FLEXIBLE QUANTILE REWARD MODELING FOR RLHF WITH GRADED-SCORE SUPERVISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reward models are central to RLHF because they determine which model behaviors are reinforced during policy optimization. However, preference datasets with graded response scores are often reduced to binary chosen/rejected labels, discarding information about preference strength and reward location, while existing distributional approaches may impose restrictive parametric assumptions. We introduce the Flexible Quantile Reward Model (FQRM), which represents each response with a scalar-compatible center and an ordered quantile support capable of modeling skewed, asymmetric, and heavy-tailed reward distributions. Given comparable response-level scores, FQRM derives complementary supervision for relative ranking, preference strength, support location, and coarse reward regions, allowing the available graded information to constrain a single response-level representation. Unlike attribute-based quantile reward models, FQRM requires neither a predefined reward-attribute ontology nor attribute-level annotations. Experiments across multiple reward model benchmarks demonstrate the effectiveness of the proposed framework.

1 INTRODUCTION

Large language models (LLMs) acquire broad capabilities through large-scale pretraining, but next-token prediction alone does not ensure that their behavior follows human intentions. Reinforcement learning from human feedback (RLHF) addresses this gap by collecting preference feedback, training a reward model (RM), and using the learned reward to optimize the language-model policy (Ziegler et al., 2019; Stiennon et al., 2020; Christiano et al., 2017; Ouyang et al., 2022). Given a prompt x and a response y, an RM assigns a numerical reward intended to serve as a proxy for how well the response satisfies the desired behavior. During policy optimization, this reward determines which responses are reinforced, and RMs are also commonly used to rank candidate outputs. Consequently, the quality of the learned reward directly shapes the effectiveness and reliability of downstream alignment.

Most preference-based RMs are point-valued, assigning each prompt–response pair a scalar score learned from chosen–rejected comparisons with a Bradley–Terry objective (Bradley & Terry, 1952; Christiano et al., 2017; Ouyang et al., 2022). Graded-score regression and strength-aware objectives exploit richer feedback, but still produce point-valued rewards (Wang et al., 2024b;a). PURM instead predicts a Gaussian reward distribution for each response (Sun et al., 2025), but this symmetric, unimodal family can be misspecified for skewed or multimodal rewards. This motivates a flexible distributional representation without a fixed parametric family.

Distributional reinforcement learning provides a useful precedent for such a representation. Rather than estimating only the expected return, distributional RL models the return distribution itself. The seminal categorical formulation paired theoretical analysis with state-of-the-art Atari results (Bellemare et al., 2017), while quantile-based successors represented return distributions without a Gaussian assumption and reported further empirical improvements (Dabney et al., 2018b;a). Although return distributions and response-level rewards are different objects, this precedent motivates us to adapt the same representational idea to construct a more flexible reward model.

Motivated by this connection, we introduce the Flexible Quantile Reward Model (FQRM), which represents each prompt–response pair with a strictly ordered empirical reward support whose mean

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.

serves as a scalar center. Variable spacing among the support values allows FQRM to adapt its distributional shape without imposing a Gaussian family. Given scored response pairs, source-normalized scores anchor the support locations through pinball losses, while score differences define soft preference targets for both center comparisons and cross-support win probabilities; coarse score regions provide additional supervision. At inference time, only the scalar center is used for ranking, so FQRM retains the standard RM interface and requires neither scores nor attribute annotations.

Our contributions are summarized as follows.

- We transfer the central principle of distributional reinforcement learning from return modeling to reward modeling by representing each response with a learned ordered reward support.
- We develop a unified objective that uses scores normalized within each source to anchor support locations and score gaps to supervise both comparisons between scalar centers and comparisons across support points. The method requires no predefined reward ontology or attribute annotations.
- In a controlled setting with graded score supervision and additional ambiguous preference pairs, we show that FQRM is competitive with and generally stronger than scalar and Gaussian probabilistic baselines across three reward model benchmarks.

2 Related Work

Scalar and score-aware reward modeling. Most RLHF pipelines learn a scalar reward from pairwise preferences (Christiano et al., 2017; Ouyang et al., 2022), commonly using the Bradley–Terry formulation (Bradley & Terry, 1952). Beyond preferred and dispreferred response labels, regression-based RMs use graded response scores as direct supervision (Wang et al., 2024b), and combined rating–preference datasets study how the two supervision types complement one another (Wang et al., 2025), while strength-aware objectives use preference magnitudes to construct soft targets or adaptive margins (Wang et al., 2024a). Studies of annotator disagreement further examine how heterogeneous feedback affects the learned preference signal (Zhang et al., 2024). Reward-model overoptimization also shows that better fit to a learned proxy need not imply better downstream behavior (Gao et al., 2022); robust training has therefore been studied as a way to reduce reward hacking (Liu et al., 2024a). These methods exploit richer supervision than a hard preference label, but point-valued formulations do not explicitly preserve it in the learned reward.

Distributional reinforcement learning. Distributional reinforcement learning provides a mature precedent for preserving information beyond an expected value. C51 represents the return distribution as a categorical distribution over fixed support locations (Bellemare et al., 2017). Rather than learning probabilities over fixed locations, QR-DQN uses equally weighted support points whose locations are learned through quantile regression (Dabney et al., 2018b). IQN further replaces a fixed quantile grid with a learned implicit quantile function (Dabney et al., 2018a). Together, these methods show how flexible distributions can be learned without imposing a Gaussian family, providing a natural methodological reference for applying the same idea to reward modeling.

Distributional reward modeling. Recently, several studies have begun to model rewards as distributions rather than point values. PURM assigns each prompt and response pair a Gaussian reward distribution and extends Bradley–Terry comparisons to pairs of such distributions (Sun et al., 2025). QRM uses quantile regression to estimate distributions over predefined reward attributes and combines them through a learned gate (Dorka, 2024). PURM is restricted to a Gaussian family, while QRM relies on an attribute ontology and attribute-level annotations. Other uncertainty-aware approaches use ensembles or diverse reward heads to estimate disagreement and mitigate overoptimization (Coste et al., 2024; Eisenstein et al., 2024; Zhai et al., 2024). FQRM instead learns one ordered support for each response directly from graded scores, score differences, and pairwise order, without imposing a Gaussian family or requiring attribute-level annotations.

3 Method

FQRM applies the distributional-RL viewpoint to response-level reward modeling. For each prompt–response pair, it retains a scalar center for compatibility with conventional reward-model

interfaces while learning an ordered empirical reward support that can preserve response-level variation. The support is trained jointly with pairwise preferences, soft preference probabilities, and graded response scores. We first review the concepts that motivate this construction, then define the ordered representation and its mixed supervision objectives; implementation and data-processing details are deferred to the experimental setup and appendix.

3.1 Problem Setup and Supervision

Let x denote a prompt and let y^+ and y^- be two responses to x, where y^+ is preferred to y^-. When graded feedback is available, the training record additionally contains raw response scores g^+ and g^- together with their data source m. These annotations are used only to construct training targets. The model itself takes only a prompt–response pair (x, y) as input.

Because different sources may use different scoring scales, a source-specific transformation $\mathcal{N}_m$ maps the raw scores to $r^\pm = \mathcal{N}_m(g^\pm)$. The exact transformations and clipping ranges are given in Appendix B.1. The individual values $r^\pm$ provide response-level reward-location targets, while their difference defines the soft pairwise target

$$p = \mathrm{clip}\left(\sigma\left(\frac{r^+ - r^-}{T_p}\right)\right),$$

where σ is the logistic sigmoid and $T_p > 0$ controls sensitivity to the score gap. Thus, p represents the score-derived strength and direction of the comparison. For a pair with only a hard preference label, we place the preferred response in y^+ and set $p = 1$.

