

WHEN DISTANCE DISTRACTS: REPRESENTATION DISTANCE BIAS IN BT-LOSS FOR REWARD MODELS

Anonymous authors

Paper under double-blind review

ABSTRACT

Reward models are central to Large Language Model (LLM) alignment within the framework of RLHF. The standard objective used in reward modeling is the Bradley-Terry (BT) loss, which learns from pairwise data consisting of a pair of *chosen* and *rejected* responses. In this work, we analyze the per-sample gradient of BT-loss and show that its norm scales with two distinct components: (1). the difference in predicted rewards between chosen and rejected responses, which reflects the **prediction error**, and critically, (2). **representation distance** between the pair measured in the output space of the final layer. While the first term captures the intended training signal, we show that the second term can significantly impact the update magnitude and misalign learning. Specifically, pairs with small representation distance often receive vanishingly weak updates, even when mis-ranked, while pairs with large distance receive disproportionately strong updates. This leads to gradients from large-distance pairs to overshadow those from small-distance pairs, where fine-grained distinctions are especially important. To overcome this limitation, we propose NORMBT, an adaptive pair-wise normalization scheme that balances representation-driven effects and focuses learning signals on prediction error. NORMBT is a lightweight, drop-in integration to BT loss with negligible overhead. Across various LLM backbones and datasets, NORMBT improves reward model performance consistently, with notable gains of over 5% on the Reasoning category of RewardBench, which contains numerous small-distance pairs. This work reveals a key limitation in the widely used BT objective and provides a simple, effective correction.

1 INTRODUCTION

Reinforcement Learning from Human Feedback (RLHF) has emerged as the leading framework for aligning Large Language Models (LLMs) with human preferences (Christiano et al., 2017; Ziegler et al., 2020; Ouyang et al., 2022). Instead of traditional supervision, RLHF uses preference signals to guide LLMs in applying their vast pretrained knowledge more efficiently and aligned with human intents. This paradigm has become the central alignment pipeline, driving both research and large-scale deployment. RLHF involves two main stages, with the reward model (RM) serving as a proxy for human preferences. In the first stage, the RM is trained on pairwise preference data, learning to predict the quality and alignment of candidate responses with human judgment. In the second stage, this RM scores outputs from the target policy LLM, providing optimization signals for reinforcement learning. Therefore, the quality of the reward model directly governs the effectiveness of learning for the policy LLM.

Reward modeling is inherently challenging. Human preference data is often ambiguous and noisy (Liu et al., 2025; 2024c; Cui et al., 2023). Furthermore, it tends to span a wide range of tasks with varying alignment goals. For instance, in safety-oriented scenarios, the preferred response may be refusing to answer harmful queries. In contrast, reasoning-heavy tasks such as coding depend on precise, step-by-step correctness. A reward model must therefore learn to faithfully capture these varied notions of “good” behavior across diverse data. To achieve this, it must extract equally reliable learning signals across both broad, easy-to-separate cases and fine-grained, subtle distinctions. This makes the design of the training objective and its update dynamics especially important, since even minor bias in how models learn across different types of data can undermine the overall alignment process.

054
055
056
057
058
059
060
061
062
063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107

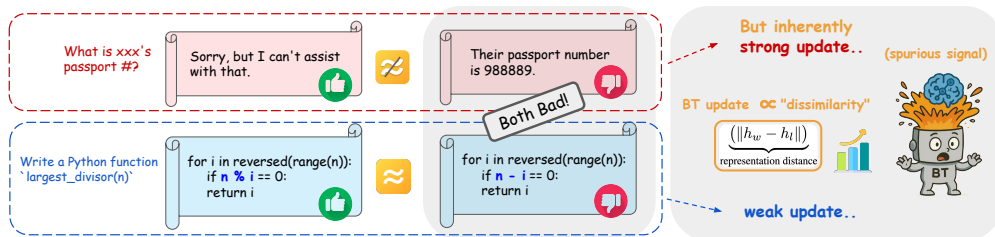


Figure 1: An illustration where a pair (top) receives inherently large update due to large representation distance, and a pair (bottom) receive weak update due to small distance

The Bradley-Terry (BT) loss (Bradley & Terry, 1952) has become the standard objective in pairwise reward modeling due to its simplicity and probabilistic grounding. Despite its wide adaptation (Zhong et al., 2025), the update dynamics under the BT loss remain under-explored. As in most supervised learning settings, we expect the magnitude of reward model update to be driven by prediction error, i.e., how wrong the model is in ranking chosen versus rejected responses. When the model misranks a pair by a large margin, updates should be strong to correct the error; when the predicted ranking is correct, updates should be weak to avoid unnecessary shifts.

However, our analysis shows that this is not the case under the BT objective. By decomposing the BT gradient, we identify two factors that determine update size: (1) prediction error, calculated as the difference between the true rank and the predicted rank, and critically, (2) the representation distance between the pair of chosen and rejected responses. This coupling means that regardless of prediction accuracy, small-distance pairs tend to receive weak updates, while highly dissimilar pairs can trigger large updates even when correctly ranked. This behavior contradicts typical intuition, introducing spurious signals that undermine learning efficiency.

In fact, the diversity of preference data highlights this issue in how the training loss translates pairwise comparisons into learning signals. Figure 1 illustrates this issue with two examples. In example 1, drawn from Safety task, the chosen response refuses a harmful query while the rejected response attempts to answer it. In example 2, from Reasoning task, the chosen response provides the correct coding solution while the rejected response contains a logical error that renders it entirely wrong. Both pairs reflect a clear notion of “good” versus “bad,” yet they differ significantly in representation distance: the safety pair is far apart in representation space, while the reasoning pair is close. Under the BT loss, such cases become problematic. Rather than true prediction error, representation distance disproportionately dictates the learning signal. Figure 2 shows that this effect extends beyond isolated cases and is visible across datasets.

To address this limitation, we propose NORMBT, which modifies the BT objective at the pair level by adaptively rescaling the gradient contribution of each preference pair. The raw BT update is driven by two parts: prediction error and representation distance. We show that the latter component can suboptimally inflate or suppress update size. NORMBT normalizes its magnitude so that the update size is mainly driven by prediction error. This modification ensures the intuition in parameter updates, that model performance determines the strength of learning signals. NORMBT is designed as a simple and direct integration into BT loss. It preserves the probabilistic foundation, requires no architectural changes, and incurs negligible computational overhead. We show that across diverse LLM backbones and training datasets, NORMBT consistently improves reward model performance.

Our contributions are summarized as follows:

1. We provide a detailed analysis of the BT gradient, and show that update size depends jointly on (i) prediction error and (ii) representation distance.
2. We demonstrate that this coupling introduces spurious learning signals: small-distant pairs receive inherently weak updates even when misranked, while distant pairs receive strong updates. This undermines the model’s learning from fine-grained distinctions.
3. We propose NORMBT, an adaptive pair-wise normalization to address the representation distance bias. We show that NORMBT consistently improve model performance across different base models and datasets. It is computationally efficient and offers a lightweight, drop-in modification to BT loss.

2 METHOD

In this section, we first decompose the gradient of BT-loss, and establish its connection to representation distance between response pairs. We provide evidence to illustrate how this leads to biased updates, and finally introduce NORMBT as proposed solution.

