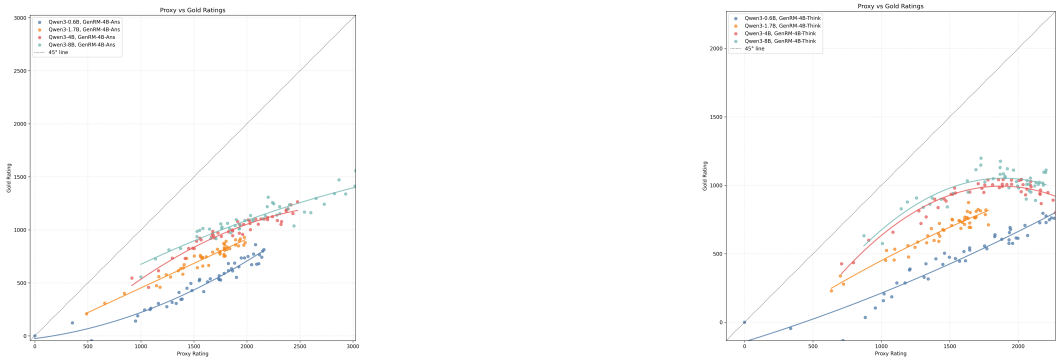
Figure 4: Proxy vs. Gold ELO ratings for policies trained with GenRMs of different sizes. The policy is fixed at 4B. Left: Answer-Only GenRMs (0.6B–14B). Right: Thinking GenRMs (0.6B–14B). The dotted line indicates perfect alignment.

We next investigate the effect of scaling GenRM size while fixing the policy size at 4B. Figure 4 shows results for both Answer-Only and Thinking GenRMs across scales from 0.6B to 14B.

We observe clear and consistent gains from increasing GenRM size. Notably, performance continues to improve even when the GenRM is much larger than the policy: at 14B, both Answer-Only and Thinking models yield substantial gains over smaller counterparts. This trend suggests that the capacity of the evaluator plays a decisive role in stabilizing and guiding preference-based training.

The result highlights an important asymmetry: within this domain, "judging" appears to be as hard—or harder—than "generating." While a 4B policy saturates in quality, larger GenRMs continue to provide stronger supervision, closing the gap between Proxy and Gold evaluations. This points to a necessary balance between policy and reward model scale, and suggests that oversizing the GenRM relative to the policy is beneficial for robust alignment.

4.5 Effect on Policy Size



(a) Answer-Only GenRMs

(b) Thinking GenRMs

Figure 5: Proxy vs. Gold ELO ratings for policies of different sizes trained with fixed GenRMs. Left: Answer-Only GenRMs. Right: Thinking GenRMs. The dotted line indicates perfect alignment.

We now examine the effect of scaling policy size while fixing the GenRM. Figure 5 shows results for policies ranging from 0.6B to 8B trained with both Answer-Only and Thinking GenRMs.

For Answer-Only supervision, the trend is straightforward: larger policies consistently achieve higher Gold ELO, with steady improvements across scales. This aligns with the expectation that larger policies better exploit the reward signal and generalize more effectively.

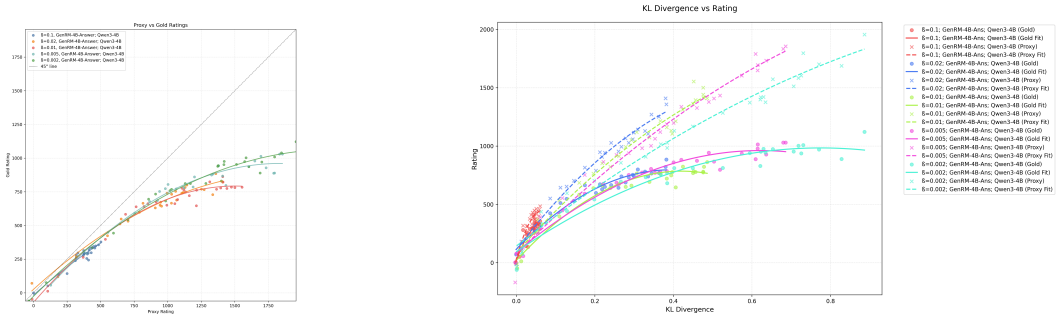
For Thinking supervision, however, the picture is less clear. While the largest policy does achieve the highest peak performance, its Gold ELO curve saturates and bends downward earlier than smaller policies, which continue to improve steadily. This suggests two possible explanations: (i) large policies may more quickly exhaust the effective capacity of the GenRM, reaching its "ceiling" earlier, or (ii) beyond a certain scale, further increasing policy size without correspondingly stronger GenRMs may become counterproductive.