Each normalized score is also assigned a coarse region label $z^\pm \in \{0, 1, 2\}$ using fixed thresholds. The normalized scores $r^\pm$ constrain the locations of the two reward supports, the soft target p supervises their relative comparison, and the region labels $z^\pm$ provide a coarser location constraint. We next define the response-level representation on which these targets operate.

The three signals are complementary: scores locate each response on the shared reward scale, their difference determines comparison strength, and region labels provide a coarse low/middle/high constraint. They supervise training only; the model takes (x, y) and requires no annotations at inference.

3.2 Response-Level Ordered Reward Representation

FQRM represents each prompt–response pair with ordered support values $q_1 < \cdots < q_N$ whose arithmetic mean is the scalar center used for ranking. Let $h_\phi(x, y) \in \mathbb{R}^d$ denote the representation produced by the language-model backbone. The reward head produces

$$\mathbf{o}(x, y) = W h_\phi(x, y) + b = \big(c(x, y), s_{\mathrm{raw}}(x, y), a_1(x, y), \ldots, a_N(x, y)\big),$$

where c is the center, s_{raw} is a scale logit, and $\mathbf{a} = (a_1, \ldots, a_N)$ contains the shape logits. Here $W \in \mathbb{R}^{(N+2)\times d}$ and $b \in \mathbb{R}^{N+2}$. We suppress the dependence on (x, y) below.

The scale and shape logits are transformed as

$$s = s_{\min} + s_{\max}\sigma(s_{\mathrm{raw}}/T),$$
$$\omega_i = \mathrm{softmax}(\mathbf{a})_i, \qquad \psi_i = \sum_{k=1}^{i} \omega_k,$$

where $s_{\min} > 0$ is the scale floor, $s_{\max} > 0$ is the available additive range, and $T > 0$ is the scale temperature. The positive increments ω_i follow the non-crossing construction used in distributional reinforcement learning (Zhou et al., 2020).

Let $\bar{\psi} = N^{-1}\sum_{k=1}^{N} \psi_k$. The support values are

$$q_i = c + s(\psi_i - \bar{\psi}), \qquad i = 1, \ldots, N.$$

Because $\omega_i > 0$, adjacent support values satisfy $q_i - q_{i-1} = s\omega_i > 0$. Centering the cumulative increments also gives $N^{-1}\sum_i q_i = c$. Together, these properties enforce strict ordering and make c the support mean without post-hoc sorting or a monotonicity penalty.

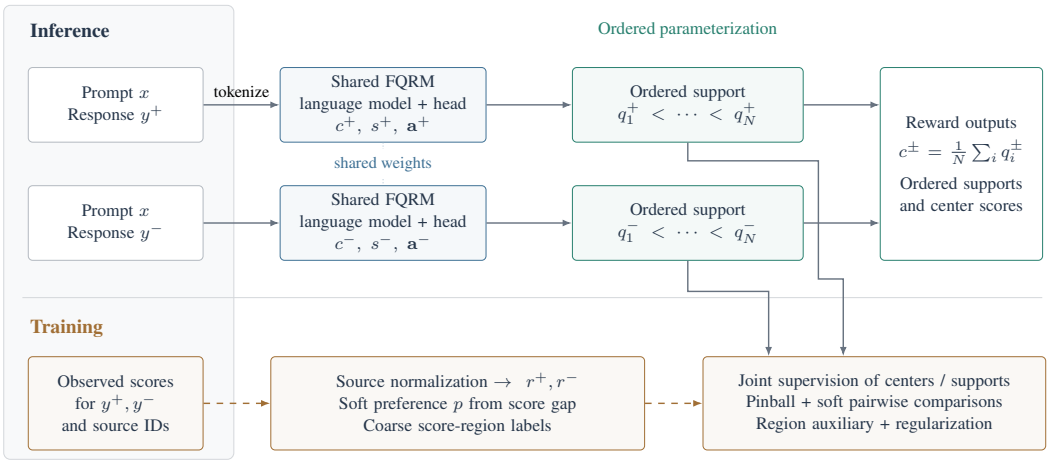

Figure 1: FQRM training and inference workflow.

The center c determines the overall reward location, the scale s controls its spread, and the increments ω_i determine the relative spacing of the support values. We interpret the resulting equally weighted point masses $N^{-1}\sum_i \delta_{q_i}$ as an empirical reward distribution. Within its finite resolution and bounded span, variable support spacing allows the representation to approximate asymmetric or multimodal shapes without prescribing a Gaussian family.

The center, scale, and increments respectively control reward location, spread, and shape. Directly predicting N values could produce unordered supports; the cumulative-softmax construction enforces ordering by design.

As shown in Figure 1, the shared model processes (x, y^+) and (x, y^-) separately and produces $(c^+, \mathbf{q}^+)$ and $(c^-, \mathbf{q}^-)$. The next subsection uses $r^{\pm}$ to supervise the individual supports, p to compare their centers and support values, and $z^{\pm}$ to constrain their coarse reward regions. At inference time, none of these targets is required, and responses are ranked using $c(x, y)$.

3.3 Training Objective

FQRM is trained with complementary signals on the same ordered support. The pairwise signal determines how two responses should be ranked, the graded-score signal anchors each support to the observed reward scale, and auxiliary structural terms keep the support stable and interpretable. The distinction is conceptual: one objective jointly updates the center, spread, and shape.

For a pair (x, y^+, y^-), let p denote the soft preference target obtained from the normalized score difference. Distributional comparison uses all pairwise combinations of the two supports, with a smooth win probability matched to p, rather than comparing only their centers. The center difference is trained against the same target, preserving compatibility with conventional scalar reward-model interfaces. The exact pairwise probability and its derivatives are given in Appendix C.

The normalized scores additionally anchor the support through the pinball regression described in Section 3.1. Rather than treating the score as a separate scalar target, this term pulls the ordered support toward the observed reward level at its nominal quantile positions. Finally, a coarse low/middle/high region anchor and compact regularizers stabilize the support scale, width, and center. The complete minibatch objective can therefore be written compactly as

$$\mathcal{L}_{\text{mixed-score}} = \alpha_{\text{pin}}\mathcal{L}_{\text{pinball}} + \alpha_{\text{pair}}\left(\mathcal{L}_{\text{center}} + \mathcal{L}_{\text{dist-win}}\right) + \alpha_{\text{anchor}}\mathcal{L}_{\text{anchor}} + \mathcal{L}_{\text{reg}}, \qquad (1)$$

where $\alpha_{\text{pin}}, \alpha_{\text{pair}}, \alpha_{\text{anchor}}$ are nonnegative weights. The region anchor links the support to the low, middle, and high score regions, while $\mathcal{L}_{\text{reg}}$ stabilizes its scale, width, and center. We average this

mixed objective over each minibatch and defer exact definitions, derivatives, and coefficient settings to Appendices C and B.3, which provide the complete training specification.

4 THEORY

This section presents two representation-level results for FQRM's ordered reward support. The first establishes the approximation power of finite supports, whereas the second contrasts them with a fixed parametric family. Together, they show how FQRM can represent distributional information without fixing a parametric reward shape before the data are observed.

Proposition 1 (Ordered supports approximate integrable reward laws)**.** *Let P be a distribution on $\mathbb{R}$ with finite first moment and quantile function Q_P. The midpoint empirical measures*

$$P_N = \frac{1}{N}\sum_{i=1}^{N} \delta_{Q_P((i-\frac{1}{2})/N)}$$

converge to P in W_1 as $N \to \infty$.