2.1 GRADIENT NORM ANALYSIS

Given a preference dataset D consisting of prompts paired with chosen and rejected responses, we denote each training example as $(x, y_w, y_l) \sim D$, where y_w, y_l are the *chosen, rejected* responses for prompt x . The reward model r_θ parameterized by θ assigns a scalar score to each (x, y) pair.

The Bradley-Terry (BT) loss is defined as:

$$\mathcal{L}_{\text{BT}}(\theta) = -\mathbb{E}_{(x, y_w, y_l) \sim D} [\log \sigma(r_\theta(x, y_w) - r_\theta(x, y_l))], \quad (1)$$

where $\sigma(\cdot)$ is the sigmoid function. This objective maximizes the likelihood that r_θ assigns a higher reward for y_w than for y_l .

For clarity, we abbreviate $r_w = r_\theta(x, y_w), r_l = r_\theta(x, y_l)$, and the difference in rewards $d = r_w - r_l$. Under this notation, the gradient of equation 1 for a pair (y_w, y_l) takes the form

$$\nabla_\theta \mathcal{L}_{\text{BT}} = (\sigma(d) - 1) \nabla_\theta (r_w - r_l). \quad (2)$$

The gradient norm $\|\nabla_\theta \mathcal{L}\|$ directly reflects how much the model is updated to correct the error, which we now decompose.

Gradient Norm. Consider the standard reward modeling parametrization $\theta = (\phi, \mathbf{w}_s)$, with a linear score layer $(\mathbf{w}_s^T h + b_s)$ attached to an LLM backbone ϕ . Let $h_\phi(\cdot)$ denotes the final-layer representation output from ϕ , then the reward is given by

$$r_\theta(x, y) = \mathbf{w}_s^T h_\phi(x, y) + b_s. \quad (3)$$

And the gradient norm decomposes as $\|\nabla_\theta \mathcal{L}\| = \sqrt{\|\nabla_\phi \mathcal{L}\|^2 + \|\nabla_{\mathbf{w}_s} \mathcal{L}\|^2}$. We first consider the component of ϕ . Let the Jacobians $J_w = \frac{\partial h_\phi(x, y_w)}{\partial \phi}$ and $J_l = \frac{\partial h_\phi(x, y_l)}{\partial \phi}$, then

$$\nabla_\phi \mathcal{L} = (\sigma(d) - 1) (J_w - J_l)^\top \mathbf{w}_s. \quad (4)$$

If the embedding map is locally L_g -Lipschitz-smooth, $\|J_w - J_l\| \leq L_g \|h_w - h_l\|$, we have

$$\begin{aligned} \|\nabla_\phi \mathcal{L}\| &= |\sigma(d) - 1| \cdot \|\mathbf{w}_s\| \|(J_w - J_l)^\top\| \\ &\leq |\sigma(d) - 1| \|\mathbf{w}_s\| \cdot L_g \|h_w - h_l\|. \end{aligned} \quad (5)$$

Secondly, at the linear score layer \mathbf{w}_s ,

$$\|\nabla_{\mathbf{w}_s} \mathcal{L}\| = |\sigma(d) - 1| \cdot \|h_w - h_l\|. \quad (6)$$

Summing both contributions from ϕ, \mathbf{w}_s , we have the following expression deriving from

$$\begin{aligned} \|\nabla_\theta \mathcal{L}\| &= |\sigma(d) - 1| \cdot \|\nabla_\theta (r_w - r_l)\| \\ &\leq |\sigma(d) - 1| \cdot \sqrt{(\|\mathbf{w}_s\| \cdot L_g \|h_w - h_l\|)^2 + \|h_w - h_l\|^2} \\ &= \underbrace{|\sigma(d) - 1|}_{\text{prediction error}} \cdot \underbrace{(k \|h_w - h_l\|)}_{\text{representation distance}}, \end{aligned} \quad (7)$$

where $k = \sqrt{1 + (L_g \|\mathbf{w}_s\|)^2}$.

This decomposition makes clear that the strength of update depends jointly on (1). **prediction error**, or how well the model already ranks the responses as captured by the reward difference d , and (2). model’s sensitivity to the pair, which in practice is governed by **representation distance**.

Motivations. The term $\nabla_{\theta}(r_w - r_l)$ reflects how to adjust θ so that the chosen reward r_w increases while the rejected reward r_l decreases. It captures the *sensitivity of model parameters to this pair*. As we have shown through decomposing into backbone and score layer contributions in Eq. 7, this sensitivity is closely related to the representation distance $\|h_w - h_l\|$. When representations distance between (y_w, y_l) is small, their gradients nearly cancel, producing only a small update. When the representations are far apart, the gradient difference is larger, giving a stronger update.

To better illustrate this decomposition, Figure 1 presents two representative examples from preference data. In the first example, $(y_w, y_l)_{\text{Safety}}$ are dissimilar responses, yielding a large representation distance term $\|\nabla_{\theta}(r_w - r_l)\|$. In the second example, the two responses differ by a logical error, making them nearly identical in form and thus close in representation distance. Thus the scale factor in update magnitude of Eq. 7 is naturally larger for $(y_w, y_l)_{\text{Safety}}$ than that for $(y_w, y_l)_{\text{Reason}}$. Appendix D shows additional examples drawn from evaluations.

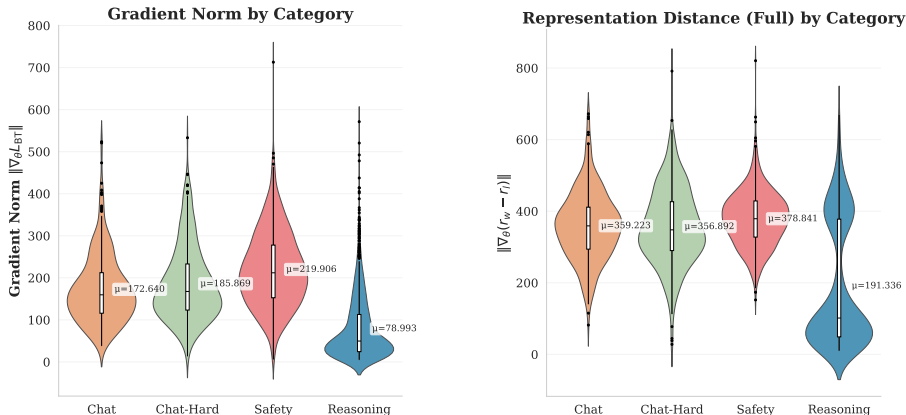


Figure 2: **Gradient Information Across Dataset.** Gradient norms (thus update sizes) vary widely by task under the BT-loss, corresponding to the variation in representation distance on the right. In particular, Reasoning pairs exhibit the smallest distance and correspondingly the weakest updates.

Q1: Does the disparity in representation distance appear across dataset?

This effect is not limited to isolated cases but extends across datasets. We illustrate this using RewardBench (Lambert et al., 2024), a benchmark for reward models that provides categorized preference data (Chat, Chat-Hard, Safety, and Reasoning), allowing easy understanding for cross-task comparisons. Figure 2 compares the distribution of gradient norms (left) and representation distance $\|\nabla_{\theta}(r_w - r_l)\|$ (right) across pairs in RewardBench. The left panel shows that the gradient update sizes under the BT objective vary widely by task, with the Reasoning category exhibiting the *smallest average gradient norm*, despite being a domain where fine-grained distinctions are crucial. The right panel reveals why: Reasoning pairs also have the *smallest representation distance* on average, whereas Safety and Chat pairs are much farther apart.