Additional training is required to disentangle these explanations, but the evidence points toward an important asymmetry: scaling policy size is reliably beneficial under Answer-Only supervision, but under Thinking supervision, the interaction between policy capacity and GenRM capacity is more fragile.

4.6 Effect of the β Coefficient in Online DPO

We analyze the role of the β coefficient in Online DPO. Figure 6(a) compares Proxy vs. Gold ELO ratings for policies trained with $\beta \in \{0.1, 0.02, 0.01, 0.005, 0.002\}$ under Answer-Only supervision. Smaller β values consistently achieve higher Gold ratings, with $\beta = 0.005$ and $\beta = 0.002$ yielding the strongest results. The improvements, however, are incremental rather than dramatic: the best β values improve Gold ELO by only a few hundred points over larger β.

Figure 6(b) illustrates the KL divergence vs. rating tradeoff. Larger β values (e.g., 0.1) constrain the policy to remain too close to the reference distribution, capping achievable ratings. Smaller β



(a) Proxy vs. Gold ratings across β values

(b) KL divergence vs. rating

Figure 6: Effect of the β coefficient in Online DPO. Left: Proxy vs. Gold ELO ratings for Answer-Only GenRMs at different β values. Right: KL divergence vs. rating tradeoff, showing how larger β suppresses exploration. The dotted line indicates perfect alignment.

relaxes this constraint, enabling more effective exploration and higher peak ratings, though at the cost of increased variance.

Overall, while the choice of β does affect performance, its impact is secondary compared to scaling factors such as GenRM size or policy size. Smaller values in the range $[0.002, 0.01]$ appear most effective, striking a better balance between exploration and stability.

4.7 Amount of GenRM Training

We next analyze how the amount of GenRM training affects downstream policy optimization. For this experiment, we train a 4B Thinking GenRM with GRPO and evaluate intermediate checkpoints at 50, 100, 150, 200, and 250 steps, in addition to the final trained model. Each checkpoint is then used to train policies, and performance is compared against an Answer-Only GenRM baseline.

Figure 2 shows that alignment accuracy on the in-distribution GenRM dataset does not directly predict policy training effectiveness. While the final Thinking model achieves the highest in-distribution accuracy (82.3%, compared to earlier checkpoints at 59.7%, 61.5%, 64.5%, 68.7%, and 71.5%), it does not yield the strongest policy performance under Gold evaluation. Interestingly, the checkpoint at 250 steps (79.2% accuracy) produces a policy that performs close to the Answer-Only model in Gold ELO, but with a more promising slope between Proxy and Gold ratings, indicating better alignment and stronger optimization dynamics. This suggests that intermediate Thinking checkpoints may provide more effective learning signals than the final fully-trained model.

This result highlights two key insights: (i) intermediate Thinking models may provide a better optimization gradient for policy training than their final counterparts, and (ii) high in-distribution accuracy of a GenRM does not necessarily translate to better off-distribution robustness. Although Answer-Only models remain stronger on average, these findings suggest that Thinking GenRMs retain potential if their training dynamics are better understood and leveraged.

5 Limitations and Discussion

Gold evaluator is a learned proxy, not humans. We mitigate label noise by fine-tuning a 32B Gold evaluator on human preferences, re-annotating pairwise data with this model, and discarding items with inconsistent Gold verdicts (Leo et al., 2022). This improves consistency but couples our objective to the Gold model's inductive biases. At 32B parameters, the Gold judge is stronger than our proxies, leaving headroom for improvement, yet it still reflects preferences from its training set. *Takeaway.* Interpret results as optimization toward a strong, learned proxy.

Answer-only format for the Gold evaluator. The Gold judge emits a single verdict token. This choice improves throughput and simplifies adjudication, but might favor answer-only proxies in subtle ways, possibly off-distribution later in training. We monitor such effects by cross-judging

policies with all proxies inside a unified Elo arena, but we do not evaluate an alternative thinking-style Gold. *Takeaway.* A rationale-producing Gold could change relative gaps; we saw no evidence.