The proposition shows that a one-dimensional reward distribution can be represented by finitely many ordered support values placed at its quantiles. As the support size grows, the resulting empirical law converges to the target law in Wasserstein distance. FQRM can therefore represent location, spread, skew, and other aspects of response-level reward shape with a finite output. For the implemented bounded-scale head, the guarantee applies to target laws within the configured span, which can be enlarged for broader reward ranges. Thus, FQRM can refine its reward representation through ordered support values rather than encode each response as a point score.

Proposition 2 (A misspecified family has irreducible approximation error)**.** *Let $\mathcal{P}_1(\mathbb{R})$ denote probability laws with finite first moment, and let $\mathcal{F} \subseteq \mathcal{P}_1(\mathbb{R})$ be a fixed family and $P^\star \in \mathcal{P}_1(\mathbb{R})$ a target law. Here W_1 is the Wasserstein-1 distance and the overline denotes closure in this metric. If $P^\star \notin \overline{\mathcal{F}}^{W_1}$, then*

$$\inf_{P\in\mathcal{F}} W_1(P^\star, P) > 0.$$

The result formalizes a general source of representation error: if a target reward law lies outside the closure of a fixed family, additional training cannot remove the resulting mismatch within that family. Thus, a Gaussian output can be poorly suited to a skewed or otherwise non-Gaussian target, although the proposition does not assert that every empirical reward law is non-Gaussian. Combined with Proposition 1, this motivates using an ordered support as a flexible alternative whose approximation error can be refined with the support resolution, while its mean remains available through the standard scalar reward interface. Proofs are given in Appendix A.

Proposition 1 concerns what ordered supports can represent, whereas Proposition 2 concerns the irreducible error of a fixed family. Together they motivate a finite support that remains flexible across reward shapes; the experiments test whether this flexibility improves ranking.

5 EXPERIMENTAL SETUP AND RESULTS

This section evaluates whether FQRM can use information that binary scalar training discards, whether its ordered representation improves ranking on ambiguous comparisons, and whether the observed gains persist across model variants and random seeds. We report controlled benchmark comparisons, category-level behavior, robustness to superficial perturbations, and a small downstream compatibility study.

5.1 EXPERIMENTAL SETUP

Training data. The shared score-bearing mixture combines HelpSteer2 (Wang et al., 2024b), preference_700K, and binarized UltraFeedback. HelpSteer2 is a multi-dimensional helpfulness dataset with response-level ratings that provide graded supervision. Preference_700K contains large-scale chosen/rejected comparisons together with scalar quality scores, supplying both pairwise and score-bearing examples. UltraFeedback aggregates model-generated feedback over diverse prompts and

Table 1: Main reward-model comparison.

Model	RewardBench-v1 avg	RewardBench-2 avg	RM-Bench avg
Scalar BT	0.508 ± 0.012	0.173 ± 0.010	0.458 ± 0.010
Scalar regression (MSE)	0.730 ± 0.169	0.460 ± 0.193	0.643 ± 0.121
Scalar regression + BT	0.722 ± 0.169	0.466 ± 0.200	0.641 ± 0.126
Gaussian PURM	0.620 ± 0.057	0.179 ± 0.119	0.377 ± 0.169
FQRM	$\mathbf{0.876 \pm 0.006}$	$\mathbf{0.609 \pm 0.010}$	$\mathbf{0.714 \pm 0.005}$

contributes additional scored responses that broaden coverage beyond a single annotation source. We select an ambiguity-enriched subset with a smaller clear-comparison pool and apply source-wise score normalization; all controlled baselines use the same selected records.

Evaluation benchmarks and metrics. We evaluate RewardBench-v1 (Lambert et al., 2025), RewardBench 2 (Malik et al., 2025), and RM-Bench (Liu et al., 2024b). RewardBench-v1 provides broad chat, safety, and reasoning-oriented pairwise comparisons for standard reward-model evaluation. RewardBench 2 extends this evaluation with more targeted categories, including factuality, focus, math, precise instruction following, safety, and ties. RM-Bench focuses on finer domain-level distinctions such as coding, mathematics, chat, and safety behavior. We report macro-averaged pairwise accuracy overall and by benchmark category/domain. Additional analyses measure sensitivity to surface perturbations and, in a small pilot, downstream policy preference rates.

Baselines. Scalar BT is the conventional point-valued Bradley–Terry reward model. Scalar regression predicts normalized response scores with MSE, and scalar regression+BT combines this objective with soft pairwise supervision. Gaussian PURM (Sun et al., 2025) predicts a Gaussian reward distribution. FQRM is our response-level ordered-support model. All main repeated recipes use the same backbone, selected data, nominal update budget, and final-checkpoint evaluation rule. Detailed preprocessing, optimization, seeds, and checkpoint conventions are given in Appendix B.

5.2 Main Results

This experiment tests the primary claim that a response-level ordered reward representation can combine graded scores and preference supervision more effectively than point-valued or fixed-family alternatives. We train every recipe on the same source-normalized mixture and evaluate the final 2000-step checkpoints on three independent benchmark families. The comparison asks whether the complete FQRM recipe improves average ranking quality and whether that improvement is consistent across random seeds, without selecting checkpoints using benchmark results.

Across three seeds, FQRM reaches 87.6 ± 0.6, 60.9 ± 1.0, and 71.4 ± 0.5 percent on RewardBench-v1, RewardBench 2, and RM-Bench. These scores exceed the strongest non-FQRM recipe by 14.6, 14.3, and 7.1 points, respectively; the corresponding FQRM deviations are all below one point. The effect is therefore not a single-seed peak but a repeatable advantage of the complete FQRM recipe.

Taken together, these results show that the ordered support is useful not only as an alternative output format, but as a common space in which heterogeneous feedback can be combined. Graded scores provide information about reward location, soft preferences preserve comparison strength, and the ordered support retains variation that would otherwise be compressed into a single point estimate. The resulting model improves ranking across all three benchmark families while maintaining a scalar center for standard reward-model interfaces, and the small seed deviations indicate that this benefit is repeatable under the shared training recipe rather than dependent on one favorable run.

The comparison is deliberately endpoint-based: all methods are evaluated at their final 2000-step checkpoints, without benchmark-driven checkpoint selection or method-specific early stopping. This common rule avoids favoring recipes whose curves peak or fluctuate at different times, especially when their data sources use different score scales. The reported margins and seed variation therefore compare complete training recipes at the same nominal endpoint. This mirrors deploy-

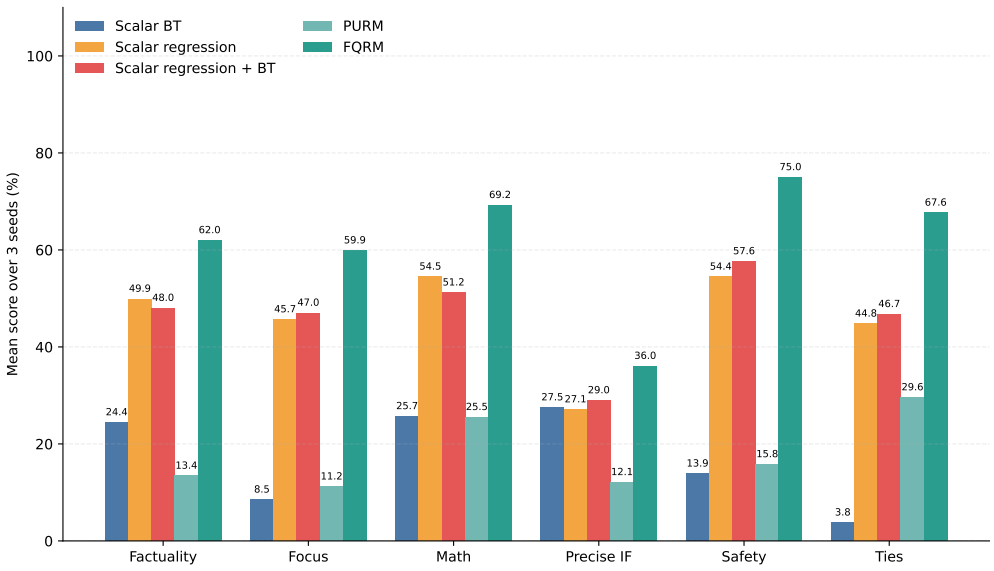

Figure 2: RewardBench-2 categories.

ment, where a checkpoint is fixed before external benchmark outcomes are observed, so the comparison reflects both ranking quality and reliability under a common training protocol.