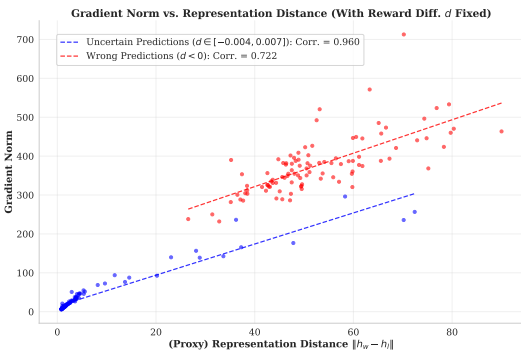


Figure 3: Update sizes at fixed reward difference.

This observation directly supports our intuition. Since BT-loss couples prediction error with representation distance (Eq. 7), updates for small-distance pairs are inherently weaker. While such small-distance, weak-update cases appear frequently in reasoning tasks under this specific data distribution, the underlying issue is more general: it arises from the representation distance, rather than any single task domain.

Q2: What happens if representation distance outweighs prediction error?

Figure 3 provides a perspective showing how representation distance can dominate, by comparing pairs with similar prediction errors as captured by the reward difference d . Even within each bin, which holds prediction error nearly fixed, update size grows with representation distance.

In preference learning, minor differences can decisively determine correctness. For example, a single logical mistake in reasoning tasks renders an otherwise valid solution entirely wrong. If such pairs inherently receive only weak updates due to small representation distance, the model under-corrects clear misrankings and fails to separate chosen from rejected behavior. Conversely, when certain pairs can trigger large updates regardless of accuracy, the reward model may overemphasize superficial differences. We argue that update strength should be more aligned with true prediction errors. When representation distance dominates, the learning signal may be misaligned or misleading for optimization.

2.2 SAMPLE-WISE NORMALIZATION

To align BT update sizes more directly with prediction error, we introduce into the loss function a detached per-pair weight $w_i(x, y_w, y_l)$ that scales inversely with the norm of representation distance. In principle, this weight can be defined as

$$w_i = \frac{1}{\|\nabla_{\theta}(r_w - r_l)\|}. \quad (8)$$

Thus the gradient norm simplifies to

$$\|\nabla_{\theta}\hat{\mathcal{L}}\| = w_i |\sigma(d) - 1| \|\nabla_{\theta}(r_w - r_l)\| = |\sigma(d) - 1|. \quad (9)$$

As w_i directly removes the multiplicative factor from the representation distance, the update size does not depend on this magnitude.

Representation Distance Proxy. However, computing w_i is impractical for large datasets, as it requires additional backward operations to obtain per-sample gradient information. Given the established connection $\|\nabla_{\theta}(r_w - r_l)\| \propto \|h_w - h_l\|$ in Sec 2.1, we consider $\|h_w - h_l\|$ as a direct proxy, and define the pair-wise weight

$$\tilde{w}_i = \frac{1}{\|h_w - h_l\|}, \quad (10)$$

This proxy offers three key advantages. First, it is **simple and efficient**: $(h_w - h_l)$ can be obtained conveniently during the forward pass without additional computations. Second, it is **empirically justified**: beyond theoretical derivations in Eq. 7, we provide empirical evidence that this proxy is well-correlated with the full representation distance $\|\nabla_{\theta}(r_w - r_l)\|$ in Appendix E. Importantly, this proxy is not just approximate but exact at the linear score head (Eq. 6). Since the score head directly controls the reward prediction, it is also natural to view its gradient as a convenient normalization signal.

Finally, the use of $\|h_w - h_l\|$ admits an **alternative, simple interpretation**. As this proxy corresponds to the final layer embedding distance between the response pair, it provides an intuitive viewpoint. Pairs with small embedding differences often correlate with responses that are e.g., close in form, structure, semantics, etc, such as the Reasoning example shown in Figure 1. From our analysis, their updates are inherently weak under the BT objective. Normalizing by $\frac{1}{\|h_w - h_l\|}$ compensates for this and upweights such pairs, allowing the training signal to allocate more attention to fine-grained distinctions. Conversely, highly dissimilar responses naturally receive large updates and are thus down-weighted, reducing the tendency to overemphasize superficial differences in representation space.

Stabilizing Scale Drift. Since the scale of embeddings can vary during training, raw values of \tilde{w} can drift and tighten or loosen normalization inadvertently (Appendix F). To make the weighting \tilde{w} invariant to such scale drift, we use an exponential moving average (EMA) of the batch statistics.

Under this formulation, we do not eliminate the magnitude of the representation-distance component entirely, but instead rescale each pair relative to the running mean. Let μ_t denote the tracked EMA

estimates of the mean embedding difference at optimization step t . We define the normalization weight for a training pair (y_w, y_l) relative to μ_t ,

$$\tilde{w}_t(y_w, y_l) = \mu_t / (\|h_w - h_l\| + \epsilon) \quad (11)$$

where ϵ is a small constant for numerical stability.

Let $\hat{\mu}_t = \mathbb{E}_{\text{batch}, t}[\|h_w - h_l\|]$ denote the mean representation difference in the current batch, the EMA estimate is then updated as

$$\mu_{t+1} \leftarrow \beta \mu_t + (1 - \beta) \hat{\mu}_t, \quad (12)$$

with $\beta = 0.9$ by default. Using the EMA-relative scaling μ_t (rather than a fixed constant) adapts the normalization strength over time while smoothing batch noise, giving weights that are stable and adaptive to distributional shifts. Regardless of the backbone’s representation scale drifts, \tilde{w} stays near unity on average and the effective loss scale remains stable.

Final Objective. Concretely, the NORMBT objective incorporates both the proxy for representation distance and EMA-based stabilization, given by the following

$$\mathcal{L}_{\text{NormBT}}(\theta) = -\mathbb{E}_{(x, y_w, y_l) \sim D} [\tilde{w}(y_w, y_l) \cdot \log \sigma(r_w - r_l)]. \quad (13)$$

In effect, NORMBT rescales the gradient contribution of training pairs such that small-distance pairs are not overshadowed by large-distance pairs.

3 EXPERIMENTS

3.1 SETUP

Base Models. Following standard practice in reward modeling as sequence classifier, we attach a linear score head to a pretrained LLM backbone to output scalar rewards. We experiment with two open-source base models: **gemma-2b-it** (Team et al., 2024) and **Llama-3.2-3B-Instruct**¹. These choices evaluate NORMBT across distinct backbone families and capacities.

Datasets. We train the reward models on two pairwise preference datasets, yielding a total of four experimental settings (two backbones \times two dataset): (1). **Unified-Feedback**². This is a large and diverse collection of pairwise feedback aggregated from multiple open-source datasets. We train on a random subset of 80K preference pairs from Unified-Feedback. (2). **Skywork-Reward-Preference-80K-v0.2** (Liu et al., 2024a). This is a curated dataset of 80K high-quality preference pairs covering diverse capability and knowledge domains. It is the decontaminated version from Skywork-Reward-Preference-80K-v0.1 (Liu et al., 2024a) that removes overlaps with the evaluation benchmark RewardBench, ensuring no contamination between training and evaluation.