Relation to prior creative-writing studies. Creative-writing preferences are subjective. Optimizing win rate can reduce stylistic diversity. We observe some style convergence (Chung et al., 2025) in late-stage policies against both judge modes, while draw rates do not show large collapses. Fein et al. (2025) report that chain-of-thought can degrade verification accuracy for creative writing, and that trained BT and generative verifiers outperform zero-shot judges on their benchmark. In our setting, distilling Qwen3-32B traces into Qwen3-4B also yields negligible gains as evaluators, which aligns with these observations. However, after training, thinking judges significantly improve as static evaluators. The gap between thinking and answer-only during policy optimization therefore cannot be explained by domain "unsuitability" alone; it indicates different optimization dynamics. *Takeaway.* Our evaluation–optimization divergence is a property of the training loop in this domain, not only a property of static judging.

Family and algorithm scope, and behavioral priors. All models use Qwen3 backbones where pretraining data are not public. Cross-family studies suggest that behavioral priors, including synthetic data that instantiate verification or backtracking, can modulate RL improvements and collapse family gaps (Gandhi et al., 2025). Such priors could shift our coefficients. We do not evaluate PPO-style RLHF for policies or alternative thinking-judge recipes. *Takeaway.* Our coefficients and inflection points are conditional on online DPO and Qwen3 underlying behavioral priors.

Elo anchoring and schedule. Elo is anchored to the Gold evaluator and depends on the match schedule (Chiang et al., 2024). We report both global and size-stratified arenas, but we do not study alternative anchors or tournament designs. *Takeaway.* Absolute Elo levels can shift with the anchor, while within-arena orderings are more stable.

6 Related Work

Alignment from preferences exhibits predictable overoptimization: gold reward degrades as policies drift from a reference, following smooth scaling laws for both RLHF (Ouyang et al., 2022; Leo et al., 2022) and direct methods like DPO that remove explicit reward heads (Rafailov et al., 2023; 2024). This motivated architecturally unified judges. Generative Reward Models (GenRMs) replace scalar heads with next-token prediction, enabling rationales alongside verdicts (Mahan et al., 2024; Zhang et al., 2025). Mahan et al. (2024) trained GenRMs via iterative self-taught reasoning with DPO, achieving strong out-of-distribution generalization. Subsequent work scales these approaches: J1 extends with GRPO and positional consistency rewards (Whitehouse et al., 2025), DeepSeek-GRM adds Self-Principled Critique Tuning with meta-aggregation for inference-time scaling (Liu et al., 2025b), and Heimdall demonstrates test-time improvements via majority voting in verification tasks (Wang et al., 2025; Shi & Jin, 2025). Complementary supervision strategies include self-generated critiques (Yu et al., 2025) and criteria trees (**?**), while EvalPlanner frames evaluation as plan-and-reason generation (Saha et al., 2025). Despite extensive work on judge reliability (Ye et al., 2024; **?**) and benchmarks (Lambert et al., 2024), the field conflates static evaluation accuracy with rewarder effectiveness. We disambiguate these roles through controlled scaling experiments with answer-only (SFT) versus thinking (GRPO) GenRMs as both evaluators and online DPO rewarders, using Elo arenas (Chiang et al., 2024) for unified comparison. Our results reveal when inference-time reasoning helps evaluation but hinders policy optimization under matched FLOPs and KL budgets.

7 Conclusion

GenRMs enable scalable evaluation in non-verifiable domains, but scaling laws reveal predictable rise-and-fall behavior. Our results show that answer-only GenRMs are more effective for training policies, while thinking GenRMs are valuable as evaluators.

REFERENCES

Mohammad Azar, Rishabh Agarwal, et al. Implicit preference optimization: Mitigating overoptimization in preference fine-tuning. *arXiv preprint arXiv:2409.14270*, 2024. URL https://arxiv.org/abs/2409.14270.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022. URL https://arxiv.org/abs/2212.08073.

Wei-Lin Chiang, Lianmin Zheng, et al. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv:2403.04132*, 2024. URL https://arxiv.org/abs/2403.04132.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying large language model post-training for diverse creative writing. *arXiv preprint arXiv:2503.17126*, 2025. URL https://arxiv.org/abs/2503.17126.