The benchmarks probe complementary settings: RewardBench-v1 provides broad preference ranking, RewardBench 2 separates factuality, math, instruction following, and safety, and RM-Bench emphasizes subtle domain distinctions. Improvement across all three is consistent with preserving scalar location while retaining extra structure for close or multi-criteria comparisons.

Having established the aggregate result, we next ask whether the gain is confined to one category or extends across evaluation dimensions. Figure 2 reports the three-seed category means for RewardBench 2, with the RewardBench-v1 and RM-Bench breakdowns in the appendix. The categories probe distinct failure modes: factuality and math require content-sensitive comparisons, focus and precise instruction following require task matching, and safety and ties emphasize boundary cases in which scalar margins provide limited information about comparison strength.

FQRM is the strongest model in all six categories. Its largest gains occur on Ties (+20.9 points) and Safety (+17.3), while it also improves Math (+14.8) and Factuality (+12.1). The concentration of gains in close-comparison and safety-sensitive categories is informative: these are precisely the settings where a single binary ordering hides how far apart the responses are and where multiple quality dimensions can conflict. By retaining a graded center and an ordered support, FQRM can represent these distinctions instead of forcing them into one hard margin. The appendix breakdowns preserve the same top ranking across RewardBench-v1 and RM-Bench, although the margin depends on the domain. The category experiment therefore shows that the gain is not confined to a single benchmark subset. Its concentration in content-sensitive and boundary-case comparisons is consistent with the ordered support being useful when a binary ordering is likely to discard more information. In particular, the advantage on Ties suggests that FQRM is better able to preserve information when the preferred and rejected responses have similar overall quality, while the gains on Safety indicate that the representation remains useful when a response may be acceptable on one dimension but problematic on another. The gains on Math and Factuality further show that this is not limited to preference ambiguity: the same representation also helps when correctness depends on detailed response content. Together, these patterns connect the aggregate benchmark improvement to the kinds of multi-factor comparisons for which scalar compression is most restrictive.

The supplementary RewardBench-v1 and RM-Bench breakdowns (Figures 4 and 5) show the same ordering at the top: FQRM has the highest mean in all four RewardBench-v1 sections and all five RM-Bench domains. In RM-Bench, however, the margin over the stronger scalar regression control

ranges from only 1.9 points on Code to 15.8 on Safety-refusal. Taken together, these results support broad ranking gains for the complete recipe while showing that the benefit varies by domain.

5.3 Support Ablation

The main comparison establishes the overall benefit of FQRM; we next isolate whether retaining multiple ordered support values contributes beyond the shared training recipe. The standard FQRM configuration uses $N = 32$ support values. The $N = 1$ control uses the same backbone, source-normalized data, supervision terms, and nominal 2000-step budget, but collapses the support to a single learned value. It therefore removes response-level variation while keeping the scalar interface and the rest of the pipeline unchanged. The three-seed results are summarized in Table 2. FQRM exceeds the collapsed model by 15.2, 14.2, and 7.1 points on RewardBench-v1, RewardBench 2, and RM-Bench, respectively. The gap is accompanied by much larger seed deviations for $N = 1$ (24.5, 24.7, and 13.7 points) than for FQRM (below one point on all three benchmarks). Both variants use the same normalization, soft preference targets, score anchoring, region labels, backbone, and optimization schedule; the only representational change is whether reward mass can be distributed across ordered support values. Thus, the ablation tests the role of support resolution directly rather than changing the data or training objective. The results indicate that multiple supports provide a more expressive and stable interface for heterogeneous score supervision, while a single support loses both variation and robustness across seeds. This distinction matters because absolute scores constrain where a response should lie, whereas soft preferences constrain how responses should be ordered. With $N = 1$, both signals compete for one scalar degree of freedom; with $N = 32$, the center can preserve the coarse score level while remaining supports absorb response-specific spread. The lower seed dispersion therefore suggests that the mixed objective is easier to optimize, not merely that the output is larger.

Table 2: Support ablation.

Model	RewardBench-v1	RewardBench-2	RM-Bench
$N = 1$ FQRM	0.722 ± 0.245	0.473 ± 0.247	0.640 ± 0.137
FQRM ($N = 32$)	0.874 ± 0.005	0.615 ± 0.001	0.710 ± 0.005

5.4 Surface-Perturbation Robustness

The preceding experiments establish ranking quality across natural benchmark families and categories. We next test a different property: whether the learned reward is insensitive to simple, non-semantic surface changes. Aggregate benchmark accuracy does not directly reveal whether a reward model is reacting to response content or to superficial length and formatting cues. We therefore run a development-only surface-perturbation probe. For each benchmark example, we keep the prompt and the original preferred response fixed, and use a script to create a second response by applying exactly one deterministic string-level edit: *verbosity* appends the generic sentence "In conclusion, the answer above is important and should be considered carefully."; *format* wraps the unchanged response between Markdown-style "### Answer" and "### Summary" sections and adds a generic summary sentence; and *length* appends eight repetitions of "Additional context." without adding task-relevant content. Thus the original response is the intended preferred member of every constructed pair, while the perturbed copy differs only through the selected surface edit. Each benchmark contributes 600 pairs: 200 for each perturbation mode.

The score in this probe is the fraction of pairs for which a model assigns a higher reward to the unperturbed response than to its perturbed copy. Higher values therefore mean that the model preserves the original ranking under the tested edit; they do not measure general robustness or calibrated human preference. Figure 3 summarizes the five-model comparison. FQRM reaches 93.8%, 94.8%, and 97.9%, remaining above every baseline on each benchmark and exceeding the next-best model by 13.1, 14.1, and 26.3 points. The largest separation occurs on RM-Bench, where domain-specific comparisons provide more opportunities for length and formatting heuristics. This pattern is consistent with FQRM being less sensitive to the tested surface cues and preserving the original content-based ordering more reliably. In other words, the same representation that improves difficult natural comparisons also produces a more stable ranking under these controlled edits. This is