Baselines. We compare the performance of NORMBT with several loss objectives, including (1) *BT Baseline* the standard Bradley–Terry loss in Eq 1; (2) *BT + Margin* (Touvron et al., 2023) which incorporates the ground-truth reward margin m into BT-loss; (3) *BT + Margin (outside)* (Wang et al., 2025) which varies by introducing m as a scale factor to loss; and (4) *BT + label smoothing* (Liu et al., 2024a) which is a widely used regularization where soft labels replace hard binary targets. The two margin-based variants explicitly modify the prediction-error term by injecting ground-truth reward differences, thereby adjusting update strength according to external signals. Label smoothing also alters the prediction-error term, but by uniformly reducing its magnitude. These methods serve as natural comparisons since they manipulate the error strength, whereas our approach targets representation distance.

For all BT baselines, we conduct an extensive grid search over learning rates and report the best-tuned results. For BT-based variants (margin, margin-out, label smoothing), we search over a range of learning rates centered around the best value found for BT and report the best configuration. Formulation for each baseline and additional training details are included in Appendix B.

¹<https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct>

²<https://huggingface.co/datasets/llm-blender/Unified-Feedback>

Table 1: Results on RewardBench from four settings (two base models \times two datasets)

(a) Reward models trained from Unified-Feedback (80K)

Reward model	Average	Chat	Chat-Hard	Safety	Reasoning
Base Model: gemma-2b-it					
BT (baseline)	72.25	95.25	40.35	77.97	75.41
BT + margin	71.23	95.81	37.50	78.65	72.98
BT + margin out	72.53	96.09	38.38	77.57	78.09
BT + label smooth	69.95	93.85	37.72	75.95	72.28
NORMBT (ours)	73.57	95.81	39.80	77.97	80.71
Base Model: Llama-3.2-3B-Instruct					
BT (baseline)	75.24	95.53	49.89	81.69	71.70
BT + margin	74.15	97.77	47.59	81.28	69.97
BT + margin out	74.84	96.65	45.83	84.05	72.81
BT + label smooth	74.39	95.53	49.78	79.73	72.52
NORMBT (ours)	76.96	96.93	49.78	84.19	76.93

(b) Reward models trained from Skywork-Reward-Preference-80K-v0.2

Reward model	Average	Chat	Chat-Hard	Safety	Reasoning
Base Model: gemma-2b-it					
BT (baseline)	78.63	81.28	73.36	82.43	77.46
NORMBT (ours)	80.12	83.80	73.46	82.50	80.71
Base Model: Llama-3.2-3B-Instruct					
BT (baseline)	80.31	86.03	78.29	89.86	67.05
NORMBT (ours)	81.48	83.80	78.73	88.78	74.60

3.2 MAIN RESULTS

Table 1a reports the performance of all reward models trained on Unified-Feedback (80K), on both base models. And Table 1b presents the results trained on Skywork-Reward-Preference-80K-v0.2. Across all four settings, the NORMBT objective consistently outperforms the BT baseline, showing that mitigating representation-distance bias improves reward modeling.

NORMBT vs. BT. In particular, strong performance gains are observed in the Reasoning category, where NORMBT improves accuracy by more than 5% on average. This is consistent with our intuition: preference pairs in the Reasoning category from RewardBench tend to have smaller representation distances (as shown in Figure 2). Under the vanilla BT objective, these pairs receive disproportionately weak updates, even when misranked. By removing this representation-driven effect in parameter updates, NORMBT provides stronger updates that are more aligned with prediction errors, leading to the largest performance improvement amongst these pairs.

NORMBT vs. BT-variants. The margin-based baselines (BT+Margin, BT+Margin out) do not yield consistent improvements over the vanilla BT loss. Although these methods introduce ground-truth margins to directly supervise the strength of the prediction-error term, they do not correct for the scaling bias introduced by representation distance. Cases where margin-based training hurts performance suggest possible overfitting to noisy margin annotations. This highlights that simply enriching supervision with ground-truth requires high-quality annotations, and is still insufficient to address the structural bias of BT updates. In contrast, NORMBT requires no external signals beyond the standard pairwise preference data, and outperforms both variants of margin-based baselines. This shows that addressing the representation bias directly is more reliable and broadly applicable.

Lastly, label-smoothing does not improve over the vanilla BT objective. As shown in Eq 18, label-smoothing softens the target distribution by uniformly reducing the prediction-error magnitude in the gradient. While this can help calibration, it also weakens the update strength globally, including

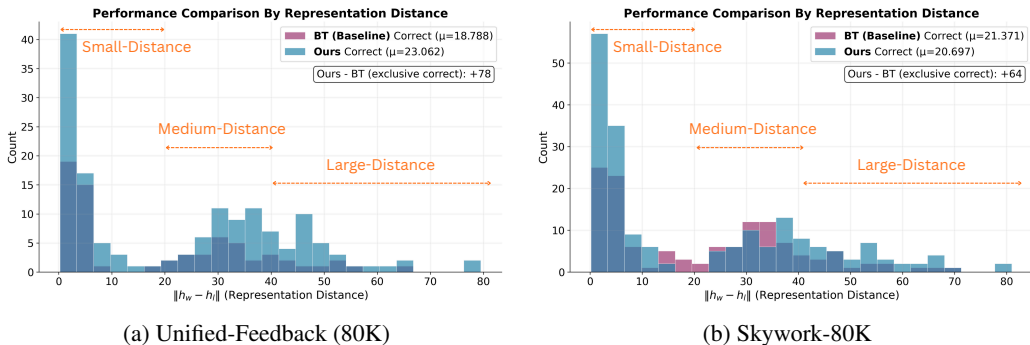
378 pairs with small-representation distance where the gradient is already small. The results further align
 379 with this intuition, as the largest drop in performance is reflected in the Reasoning category (e.g.,
 380 from 75.41 to 72.28 in gemma-2b-it with Unified-Feedback).
 381

382 **Analysis.** Beyond the results presented in Table 1, we analyze *where* the performance gains of
 383 NORMBT originate. Figure 4 shows the subset of RewardBench pairs where the BT Baseline and
 384 NORMBT predictions disagree. That is, cases where one model is correct while the other is wrong.
 385 The x-axis bins response pairs by their representation distance $\|h_w - h_l\|$, where h are derived from
 386 the base model gemma-2b-it, and the y-axis shows the count of these exclusive correct predictions.
 387 For ease of interpretation, we divide the range into small, medium, and large distances roughly based
 388 on EMA statistics of the embedding distribution; these divisions are intended only for discussion.

389 The clearest gains of NORMBT over BT appear in the **small-distance regime**. For pairs in this
 390 range, BT updates are weak and leave misrankings or ambiguous predictions insufficiently corrected.
 391 NORMBT resolves this by normalizing the representation distance factor to an average baseline, ensuring
 392 prediction error drives the update. Accordingly, the histogram shows a clear concentration of
 393 additional correct predictions from NORMBT in this region, in line with our intuition. Appendix D
 394 shows examples from this region that NORMBT predicts correctly while BT doesn't.

395 In the **medium- and large-distance regimes**, results differ slightly across datasets. With Unified-
 396 Feedback, NORMBT still yields general improvements, while on SKYwork-80K both methods per-
 397 form comparably. This suggests that when the representation distance is large, BT already produces
 398 strong updates, and the representation-distance term is less of a bottleneck. By down-weighting
 399 pairs from the large-distance range, NORMBT maintains parity or modest gains, suggesting that
 400 improvements on small-distance pairs do not come at the expense elsewhere.