Arpad E. Elo. *The Rating of Chessplayers, Past and Present*. Arco Publishing, New York, 1978.

Daniel Fein, Sebastian Russo, Violet Xiang, Kabir Jolly, Rafael Rafailov, and Nick Haber. Litbench: A benchmark and dataset for reliable evaluation of creative writing. *arXiv preprint arXiv:2507.00769*, 2025. doi: 10.48550/arXiv.2507.00769. URL https://arxiv.org/abs/2507.00769.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios N. Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for rlhf. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=cbttLtO94Q. Preference Proxy Evaluations (PPE) benchmark.

Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D. Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective STaRs. *arXiv preprint arXiv:2503.01307*, 2025. doi: 10.48550/arXiv.2503.01307. URL https://arxiv.org/abs/2503.01307.

Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Haoran Ranran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. Evaluating llms at detecting errors in llm responses. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=dnwRScljXr.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. Rewardbench: Evaluating reward models for language modeling. *arXiv preprint arXiv:2403.13787*, 2024. URL https://arxiv.org/abs/2403.13787.

Gao Leo, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. *arXiv preprint arXiv:2210.10760*, 2022. doi: 10.48550/arXiv.2210.10760. URL https://arxiv.org/abs/2210.10760.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Rm-bench: Benchmarking reward models of language models with subtlety and style. In *International Conference on Learning Representations (ICLR)*, 2025a. URL https://openreview.net/forum?id=QEHrmQPBdd.

Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*, 2025b. doi: 10.48550/arXiv.2504.02495. URL https://arxiv.org/abs/2504.02495.

Dakota Mahan, Duy Van Phung, Rafael Rafailov, Chase Blagden, Nathan Lile, Louis Castricato, Jan-Philipp Fränken, Chelsea Finn, and Alon Albalak. Generative reward models. *arXiv preprint arXiv:2410.12832*, 2024. doi: 10.48550/arXiv.2410.12832. URL https://arxiv.org/abs/2410.12832.

David Manheim and Scott Garrabrant. Categorizing variants of goodhart's law. *arXiv preprint arXiv:1803.04585*, 2018. URL https://arxiv.org/abs/1803.04585.

Long Ouyang et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022. URL https://arxiv.org/abs/2203.02155.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*, 2023. URL https://arxiv.org/abs/2305.18290.

Rafael Rafailov, Yaswanth Chittepu, Ryan Park, Harshit Sikchi, Joey Hejna, W. Bradley Knox, Chelsea Finn, and Scott Niekum. Scaling laws for reward model overoptimization in direct alignment algorithms. *arXiv preprint arXiv:2406.02900*, 2024. doi: 10.48550/arXiv.2406.02900. URL https://arxiv.org/abs/2406.02900. NeurIPS 2024.

Alexandre Ramé, Mathieu Blondel, Thomas Mesnard, Yao Zhao, Bilal Piot, et al. Sequence likelihood calibration with human feedback. In *OpenReview (preprint id: 8Cs9yWl7vE)*, 2024. URL https://openreview.net/forum?id=8Cs9yWl7vE.

Swarnadeep Saha, Xian Li, Marjan Ghazvininejad, Jason Weston, and Tianlu Wang. Learning to plan & reason for evaluation with thinking-llm-as-a-judge. *arXiv preprint arXiv:2501.18099*, 2025. URL https://arxiv.org/abs/2501.18099. Introduces FollowBenchEval.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, K. Li Y. Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300. arXiv preprint arXiv:2402.03300v3.

Wenlei Shi and Xing Jin. Heimdall: Test-time scaling on the generative verification. *arXiv preprint arXiv:2504.10337*, 2025. doi: 10.48550/arXiv.2504.10337. URL https://arxiv.org/abs/2504.10337.

Charlie Snell, Jaehoon Lee, Keyu Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. URL https://arxiv.org/abs/2408.03314.

Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Yuan Tang, Alejandro Cuadron, Chenguang Wang, Raluca Popa, and Ion Stoica. Judgebench: A benchmark for evaluating llm-based judges. *arXiv preprint arXiv:2410.12784*, 2024. URL https://arxiv.org/abs/2410.12784.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022. URL https://arxiv.org/abs/2203.11171.