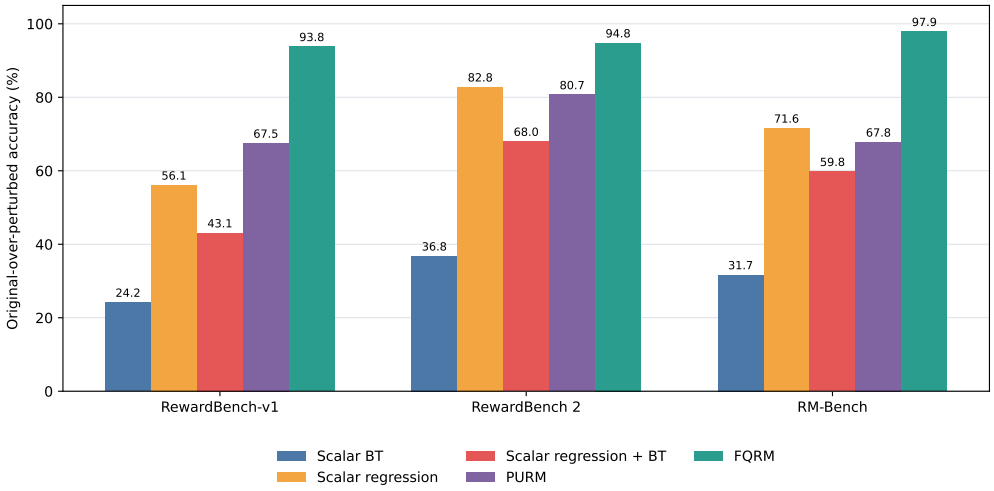

Figure 3: Surface-perturbation accuracy.

especially relevant for reward-model deployment, because a policy can easily discover superficial features that correlate with reward on the training mixture but are unrelated to the underlying preference. Preserving the original ranking after controlled length and formatting edits makes such reward shortcuts less attractive. Because the pairs are synthetic, this result supports robustness only to the tested edits rather than general invariance. Details appear in Appendix B.4.

5.5 Downstream Compatibility

After measuring offline ranking and surface sensitivity, we finally consider whether the learned reward can support downstream policy optimization. This is a compatibility check rather than a new benchmark comparison. The FQRM center provides a directly compatible scalar reward for PPO, so the model can be inserted into a standard reward-model pipeline without changing the policy optimizer or its interface. The ordered support additionally exposes response-level dispersion and shape information that is unavailable to a point-valued reward. A downstream PPO implementation can use this information through an auxiliary scalarization: for example, it can combine the center with a dispersion penalty, a risk-sensitive summary, or a sensitivity estimate derived from the support and its gradients. These components do not alter the training objective of FQRM; they provide optional ways to modulate how strongly PPO trusts individual reward queries. Our pilot is intended to demonstrate this compatibility and the use of distributional information in downstream reward queries; the specific scalarizers and numerical results are deferred to Appendix D.

In the simplest deployment, PPO receives only $c(x, y)$. When uncertainty-aware optimization is desired, the same query can expose support width, a risk-sensitive summary, or a gradient-based sensitivity score for reward weighting or conservative scalarization. This preserves a single reward-model interface while letting users choose which distributional signal to expose during policy learning.

6 Conclusion and Future Work

FQRM brings the distributional-RL viewpoint to response-level reward modeling. It represents each response with an ordered reward support while retaining a scalar-compatible center, and combines pairwise order, soft preference strength, and graded response scores in a shared representation. In a controlled score-bearing, ambiguity-enriched setting, FQRM achieves the strongest mean performance across RewardBench-v1, RewardBench 2, and RM-Bench among the repeated recipes, while controlled representation and robustness studies clarify the scope of these findings. Future work should test larger downstream PPO settings, human-judged calibration and ambiguity probes, and controlled variation of the representation, objective, and data recipe.

Ethics Statement

This work studies reward modeling for language-model alignment using public or standard research datasets and automated benchmark evaluations. The method could be used to improve preference ranking, but reward models can also encode dataset bias or encourage over-optimization when used in downstream policy training. We therefore evaluate category-level behavior, separate RM ranking evidence from downstream PPO pilots, and avoid claiming that the learned distributions are calibrated human uncertainty.

Reproducibility Statement

The paper specifies the FQRM parameterization, training objectives, data-selection rule, score normalization, evaluation protocol, and analytic proofs. The appendix gives the exact normalization constants, checkpoint settings, surface-perturbation construction, and PPO pilot configuration. The final repeated-run comparison uses the same selected examples, source-wise score normalization, backbone, LoRA configuration, effective batch size, learning rate, nominal warmup, step budget, and final-checkpoint rule for the scalar, PURM, and FQRM recipes. The accompanying code and configuration files can be released anonymously with the submission; all reported results are generated from saved evaluation logs and checkpoint paths recorded in the manuscript.

AI Use Statement

Generative AI tools were used during this work to assist with experiment planning and methodology discussion, implementation and modification of training/evaluation scripts, inspection and interpretation of experimental logs, literature-search assistance, and drafting and editing of the manuscript. The authors independently reviewed and verified the code changes, mathematical statements and proofs, dataset-processing decisions, experimental configurations, numerical results, citations, and final manuscript. The authors take responsibility for all content and claims in this submission.

References

Marc G. Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning*, 2017.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, 2017.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. In *International Conference on Learning Representations*, 2024.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International Conference on Machine Learning*, 2018a.

Will Dabney, Mark Rowland, Marc G. Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *AAAI Conference on Artificial Intelligence*, 2018b.

Nicolai Dorka. Quantile regression for distributional reward models in rlhf. *arXiv preprint arXiv:2409.10164*, 2024. URL https://arxiv.org/abs/2409.10164.

Jacob Eisenstein, Chirag Nagpal, Alekh Agarwal, Ahmad Beirami, Alexander D'Amour, Krishnamurthy Dvijotham, Adam Fisch, Katherine Heller, Stephen Pfohl, Deepak Ramachandran, Peter Shaw, and Jonathan Berant. Helping or herding? reward model ensembles mitigate but do not eliminate reward hacking. *arXiv preprint arXiv:2312.09244*, 2024.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. *arXiv preprint arXiv:2210.10760*, 2022.

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. In *Findings of the Association for Computational Linguistics: NAACL*, 2025. doi: 10.18653/v1/2025.findings-naacl.96.

Tianqi Liu, Wei Xiong, Jie Ren, Lichang Chen, Junru Wu, Rishabh Joshi, Yang Gao, Jiaming Shen, Zhen Qin, Tianhe Yu, Daniel Sohn, Anastasiia Makarova, Jeremiah Liu, Yuan Liu, Bilal Piot, Abe Ittycheriah, Aviral Kumar, and Mohammad Saleh. Rrm: Robust reward model training mitigates reward hacking. arXiv preprint arXiv:2409.13156, 2024a.

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. *arXiv preprint arXiv:2410.16184*, 2024b.

Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah A. Smith, Hannaneh Hajishirzi, and Nathan Lambert. Rewardbench 2: Advancing reward model evaluation. arXiv preprint arXiv:2506.01937, 2025. Benchmark collection and leaderboard: `https://huggingface.co/collections/allenai/reward-bench-2`.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Advances in Neural Information Processing Systems*, 2020.

Wangtao Sun, Xiang Cheng, Xing Yu, Haotian Xu, Zhao Yang, Shizhu He, Jun Zhao, and Kang Liu. Probabilistic uncertain reward model. *arXiv preprint arXiv:2503.22480*, 2025.

Binghai Wang, Rui Zheng, Lu Chen, Yan Liu, Shihan Dou, Caishuang Huang, Wei Shen, Senjie Jin, Enyu Zhou, Chenyu Shi, Songyang Gao, Nuo Xu, Yuhao Zhou, Xiaoran Fan, Zhiheng Xi, Jun Zhao, Xiao Wang, Tao Ji, Hang Yan, Lixing Shen, Zhan Chen, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. Secrets of rlhf in large language models part ii: Reward modeling. *arXiv preprint arXiv:2401.06080*, 2024a.

Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. Helpsteer2: Open-source dataset for training top-performing reward models. *arXiv preprint arXiv:2406.08673*, 2024b.

Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii Kuchaiev, and Yi Dong. Helpsteer2-preference: Complementing ratings with preferences. In *International Conference on Learning Representations*, 2025.

Yuanzhao Zhai, Han Zhang, Yu Lei, Yue Yu, Kele Xu, Dawei Feng, Bo Ding, and Huaimin Wang. Uncertainty-penalized reinforcement learning from human feedback with diverse reward lora ensembles. *arXiv preprint arXiv:2401.00243*, 2024.

Michael J. Q. Zhang, Zhilin Wang, Jena D. Hwang, Yi Dong, Olivier Delalleau, Yejin Choi, Eunsol Choi, Xiang Ren, and Valentina Pyatkin. Diverging preferences: When do annotators disagree and do models know? *arXiv preprint arXiv:2410.14632*, 2024.

Fan Zhou, Jianing Wang, and Xingdong Feng. Non-crossing quantile regression for distributional reinforcement learning. In *Advances in Neural Information Processing Systems*, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/b6f8dc086b2d60c5856e4ff517060392-Abstract.html`.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

A PROOFS

A.1 PROOF OF PROPOSITION 1

Because P has finite first moment, $Q_P \in L^1(0,1)$. In one dimension, the Wasserstein-1 distance has the quantile representation

$$W_1(P_N, P) = \int_0^1 |Q_{P_N}(u) - Q_P(u)|\, du.$$

The step function Q_{P_N} equals $Q_P((i-\frac{1}{2})/N)$ on the interval $((i-1)/N, i/N]$. Since Q_P is nondecreasing, these midpoint step approximations converge to Q_P almost everywhere. Together with $Q_P \in L^1(0,1)$, the standard L^1 approximation result for monotone integrable functions gives convergence in L^1. Hence the integral above tends to zero. More generally, the same identity implies that any learned support whose quantile function $\widehat{Q}_N$ converges to Q_P in $L^1(0,1)$ also converges in W_1:

$$W_1(\widehat{P}_N, P) = \int_0^1 |\widehat{Q}_N(u) - Q_P(u)|\, du \to 0.$$

A.2 PROOF OF PROPOSITION 2

Suppose for contradiction that

$$\inf_{P \in \mathcal{F}} W_1(P^\star, P) = 0.$$

Then there exists a sequence $P_n \in \mathcal{F}$ with $W_1(P^\star, P_n) \to 0$. By definition of closure, this means $P^\star \in \overline{\mathcal{F}}^{W_1}$, contradicting the assumption. Hence the infimum must be strictly positive.

A.3 ADDITIONAL PROOF DETAILS

The two propositions use only the one-dimensional quantile representation of the Wasserstein distance and the definition of metric closure. They are representation-level facts, not claims about finite-sample estimation, optimization, calibration, or downstream policy quality. The short arguments are included here to make the scope of the theory explicit. To connect the unrestricted supports to the implemented head, take any strictly ordered values $v_1 < \cdots < v_N$ of span $d = v_N - v_1 < s_{\min} + s_{\max}$. Because this condition permits a choice $s \in (\max\{d, s_{\min}\}, s_{\min} + s_{\max})$, set $p_1 = 1 - d/s$ and $p_i = (v_i - v_{i-1})/s$ for $i \geq 2$, and set $c = N^{-1} \sum_i v_i$. These positive increments sum to one and the centered cumulative construction recovers $q_i = v_i$. Repeated support values can be approximated by arbitrarily small strictly ordered perturbations. This establishes the stated bounded-span qualification, rather than universality at a fixed scale bound.

A.4 SUPPLEMENTARY PINBALL-LOSS DERIVATION

Let R have CDF F and finite first moment, let $\alpha \in (0,1)$, and define $L_\alpha(q) = \mathbb{E}[\ell_\alpha(q; R)]$, where $\ell_\alpha(q; R) = \rho_\alpha(R - q)$ uses the pinball loss from Section 3.1 and $(v)_+ = \max(v, 0)$ denotes the positive part. The pointwise loss can be rewritten as

$$\ell_\alpha(q; R) = \alpha(R - q)_+ + (1 - \alpha)(q - R)_+.$$

Hence L_α is convex in q. For $h > 0$, Fubini's theorem gives

$$L_\alpha(q + h) - L_\alpha(q) = \int_q^{q+h} \{F(t) - \alpha\}\, dt,$$

and similarly

$$L_\alpha(q) - L_\alpha(q - h) = \int_{q-h}^{q} \{F(t) - \alpha\}\, dt.$$

Taking $h \downarrow 0$, the left and right derivatives are

$$\partial_- L_\alpha(q) = F(q^-) - \alpha, \qquad \partial_+ L_\alpha(q) = F(q) - \alpha.$$

Since L_α is convex, q minimizes L_α if and only if zero lies in its subdifferential:

$$F(q^-) - \alpha \le 0 \le F(q) - \alpha.$$

This is exactly $F(q^-) \le \alpha \le F(q)$, the quantile-set condition. If the CDF crosses α uniquely, this set contains only $Q_R(\alpha)$.

A.5 Supplementary collapsed-support derivation

If all preferred support values equal $r_\theta(x, y^+)$ and all rejected support values equal $r_\theta(x, y^-)$, every term in the double sum has the same value,

$$\sigma\left(\frac{r_\theta(x, y^+) - r_\theta(x, y^-)}{T_{\text{win}}}\right).$$

The average of N^2 identical terms is that value. With $T_{\text{win}} = 1$, this is the standard Bradley–Terry likelihood. For $N = 1$, $q_1 = c$ exactly. For $N > 1$, collapse is a limiting construction because finite shape logits enforce strictly positive spacings. This equivalence concerns the pairwise likelihood, not the full objective with score and auxiliary losses.

A.6 Supplementary distributional-ranking derivation

Assume independent reward draws satisfy $P(R_+ = R_-) = 0$, and let $\widehat{p}_{K,T_{\text{win}}}$ denote the sigmoid comparison averaged over the K^2 pairs of midpoint quantile values.

Let $U, V \sim \text{Unif}(0, 1)$ be independent. Then $Q_+(U) \sim P_+$ and $Q_-(V) \sim P_-$. For fixed $T_{\text{win}} > 0$, the function

$$f_{T_{\text{win}}}(u, v) = \sigma\left(\frac{Q_+(u) - Q_-(v)}{T_{\text{win}}}\right)$$

is bounded by 0 and 1. Quantile functions are monotone, so each has at most countably many discontinuities. Consequently, the discontinuity set of $f_{T_{\text{win}}}$ is contained in the corresponding vertical and horizontal discontinuity lines and has Lebesgue measure zero. The midpoint empirical measures converge weakly to the uniform measure, and the bounded almost-everywhere continuity theorem therefore gives

$$\lim_{K\to\infty} \widehat{p}_{K,T_{\text{win}}} = \int_0^1 \int_0^1 \sigma\left(\frac{Q_+(u) - Q_-(v)}{T_{\text{win}}}\right) dudv.$$

As $T_{\text{win}} \downarrow 0$, the integrand converges pointwise to $\mathbf{1}\{Q_+(u) > Q_-(v)\}$ wherever the two values are unequal. The no-tie assumption makes the equality set have measure zero under U, V. By dominated convergence,

$$\lim_{T_{\text{win}}\downarrow 0} \int_0^1 \int_0^1 \sigma\left(\frac{Q_+(u) - Q_-(v)}{T_{\text{win}}}\right) dudv = P(Q_+(U) > Q_-(V)) = P(R_+ > R_-).$$

B Reproducibility Details

B.1 Score normalization constants

The source-normalized runs fit a median/IQR transform on each training source, clip normalized scores to $[-2, 2]$, recompute soft preference probabilities, and use normalized anchor thresholds $[-0.5, 0.5]$. The fitted statistics used in those runs are: HelpSteer2 median 0.8 and IQR 1.0; preference_700K median 4.5 and IQR 2.0; and UltraFeedback median 8.0 and IQR 1.5. These statistics define the final source-normalized comparison. Earlier HelpSteer2-only score-supervised dry-runs used a single-source affine normalization and are retained only as development baselines.