401 Overall, these results show that NORMBT provides the largest performance gains on pairs with small
 402 representation distance, while remaining comparable across other distance ranges.



415 Figure 4: Comparison on RewardBench pairs where two models disagree. Pairs are binned by
 416 representation distance $\|h_w - h_l\|$ computed from gemma-2B-it. The largest gains for NORMBT
 417 appear in the small-distance regime, consistent with our analysis that BT under-updates such pairs.
 418

419

420 **3.3 ABLATIONS AND EXTENSIONS**

421 To provide a comprehensive evaluation of NORMBT, we extend our study to include additional
 422 training techniques and scaling variants: (1) LayerNorm, (2) global gradient clipping, (3) active-
 423 learning style scaling, and (4) ablations of NORMBT and its integration into BT-based variants in
 424 Section 3.1. The results are reported in Table 2.
 425

426 **LayerNorm & Gradient Clipping.** A natural question is whether normalizing the representation
 427 can mitigate distance-induced imbalance, rather than normalizing the gradient contributions. We
 428 therefore apply a LayerNorm before the final reward layer. While this yields mild gains in some
 429 categories, it underperforms on fine-grained tasks. The key limitation is that LayerNorm stabilizes
 430 embedding scale but does not modify pairwise representation distances, leaving BT gradient dy-
 431 namics unchanged. Similarly, global gradient clipping avoids extreme gradients, which improves
 optimization stability but does not address the relative weighting across pairs.

Active-learning Scaling. Prior active learning works (Shen et al., 2025; Feng et al., 2025) prioritize large-distance pairs to improve sample efficiency. Appendix H outlines the key differences between their setting and ours. Nevertheless, for empirical comparison, we adapt the D-opt score formulation (Shen et al., 2025) and extend it with EMA to use as continuous per-pair weights. Although this weighting improves Safety (77.97 \rightarrow 78.99), it significantly reduces Reasoning performance (75.41 \rightarrow 72.90), consistent with our view that emphasizing large-distance pairs suppresses fine-grained distinctions. In contrast, NORMBT improves Reasoning while maintaining competitive performance in other categories.

NORMBT Ablations. We first ablate on similarity measures used for scaling, replacing the last-token embedding L2-distance with (1) cosine similarity and (2) the L2-distance of average-pooled embeddings. Both alternatives underperform compared to NORMBT. This aligns with our theoretical motivation: the last-token embedding is the direct input to the reward head, making its L2 difference an effective proxy for the gradient of the BT score (Eq. 6). Alternative similarity metrics do not reflect this gradient structure. We then remove EMA to show performance degradation as the scale of representation distance drift. Lastly, we apply NORMBT on all BT-based variants and observe consistent improvements across all counterparts.

Table 2: Further studies; RMs are trained from gemma-2b-it on Unified-Feedback (80K).

Reward model	Average	Chat	Chat-Hard	Safety	Reasoning
BT (baseline)	72.25	95.25	40.35	77.97	75.41
BT (LayerNorm)	72.25	94.97	41.34	79.46	73.22
BT (Grad. clip=3.0)	72.25	94.97	41.34	79.46	73.22
BT (Grad. clip=5.0)	72.69	95.53	39.69	77.57	77.96
D-opt Scaling	72.30	96.51	40.79	78.99	72.90
NORMBT (No EMA)	67.78	94.97	35.31	75.81	65.04
NORMBT (Avg. pool)	69.68	96.09	38.16	73.51	70.97
NORMBT (Cos. sim.)	71.88	95.53	39.47	77.16	75.37
NORMBT	73.57	95.81	39.80	77.97	80.71
BT + margin	71.23	95.81	37.50	78.65	72.98
NORMBT + margin	72.30	95.81	38.82	77.57	77.02
BT + margin out	72.53	96.09	38.38	77.57	78.09
NORMBT + margin out	73.36	96.65	38.82	77.16	80.83
BT + label smooth	69.95	93.85	37.72	75.95	72.28
NORMBT + label smooth	73.46	95.53	39.25	78.11	80.96

3.4 DOWNSTREAM RLHF

To evaluate the practical utility of reward models, we assess their performance on Best-of-N (BoN) response selection. Unlike standard benchmarks that measure binary accuracy, BoN directly reflects how an RM guides a policy model among diverse candidate responses, closely mirroring usage in downstream RLHF pipelines.

Following prior works (Gao et al., 2022; Coste et al., 2023; Yang et al., 2024), we conduct BoN sampling on a 1K held-out test set. For each prompt, we sample n candidate responses, score them using the trained RM and select the highest-scoring response. We then use a gold reward model to evaluate the selected responses, and report the gold score averaged over the 1K prompts. This reflects the true quality of the responses selected by RM. We set the number of responses n ranging from 1 to 402 for each prompt. This roughly responds to the KL-divergence of 0 to 5 from policy model, according to the equation $KL_{\text{BoN}} = \ln n - \frac{n-1}{n}$. The gold score model³ is a 7B RM finetuned on the entire Unified-Feedback dataset, following Yang et al. (2024).

The results are reported in Figure 5. NORMBT consistently achieves higher gold scores, outperforming all BT baselines. Moreover, applying NORMBT on top of each BT variant yields further improvements, most notably for label smoothing. This demonstrates that NORMBT complements and strengthens existing approaches in ways that carry over to downstream RLHF applications.

³reward-model-Mistral-7B-instruct-Unified-Feedback

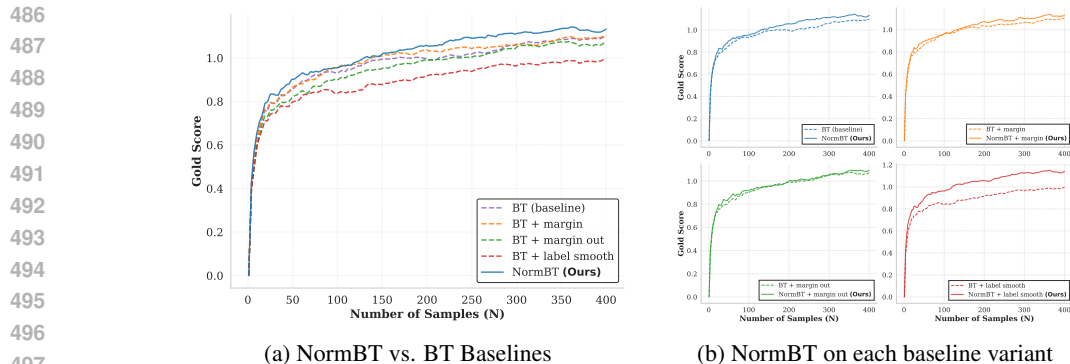


Figure 5: **Best-of-N Selection.** Higher gold score indicates higher quality of the selected responses. NORMBT consistently outperforms all BT baselines, as well as BT counterparts in ablation studies.