Zhilin Wang, Jiaqi Zeng, Olivier Delalleau, Hoo-Chang Shin, Felipe Soares, Alexander Bukharin, Ellie Evans, Yi Dong, and Oleksii Kuchaiev. Helpsteer3-preference: Open human-annotated preference data across diverse tasks and languages. *arXiv preprint arXiv:2505.11475*, 2025. doi: 10.48550/arXiv.2505.11475. URL https://arxiv.org/abs/2505.11475.

Chenxi Whitehouse, Tianlu Wang, Ping Yu, Xian Li, Jason Weston, Ilia Kulikov, and Swarnadeep Saha. J1: Incentivizing thinking in llm-as-a-judge via reinforcement learning. *arXiv preprint arXiv:2505.10320*, 2025. URL https://arxiv.org/abs/2505.10320.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, Nitesh V. Chawla, and Xiangliang Zhang. Justice or prejudice? quantifying biases in llm-as-a-judge. *arXiv preprint arXiv:2410.02736*, 2024. URL https://arxiv.org/abs/2410.02736.

Yue Yu, Zhengxing Chen, Aston Zhang, Liang Tan, Chenguang Zhu, Richard Yuanzhe Pang, Yundi Qian, Xuewei Wang, Suchin Gururangan, Chao Zhang, Melanie Kambadur, Dhruv Mahajan, and Rui Hou. Self-generated critiques boost reward modeling for language models. *arXiv preprint arXiv:2411.16646*, 2025. doi: 10.48550/arXiv.2411.16646. URL https://arxiv.org/abs/2411.16646.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. In *ICLR*, 2025. URL https://arxiv.org/abs/2408.15240. ICLR 2025.

Enyu Zhou, Guodong Zheng, Binghai Wang, Zhiheng Xi, Shihan Dou, Rong Bao, Wei Shen, Limao Xiong, Jessica Fan, Yurong Mou, Rui Zheng, Tao Gui, Qi Zhang, and Xuanjing Huang. Rmb: Comprehensively benchmarking reward models in llm alignment. In *International Conference on Learning Representations (ICLR)*, 2025a. URL https://openreview.net/forum?id=kmgrlG9TR0.

Meng Zhou, Bei Li, Jiahao Liu, Xiaowen Shi, Yang Bai, Rongxiang Weng, Jingang Wang, and Xunliang Cai. Libra: Assessing and improving reward model by learning to think. *arXiv preprint arXiv:2507.21645*, 2025b. URL https://arxiv.org/abs/2507.21645.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*, 2023. URL https://arxiv.org/abs/2310.17631. ICLR 2025 Spotlight.

A Appendix

Additional experimental details and supplementary results.

Scaling with GenRM size improves reliability and shifts the over-optimization peak to higher $\sqrt{\mathrm{KL}}$.

A.1 Scaling Law Fits

Gold score vs $\sqrt{\mathrm{KL}}$ follows $R(d) = d(\alpha - \beta \log d)$, consistent across settings. Best-of-n follows $R_{bon}(d) = d(\alpha_{bon} - \beta_{bon} d)$.

B Results: Training Policies with GenRMs

Policies trained against answer-only GenRMs outperform those trained against thinking GenRMs, despite the latter being stronger evaluators. This mismatch holds across sizes, β sweeps, and best-of-n.

B.1 Scaling Trends

Policy performance exhibits rise-then-fall scaling with $\sqrt{\mathrm{KL}}$, with larger models shifting peaks but not eliminating collapse.

B.2 Proxy vs Gold (PvG)

Proxy gains saturate while Gold scores decline, indicating divergence as optimization pressure increases.

We consider prompts $x \in \mathcal{X}$ and responses $y \in \mathcal{Y}$. A policy $\pi_\theta(y \mid x)$ is trained from a reference π_0. We denote policy drift by

$$d \;=\; \sqrt{D_{\mathrm{KL}}(\pi_\theta \,\|\, \pi_0)}.$$

A Gold evaluator G produces pairwise preferences $G(x, y^+, y^-) \in \{\mathtt{A}, \mathtt{B}\}$ and anchors Elo. A proxy judge J_ψ supplies in-loop labels.