B.2 Representation ablation

The representation ablation uses the same source-normalized data and nominal 2000-step budget as the standard FQRM configuration with $N = 32$ support values. The $N = 1$ control collapses the ordered support to a single value while retaining the same backbone and

supervision terms. The three seeds are 42, 123, and 456; the resulting table is reported in the main text (Table 2). Entries use sample standard deviations and the logged overall fields `Average accuracy`, `average`, and `overall_avg`, respectively. The FQRM seed-42 reference is the `qurm-curve-relog-2000` evaluation; the other seeds use the `qurm-sourcenorm-seed123-2000` and `qurm-sourcenorm-seed456-2000` evaluations. The gap supports the role of multiple ordered support values in combining graded scores with pairwise supervision.

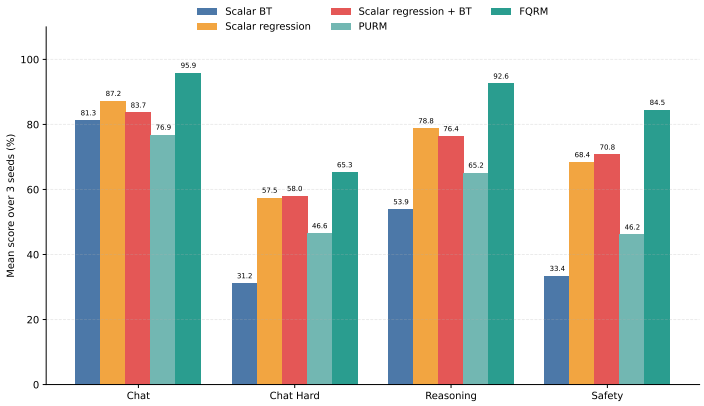

Figure 4: RewardBench-v1 category analysis.

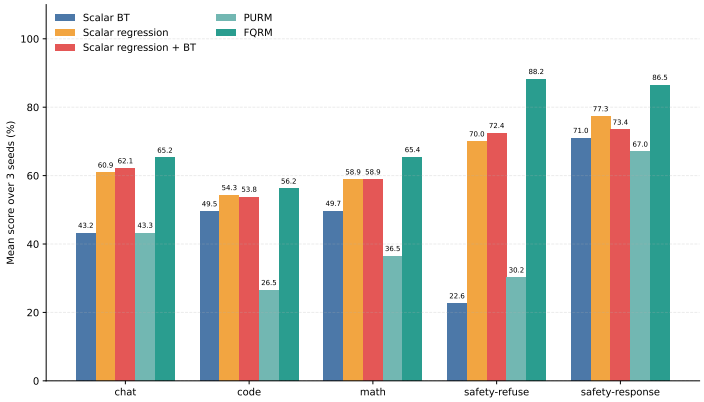

Figure 5: RM-Bench domain analysis.

B.3 Main RM training recipe

The main RM runs use the same backbone, LoRA configuration, source-normalized selected mixed-3src records, and nominal training budget for scalar, PURM, and FQRM comparisons. FQRM trains with mixed preference and score supervision through pinball regression and scalar and distributional preference losses. The reported runs additionally use the score-region auxiliary $\mathcal{L}_{\text{anchor}}$, whose targets are derived from normalized scores using thresholds $[-0.5, 0.5]$. This is a training constraint, not an identifiability guarantee or an independently validated source of performance gains. The reported checkpoint is the final saved 2000-step checkpoint for each run; the source-normalized seed-42 FQRM run was subsequently continued to 4000 steps, with its step-2800 checkpoint used only for the downstream PPO pilot.

B.4 Surface-perturbation probe

Aggregate benchmark accuracy can hide reliance on superficial artifacts. We construct synthetic pairs from all three benchmark families by keeping the original prompt and preferred response while

applying one surface edit: a generic verbosity suffix, markdown formatting, or repeated filler context. We measure how often each model ranks the unperturbed response above its perturbed copy. The values below come from one development checkpoint per recipe and are not human-annotated results.

Benchmark	Scalar BT	Scalar reg.	Scalar reg.+BT	PURM	FQRM
RewardBench-v1	24.2	56.1	43.1	67.5	93.8
RewardBench 2	36.8	82.8	68.0	80.7	94.8
RM-Bench	31.7	71.6	59.8	67.8	97.9

Table 3: Synthetic surface-perturbation probe.

B.5 Artifact-robustness probe construction

The artifact probe is built from three benchmark families: RewardBench-v1, RewardBench 2, and RM-Bench. For each family we load the benchmark examples, keep the original prompt and preferred response, and synthesize a comparison response by applying one of three surface-only edits to the preferred answer: *verbosity* appends a generic conclusion sentence, *format* wraps the response in markdown-style headers, and *length* appends repeated filler context. The construction is intended to preserve the task content while changing surface form, but the resulting pairs are synthetic and may not be perfectly semantics-preserving. We then evaluate the five reward-model recipes on the same synthetic pairs with the same pairwise scoring scripts used for the main benchmark evaluation. The concrete construction is implemented in `scripts/build_artifact_robustness_pairs.py` and `scripts/run_rm_artifact_robustness.sh`.

C Gradient Derivations

This appendix records analytic gradients for the supervised objectives. The main text gives the corresponding loss definitions and an overview of their roles.

C.1 Distributional Win Loss

Write $q_i^+ = q_i(x, y^+)$ and $q_j^- = q_j(x, y^-)$ for the two responses, let p be the soft preference target, and let $T_{\text{win}} > 0$ be the comparison temperature from the main text. Let

$$\hat{p}_{ij} = \sigma\left(\frac{q_i^+ - q_j^-}{T_{\text{win}}}\right), \qquad \widehat{p} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\hat{p}_{ij},$$

and

$$\mathcal{L}_{\text{dist}-\text{win}} = -p\log\widehat{p} - (1-p)\log(1-\widehat{p}).$$

Then

$$\frac{\partial\mathcal{L}_{\text{dist}-\text{win}}}{\partial\widehat{p}} = \frac{\widehat{p}-p}{\widehat{p}(1-\widehat{p})}.$$

Moreover,

$$\frac{\partial\widehat{p}}{\partial q_i^+} = \frac{1}{N^2 T_{\text{win}}}\sum_{j=1}^{N}\hat{p}_{ij}(1-\hat{p}_{ij}), \qquad \frac{\partial\widehat{p}}{\partial q_j^-} = -\frac{1}{N^2 T_{\text{win}}}\sum_{i=1}^{N}\hat{p}_{ij}(1-\hat{p}_{ij}).$$

Therefore

$$\frac{\partial\mathcal{L}_{\text{dist}-\text{win}}}{\partial q_i^+} = \frac{\widehat{p}-p}{\widehat{p}(1-\widehat{p})}\,\frac{1}{N^2 T_{\text{win}}}\sum_{j=1}^{N}\hat{p}_{ij}(1-\hat{p}_{ij}),$$

and the rejected-side derivative has the opposite sign. When $\widehat{p} < p$, gradient descent directly on the support values raises preferred-side values and lowers rejected-side values. The actual head shares parameters across support points, so these are support-space derivatives, not a guarantee about each point's displacement after a parameter update.