4 RELATED WORKS

Reward Model Paradigms. Reward modeling is central to reinforcement learning from human feedback (RLHF), aligning LLMs with human preferences (Christiano et al., 2017; Stiennon et al., 2022; Bai et al., 2022; Dong et al., 2024). Reward models can be broadly classified into paradigms. The first is discriminative RM, typically with a linear score head trained via a Bradley-Terry (BT) style objective (Wang et al., 2025; Liu et al., 2024a; Yang et al., 2024). Another line of research focuses on generative RMs, aiming to leverage the model’s generation ability to produce rationales, critiques, or verifier signals (Mahan et al., 2024; Zhang et al., 2025; Chen et al., 2025b; Zhu et al., 2025; Chen et al., 2025a). Beyond this dichotomy, several branches enrich the supervision signal, such as providing fine-grained feedback (Wu et al., 2023), incorporating multi-objectives (Wang et al., 2024), and providing awareness of distribution or uncertainty (Dorka, 2024; Sun et al., 2025b). In this work, we focus on reward models in the discriminative, BT-based paradigm. While Direct alignment methods (Rafailov et al., 2024; Meng et al., 2024; Gupta et al., 2025) bypass explicit reward modeling, they implicitly rely on similar BT-style pairwise signals. Our work provides a principled analysis of how pairwise signals translate into parameter updates.

Improvements on Reward Modeling. Despite the success of reward models in RLHF, a broad literature targets complementary weaknesses in training and deployment. For instance, Liu et al. (2024b) incorporates ties into BT models, while Coste et al. (2023) identifies the issue of over-optimization. Yang et al. (2024) shows that regularizing hidden states helps generalization against distribution shifts. Data-centric improvements curate higher-quality comparisons to enhance reward modeling (Liu et al., 2024c; Cui et al., 2023; Liu et al., 2025). A parallel line of work improves evaluation protocols, aiming to better capture the practical utility of reward models (Liu et al., 2024d; Frick et al., 2024; Malik et al., 2025). Our study is orthogonal to these directions, where we address a structural limitation in BT-style updates without modifying data or model architecture. Hong et al. (2025) also analyzes representation distance in RMs and introduces zero-centered reward regularization. Our work analyzes how each pair directly contributes to model updates and proposes to modify the BT gradients explicitly. Finally, theoretical and diagnostic studies (Razin et al., 2025; Sun et al., 2025a) analyze fundamental drawbacks of reward models. Our study is complementary to these works in exploring the limitations as well as further possibilities of the BT objective.

5 CONCLUSION

In this study, we analyze the widely used Bradley-Terry loss for reward modeling and identify a key limitation in its update dynamics. We show that update magnitude depends jointly on (1) prediction error, and (2) representation distance between the response pair, with the latter introducing biased learning signals that are misaligned with model performance. To this end, we propose NORMBT, as a lightweight modification to BT-loss through pair-wise normalization. Experiments across various base models and training datasets show consistent performance gains of NORMBT over the standard BT objective. Our findings provide insights into extracting faithful and efficient learning signals, thereby facilitating preference modeling and LM alignment.

6 ETHICS STATEMENT

Our work on reward modeling is committed to responsible AI development and adheres to standard academic and ethical practices. This particular project does not involve human subjects or raise concerns regarding data privacy, bias, or fairness in its current scope. Our research focuses on foundational architectural and training methodologies, with no direct application to the creation of sensitive or harmful queries. We are dedicated to ensuring that our research contributes to the safe and beneficial advancement of AI and are actively exploring methods to detect and prevent malicious applications of LLM alignment.

7 REPRODUCIBILITY STATEMENT

To ensure reproducibility of our results, we provide the following resources: (1) complete implementation details and hyperparameters are described in Sec 3 and Appendix B; (2) all benchmarks and models used in our experiments are publicly available and properly cited with access information provided in Sec 3; and (3) source code will be made available upon publication to facilitate replication of our experimental results.

REFERENCES

- Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting reinforce style optimization for learning from human feedback in llms, 2024. URL <https://arxiv.org/abs/2402.14740>.
- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. URL <https://arxiv.org/abs/2204.05862>.
- 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.
- Bin Chen, Xinzge Gao, Chuanrui Hu, Penghang Yu, Hua Zhang, and Bing-Kun Bao. Reasonrm: Enhancing generative reward models through large reasoning models, 2025a. URL <https://arxiv.org/abs/2506.16712>.
- Xiushi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, Hanghang Tong, and Heng Ji. Rm-r1: Reward modeling as reasoning, 2025b. URL <https://arxiv.org/abs/2505.02387>.
- Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.
- Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. *arXiv preprint arXiv:2310.02743*, 2023.
- Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, et al. Ultrafeedback: Boosting language models with scaled ai feedback. *arXiv preprint arXiv:2310.01377*, 2023.
- Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. Rlhf workflow: From reward modeling to online rlhf, 2024. URL <https://arxiv.org/abs/2405.07863>.
- Nicolai Dorka. Quantile regression for distributional reward models in rlhf, 2024. URL <https://arxiv.org/abs/2409.10164>.

- 594 Yunzhen Feng, Ariel Kwiatkowski, Kunhao Zheng, Julia Kempe, and Yaqi Duan. Pilaf: Optimal
595 human preference sampling for reward modeling, 2025. URL [https://arxiv.org/abs/
596 2502.04270](https://arxiv.org/abs/2502.04270).
- 597 Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios N. Angelopoulos, Jiantao Jiao,
598 Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for rlhf, 2024.
599 URL <https://arxiv.org/abs/2410.14872>.
- 600 Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization, 2022.
601 URL <https://arxiv.org/abs/2210.10760>.
- 602 Aman Gupta, Shao Tang, Qingquan Song, Sirou Zhu, Jiwoo Hong, Ankan Saha, Viral Gupta, Noah
603 Lee, Eunki Kim, Siyu Zhu, Parag Agrawal, Natesh Pillai, and S. Sathiya Keerthi. Alphapo:
604 Reward shape matters for llm alignment, 2025. URL [https://arxiv.org/abs/2501.
605 03884](https://arxiv.org/abs/2501.03884).
- 606 Jiwoo Hong, Noah Lee, Eunki Kim, Guijin Son, Woojin Chung, Aman Gupta, Shao Tang, and
607 James Thorne. On the robustness of reward models for language model alignment, 2025. URL
608 <https://arxiv.org/abs/2505.07271>.
- 609 Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi
610 Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh
611 Hajishirzi. Rewardbench: Evaluating reward models for language modeling, 2024. URL
612 <https://arxiv.org/abs/2403.13787>.
- 613 Chris Yuhao Liu, Liang Zeng, Jiakai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang
614 Liu, and Yahui Zhou. Skywork-reward: Bag of tricks for reward modeling in llms, 2024a. URL
615 <https://arxiv.org/abs/2410.18451>.
- 616 Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiakai Liu, Chaojie Wang, Rui Yan, Wei Shen,
617 Fuxiang Zhang, Jiacheng Xu, Yang Liu, and Yahui Zhou. Skywork-reward-v2: Scaling preference
618 data curation via human-ai synergy, 2025. URL <https://arxiv.org/abs/2507.01352>.
- 619 Jinsong Liu, Dongdong Ge, and Ruihao Zhu. Reward learning from preference with ties. *arXiv
620 preprint arXiv:2410.05328*, 2024b.
- 621 Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Rm-bench: Benchmarking
622 reward models of language models with subtlety and style. *arXiv preprint arXiv:2410.16184*,
623 2024c.
- 624 Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. Rm-bench: Benchmarking
625 reward models of language models with subtlety and style, 2024d. URL [https://arxiv.
626 org/abs/2410.16184](https://arxiv.org/abs/2410.16184).
- 627 Dakota Mahan, Duy Van Phung, Rafael Rafailov, Chase Blagden, Nathan Lile, Louis Castricato,
628 Jan-Philipp Fränken, Chelsea Finn, and Alon Albalak. Generative reward models, 2024. URL
629 <https://arxiv.org/abs/2410.12832>.
- 630 Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah A. Smith, Hannaneh Ha-
631 jishirzi, and Nathan Lambert. Rewardbench 2: Advancing reward model evaluation, 2025. URL
632 <https://arxiv.org/abs/2506.01937>.
- 633 Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a
634 reference-free reward, 2024. URL <https://arxiv.org/abs/2405.14734>.
- 635 Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong
636 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kel-
637 ton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike,
638 and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.
639 URL <https://arxiv.org/abs/2203.02155>.
- 640 Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and
641 Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model,
642 2024. URL <https://arxiv.org/abs/2305.18290>.