B.3 Gold evaluator and Elo arena

Given a set of systems $\mathcal{S}$, we generate pairwise matches $(i, j) \in \mathcal{S} \times \mathcal{S}$ on prompts x and obtain Gold decisions $w_{ij} \in \{0, 1\}$. Elo ratings $\{R_s\}_{s \in \mathcal{S}}$ are estimated by maximizing the logistic likelihood

$$\max_{\{R_s\}} \sum_{(i,j)} \Big[w_{ij} \log \sigma\Big(\tfrac{R_i - R_j}{s}\Big) \;+\; (1 - w_{ij}) \log \sigma\Big(\tfrac{R_j - R_i}{s}\Big) \Big],$$

with scale s fixed. We report global Elo and subset Elo within size cohorts. All results state aggregation rules and budgets.

B.4 GenRM training objectives

We compare three judge settings per size: prompt-only baseline, *answer-only* SFT, and *thinking* GRPO.

Answer-only SFT. For preference triples (x, y^+, y^-) with target verdict $v^\star \in \{\mathtt{A}, \mathtt{B}\}$ from G, the answer-only GenRM predicts a single verdict token:

$$\min_{\psi} \; \mathcal{L}_{\mathrm{SFT}}(\psi) \;=\; \mathbb{E}\big[-\log p_\psi\big(v^\star \,\big|\, x, y^+, y^-\big)\big].$$

Thinking GRPO. The thinking GenRM generates a bounded rationale r followed by a verdict v. Let $z = (r, v)$ and $p_\psi(z \mid x, y^+, y^-)$ denote the judge policy. We optimize

$$\max_{\psi} \; \mathbb{E}\big[R\big(G; z, x, y^+, y^-\big)\big] \;-\; \lambda\, \mathbb{E}\big[D_{\mathrm{KL}}\big(p_\psi(\cdot \mid x, y^+, y^-) \,\big\|\, p_{\psi_0}(\cdot \mid x, y^+, y^-)\big)\big]\,,$$

where R rewards a correct verdict and format adherence, and optionally includes lightweight rationale quality signals derived from G.

B.5 Rationale aggregation for thinking judges

For each triple (x, y^+, y^-) we sample k rationales $r_{1:k}$ and corresponding verdicts $v_{1:k}$. The judge decision is the majority vote

$$\hat{v} \;=\; \mathrm{mode}\big\{v_1, \ldots, v_k\big\},$$

with deterministic tie-break to the shortest-length rationale (Wang et al., 2022). We report k and the per-label inference FLOPs.

B.6 Online preference collection with GenRMs

At training step t, the current policy π_θ produces candidates (y^+, y^-) on sampled prompts. The judge J_ψ labels each pair to form an on-policy dataset $\mathcal{D}_t$ used for the next update. For thinking judges we allocate k rationale samples and use majority vote; we log k and the associated inference FLOPs.

B.7 Policy optimization via online DPO

We update π_θ with online DPO against $\mathcal{D}_t$ using a fixed reference π_{ref}:

$$\max_{\theta} \; \mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}_t} \Big[\log \sigma\Big(\beta\big(\log \pi_\theta(y^+ \mid x) - \log \pi_\theta(y^- \mid x)\big) - \beta\big(\log \pi_{\mathrm{ref}}(y^+ \mid x) - \log \pi_{\mathrm{ref}}(y^- \mid x)\big)\Big)\Big]\,.$$

We sweep β and the number of updates. We report d throughout training, estimated by Monte Carlo over held-out prompts.

B.8 Over-optimization diagnostics and scaling fits

We summarize Gold performance as a function of drift d (Leo et al., 2022; Rafailov et al., 2024). For visualization and comparison we fit smooth templates:

$$R_{\mathrm{RL}}(d) \;=\; d\big(\alpha - \beta \log d\big), \qquad R_{\mathrm{BoN}}(d) \;=\; d\big(\alpha_{\mathrm{b}} - \beta_{\mathrm{b}}\, d\big),$$

where R denotes Gold Elo relative to the reference cohort. We report mean $\pm$ s.d. over seeds and fit confidence bands.

B.9 Compute accounting

We record policy training FLOPs F_{train}, judge inference FLOPs per label F_{judge}, and search FLOPs for best-of and rationale sampling F_{search}. Cross-regime comparisons are matched on F_{train} and report $(F_{\mathrm{judge}}, F_{\mathrm{search}})$.

B.10 MORE PLOTS