C.2 CENTER BCE LOSS

Let $d_c = c(x, y^+) - c(x, y^-)$. Then

$$\mathcal{L}_{\text{center}} = -p \log \sigma(d_c) - (1-p) \log(1 - \sigma(d_c)).$$

Hence

$$\frac{\partial \mathcal{L}_{\text{center}}}{\partial d_c} = \sigma(d_c) - p.$$

This is the standard logistic residual.

C.3 PINBALL LOSS

For a single target r and quantile q_i,

$$\rho_{\tau_i}(r - q_i) = \max(\tau_i(r - q_i), (\tau_i - 1)(r - q_i)).$$

A subgradient with respect to q_i is

$$\partial_{q_i} \rho_{\tau_i}(r - q_i) = \begin{cases} -\tau_i, & r > q_i, \\ 1 - \tau_i, & r < q_i, \end{cases}$$

with interval $[-\tau_i, 1 - \tau_i]$ at equality. Equivalently,

$$\partial_{q_i} \rho_{\tau_i}(r - q_i) = \mathbf{1}\{r < q_i\} - \tau_i.$$

Thus, each support value is pulled upward when it lies below the target and downward when it lies above the target. The constrained FQRM head couples these updates through its center, scale, and softmax shape parameterization.

C.4 AUXILIARY SCORE-REGION LOSS

Here $a < b$ are fixed normalized-score thresholds (not shape logits), $\tau_{\text{anchor}} > 0$ is the region-smoothing temperature, and $\sigma'(u) = \sigma(u)(1 - \sigma(u))$ is the sigmoid derivative. Let

$$F_\theta(t \mid x, y) = \frac{1}{N} \sum_{i=1}^{N} \sigma\left(\frac{t - q_i(x, y)}{\tau_{\text{anchor}}} \right),$$

with

$$p_{\text{low}} = F_\theta(a), \qquad p_{\text{mid}} = F_\theta(b) - F_\theta(a), \qquad p_{\text{high}} = 1 - F_\theta(b).$$

For a region label $z \in \{0, 1, 2\}$ (low, middle, high), define the per-response loss $\mathcal{L}_{\text{anchor}} = -\log p_z$; the training objective averages this loss over the two responses. Since

$$\frac{\partial}{\partial q_i} \sigma\left(\frac{t - q_i}{\tau_{\text{anchor}}} \right) = -\frac{1}{\tau_{\text{anchor}}} \sigma'\left(\frac{t - q_i}{\tau_{\text{anchor}}} \right),$$

we have

$$\frac{\partial F_\theta(t)}{\partial q_i} = -\frac{1}{N \tau_{\text{anchor}}} \sigma'\left(\frac{t - q_i}{\tau_{\text{anchor}}} \right).$$

Thus, before the implementation's probability clipping and renormalization, the anchor loss moves support values according to whether mass should shift below a, between a and b, or above b. The constrained FQRM head couples these updates through its center, scale, and softmax shape parameterization. In code, the three probabilities are clipped to a small positive value and renormalized before the negative log likelihood is evaluated; the displayed derivatives describe the unclipped base probabilities.

D DOWNSTREAM PPO PILOT

We include the PPO results as a downstream implementation diagnostic, not as evidence for the main RM claim. We served the trained FQRM checkpoint through the OpenRLHF reward-server interface and compared the resulting policy with the base Qwen2.5-7B-Instruct policy using the same held-out set of 196 prompts and three repeated LLM-judge runs. The 1024-update runs were continuations from saved actor checkpoints for an additional 512 updates, rather than exact optimizer-state resumes. The uncertainty-shaped reward combines the running-normalized Fréchet distributional term with the Fisher-style sensitivity term; center-only uses the FQRM center alone.

D.1 PPO UNCERTAINTY SCALARIZATION

For completeness, this subsection specifies the two auxiliary uncertainty terms used by the reward server. They are relative regularizers for this downstream pilot, rather than calibrated estimates of human uncertainty. Let $q_i(x, y)$ denote the ith value on the FQRM quantile grid for a generated response (x, y), with $i = 1, \ldots, K$. The center reward passed to PPO is $c(x, y)$, the center of the same predicted distribution.

The data-side term measures dispersion of the predicted quantile shape around its barycenter. In the server's default `frechet_dispersion` mode, the K quantile values for one response are treated as the discrete support of the predicted reward law. Define

$$\bar{q}(x, y) = \frac{1}{K} \sum_{i=1}^{K} q_i(x, y), \qquad \widehat{u}_{\text{data}}(x, y) = \frac{1}{K} \sum_{i=1}^{K} |q_i(x, y) - \bar{q}(x, y)| \,. \tag{2}$$

This is the discrete W_1-style quantile dispersion implemented by `qurm_frechet_dispersion`; it uses the whole quantile grid rather than only the endpoint width. An alternative `history_w1` mode keeps quantile vectors from preceding calls and computes a last-step Wasserstein distance; that mode is documented as a separate runtime option and is not needed for the formula above.

The model-side term is based on the local sensitivity of the reward model. For training examples $j = 1, \ldots, n$, let $g_j = \nabla_\theta r_\theta(x_j, y_j)$ and define the diagonal empirical Fisher artifact

$$\widehat{f}_\ell = \frac{1}{n} \sum_{j=1}^{n} g_{j,\ell}^2 + \lambda_{\text{reg}}. \tag{3}$$

For a new response, the corresponding Fisher sensitivity is

$$u_{\text{model}}^{\text{diag}}(x, y) = \sum_{\ell} \frac{g_\ell(x, y)^2}{\widehat{f}_\ell}. \tag{4}$$

When per-quantile gradients are used, the server computes this quantity for each quantile and averages the resulting scores (the default `quantile_average` mode; tail-weighted aggregation is an optional variant). This construction makes the model-side term available for a single online PPO response.

Table 4: Compact downstream PPO pilot. Values are percentages averaged over three repeated judge runs, with standard deviations in parentheses. A is the base policy win rate and B is the PPO policy win rate.

PPO reward / budget	A win	B win	Tie	Net B-A
Center-only, 512 updates	42.6 (0.8)	43.0 (1.5)	14.5 (1.3)	+0.3 (2.1)
Full Data-U + Fisher-U, 512 updates	36.1 (2.7)	45.2 (3.7)	18.7 (1.3)	+9.1 (6.3)
Center-only, 1024 updates	35.6 (1.7)	50.2 (3.7)	14.2 (3.1)	+14.7 (4.8)
Full Data-U + Fisher-U, 1024 updates	24.5 (1.8)	57.6 (3.2)	17.9 (1.5)	+33.1 (5.0)

Before applying the penalties, each raw term is standardized by its own exponential running mean and variance over reward-server calls. Writing the normalized quantities as $\widetilde{u}_{\text{data}}$ and $\widetilde{u}_{\text{model}}$, the scalar reward used by PPO is

$$r_{\text{PPO}}(x, y) = c(x, y) - \lambda_w \widetilde{u}_{\text{data}}(x, y) - \lambda_u \widetilde{u}_{\text{model}}(x, y). \tag{5}$$

Here λ_w and λ_u are the data-side and model-side penalty weights, respectively. Setting both to zero recovers the center-only condition reported below.

The 1024-update full-reward continuation is the strongest downstream result in the current study, but it should be interpreted cautiously: the prompt split is small, the judge is automated, and the continuation changes both policy training duration and the effective actor initialization. We therefore do not use this pilot to claim that FQRM estimates calibrated uncertainty; it only shows that the reward representation and its optional shaping terms can be integrated into PPO.