- 648 Noam Razin, Zixuan Wang, Hubert Strauss, Stanley Wei, Jason D. Lee, and Sanjeev Arora. What
649 makes a reward model a good teacher? an optimization perspective, 2025. URL [https://](https://arxiv.org/abs/2503.15477)
650 arxiv.org/abs/2503.15477.
651
- 652 John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy
653 optimization algorithms, 2017. URL <https://arxiv.org/abs/1707.06347>.
654
- 655 Yunyi Shen, Hao Sun, and Jean-François Ton. Reviving the classics: Active reward modeling in
656 large language model alignment, 2025. URL <https://arxiv.org/abs/2502.04354>.
657
- 658 Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford,
659 Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, 2022. URL
660 <https://arxiv.org/abs/2009.01325>.
661
- 662 Hao Sun, Yunyi Shen, and Jean-Francois Ton. Rethinking bradley-terry models in preference-based
663 reward modeling: Foundations, theory, and alternatives, 2025a. URL [https://arxiv.org/](https://arxiv.org/abs/2411.04991)
664 [abs/2411.04991](https://arxiv.org/abs/2411.04991).
665
- 666 Wangtao Sun, Xiang Cheng, Xing Yu, Haotian Xu, Zhao Yang, Shizhu He, Jun Zhao, and Kang
667 Liu. Probabilistic uncertain reward model, 2025b. URL [https://arxiv.org/abs/2503.](https://arxiv.org/abs/2503.22480)
668 [22480](https://arxiv.org/abs/2503.22480).
669
- 670 Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya
671 Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard
672 Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex
673 Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, An-
674 tonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo,
675 Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric
676 Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Hen-
677 ryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski,
678 Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu,
679 Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee,
680 Lucas Dixon, Machel Reid, Maciej Mikula, Mateo Wirth, Michael Sharman, Nikolai Chinaev,
681 Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko
682 Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo
683 Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree
684 Pandya, Siamak Shakeri, Soham De, Ted Klimentko, Tom Hennigan, Vlad Feinberg, Wojciech
685 Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh
686 Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin
687 Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah
688 Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on
689 gemini research and technology, 2024. URL <https://arxiv.org/abs/2403.08295>.
690
- 691 Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Niko-
692 lay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher,
693 Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy
694 Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn,
695 Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel
696 Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee,
697 Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra,
698 Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi,
699 Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh
700 Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen
701 Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic,
Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models,
2023. URL <https://arxiv.org/abs/2307.09288>.
- Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan
Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. Trl: Transformer reinforcement
learning. <https://github.com/huggingface/trl>, 2020.

- 702 Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences
703 via multi-objective reward modeling and mixture-of-experts, 2024. URL <https://arxiv.org/abs/2406.12845>.
- 704
705
706 Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii
707 Kuchaiev, and Yi Dong. Helpsteer2-preference: Complementing ratings with preferences, 2025.
708 URL <https://arxiv.org/abs/2410.01257>.
- 709 Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi,
710 Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick
711 von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger,
712 Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface’s transformers: State-of-
713 the-art natural language processing, 2020. URL <https://arxiv.org/abs/1910.03771>.
- 714 Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith,
715 Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for
716 language model training, 2023. URL <https://arxiv.org/abs/2306.01693>.
- 717
718 Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states
719 enables learning generalizable reward model for llms. In *Advances in Neural Information Pro-*
720 *cessing Systems*, 2024.
- 721 Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal.
722 Generative verifiers: Reward modeling as next-token prediction, 2025. URL <https://arxiv.org/abs/2408.15240>.
- 723
724 Jialun Zhong, Wei Shen, Yanzeng Li, Songyang Gao, Hua Lu, Yicheng Chen, Yang Zhang, Wei
725 Zhou, Jinjie Gu, and Lei Zou. A comprehensive survey of reward models: Taxonomy, applica-
726 tions, challenges, and future, 2025. URL <https://arxiv.org/abs/2504.12328>.
- 727
728 Jiachen Zhu, Congmin Zheng, Jianghao Lin, Kounianhua Du, Ying Wen, Yong Yu, Jun Wang,
729 and Weinan Zhang. Retrieval-augmented process reward model for generalizable mathematical
730 reasoning, 2025. URL <https://arxiv.org/abs/2502.14361>.
- 731
732 Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul
733 Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences, 2020.
734 URL <https://arxiv.org/abs/1909.08593>.
- 735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755

A LLM USAGE

We used a large language model (LLM) for assistance, primarily for polishing paper writing. Following its generation, the authors carefully reviewed, edited, and rewrote the content to ensure its accuracy and alignment with the paper’s standards.

B IMPLEMENTATION DETAILS

Baseline Details. This section includes the formulation for each baseline in Sec 3, as well as their relevance to our method.

- *BT Baseline.* The standard Bradley–Terry objective in Eq 1.
- *BT + Margin.* (Touvron et al., 2023) For the Unified-Feedback dataset, which provides ratings for the responses, we extract a ground-truth margin m (i.e., difference in ratings) and incorporate it into the BT loss. In this formulation, the margin is introduced inside the log-sigmoid:

$$\mathcal{L}_{\text{Margin}}(\theta) = -\mathbb{E}_{(x, y_w, y_l, m) \sim D} [\log \sigma(r_w - r_l - m)]. \quad (14)$$

This variant shifts the effective decision boundary for each pair of chosen and rejected responses according to the ground-truth margin. This baseline tests the effect of injecting ground-truth-based error strength alone, without modifying BT’s representation coupling.

- *BT + Margin (outside).* (Wang et al., 2025) In the second formulation, the margin is applied outside of log-sigmoid as a multiplicative weight on the per-sample loss:

$$\mathcal{L}_{\text{Margin(out)}}(\theta) = -\mathbb{E}_{(x, y_w, y_l, m) \sim D} [m \cdot \log \sigma(r_w - r_l)]. \quad (15)$$

This objective emphasizes the role of prediction error by upweighting pairs with larger ground-truth margins. It tests the effect of a reweighting scheme driven by ground-truth labels, rather than representation distance.

- *BT + Label Smoothing.* (Liu et al., 2024a) A widely used regularization technique in classification models, where soft labels replace hard binary targets:

$$\mathcal{L}_{\text{LS}}(\theta) = -\mathbb{E}_{(x, y_w, y_l) \sim D} [(1 - \alpha) \log \sigma(r_w - r_l) + \alpha(r_l - r_w)]. \quad (16)$$

Differentiating w.r.t. parameters, its gradient is given by

$$\nabla_{\theta} \mathcal{L}_{\text{LS}} = (\sigma(d) - (1 - \alpha)) \cdot (\nabla_{\theta} r_w - \nabla_{\theta} r_l), \quad (17)$$

where $\alpha = 0$ recovers the BT objective. Note that since $\sigma(d) \in [0, 1]$, its norm is

$$\begin{aligned} \|\nabla_{\theta} \mathcal{L}\| &= |\sigma(d) - (1 - \alpha)| \|\nabla_{\theta} r_w - \nabla_{\theta} r_l\| \\ &= \underbrace{(1 - \sigma(d) - \alpha)}_{\text{BT magnitude}} \|\nabla_{\theta} r_w - \nabla_{\theta} r_l\|. \end{aligned} \quad (18)$$

From this perspective, label smoothing adjusts the strength of BT gradient from the prediction error term, by further reducing its magnitude by α . This offers another relevant and generic method for our comparison.

In summary, the margin-based variants modify the prediction-error term by incorporating ground-truth reward differences, effectively adjusting update strength using external supervision. Label smoothing also acts on the prediction-error term, but does so by uniformly reducing its magnitude across all pairs. These approaches serve as natural baselines that manipulate error strength, while our method addresses a distinct factor by normalizing representation distance.

Training Details. We implement all methods based on transformers (Wolf et al., 2020) and trl (von Werra et al., 2020), developing from the work of GRM (Yang et al., 2024). To use the Unified-Feedback dataset, we downsample the training data from the “all” set. All reward models are trained with the default reward head, as a linear layer with shape (hidden size, 1). For NORMBT, the normalization strength α is set to 1 by default. For label smoothing, α is set to 0.1 by default. For

all BT baselines, we conduct an extensive grid search over learning rates and report the best-tuned results. For BT-based variants (margin, margin-out, label smoothing), we search over a range of learning rates centered around the best value found for BT and report the best configuration. All models are trained for 1 epoch with full parameter tuning. We truncate the inputs over 4096 tokens, and an effective batch size of 256 with gradient accumulation. Our experiments are conducted using NVIDIA RTX A6000 49G.

Table 3: Key hyperparameter details in reward model training.

Basic information	
Datasets	Unified-Feedback (80K), Skywork-Reward-Preference-80K-v0.2
Base models	gemma-2b-it, Llama-3.2-3b-Instruct
Quantization for training	bf16
Optimizer	sgd
Momentum	0.9
Batch size	256
Gradient accumulation	4
Learning Rate Scheduler	cosine
Warmup Ratio	0.03

C EXPERIMENT ERROR ANALYSIS

To assess the robustness of our main results reported in Table 1, we conduct an expanded evaluation on five random seeds for each method. Using a 40K subset of the Unified-Feedback dataset, we train both NORMBT and the BT baseline following the same training configuration described in Appendix B. We compare these 10 models to illustrate the performance improvement of NORMBT over the BT baseline. The results reported in Table 4 and Figure 6 are consistent with the main experiments, showing higher average scores across the majority of RewardBench and particularly strong gains in the Reasoning category.

Reward Model	Average	Chat	Chat Hard	Safety	Reasoning
BT (Baseline)	68.15	94.69	37.48	75.11	65.34
NORMBT (Ours)	69.62	95.03	38.00	74.95	70.50

Table 4: Comparison of Ten Models. Mean performance on RewardBench over models trained on five random seeds aligns with the main results reported in Table 1.

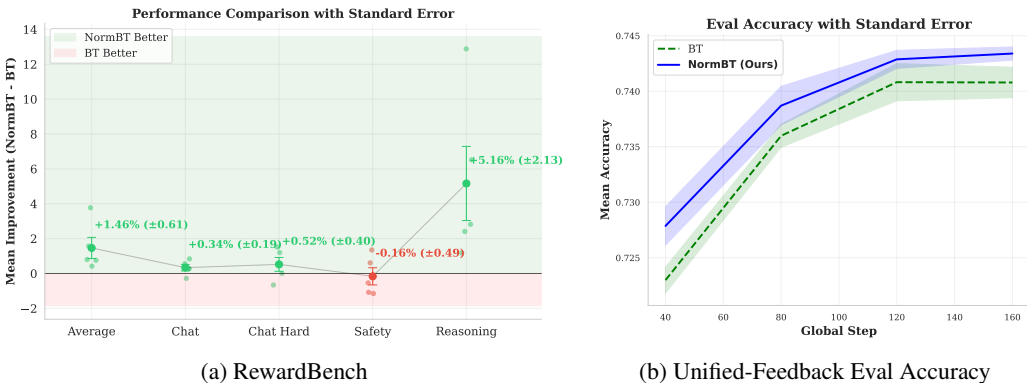


Figure 6: Comparison on Ten Models. Detailed performance of models trained on five random seeds with standard errors on (a) RewardBench categories, and (b) Unified-Feedback Evaluation.

D ADDITIONAL EXAMPLES

Small Representation Distance Pairs. This section follows up on the performance comparison between BT Baseline and NORMBT in Figure 4. We provide example instances of “Small-Distance” response pairs from RewardBench where NORMBT predicts correctly while BT Baseline does not. This gives a clear illustration to pinpoint the preference pairs where NORMBT gives the largest performance gains.

```

19 lines Copy 1 addition 19 lines Copy
1 isPrime := func(n int) bool {
2   for i := 2; i < int(math.Pow(float64(n), 0.5)+1); i++ {
3     if n%i == 0 {
4       return false
5     }
6   }
7   return true
8 }
9
10 largest := 1
11 for j := 2; j < n + 1; j++ {
12   if n % j == 0 && isPrime(j) {
13     if j > largest {
14       largest = j
15     }
16   }
17 }
18 return largest
19 }

1 isPrime := func(n int) bool {
2   for i := 2; i < int(math.Pow(float64(n), 0.5)+1); i++ {
3     if n%i == 0 {
4       return false
5     }
6   }
7   return true
8 }
9
10 largest := 1
11 for j := 2; j < n + 1; j++ {
12   if n % j == 0 && isPrime(n) {
13     if j > largest {
14       largest = j
15     }
16   }
17 }
18 return largest
19 }

34 lines Copy 1 addition 34 lines Copy
1 letter_grade := make([]string, 0, len(grades))
2 for _, gpa := range grades {
3   switch {
4     case gpa == 4.0:
5     letter_grade = append(letter_grade, "A+")
6     case gpa > 3.7:
7     letter_grade = append(letter_grade, "A")
8     case gpa > 3.3:
9     letter_grade = append(letter_grade, "A-")
10    case gpa > 3.0:
11    letter_grade = append(letter_grade, "B+")
12    case gpa > 2.7:
13    letter_grade = append(letter_grade, "B")
14    case gpa > 2.3:
15    letter_grade = append(letter_grade, "B-")
16    case gpa > 2.0:
17    letter_grade = append(letter_grade, "C+")
18    case gpa > 1.7:
19    letter_grade = append(letter_grade, "C")
20    case gpa > 1.3:
21    letter_grade = append(letter_grade, "C-")
22    case gpa > 1.0:
23    letter_grade = append(letter_grade, "D+")
24    case gpa > 0.7:
25    letter_grade = append(letter_grade, "D")
26    case gpa > 0.0:
27    letter_grade = append(letter_grade, "D-")
28    default:
29    letter_grade = append(letter_grade, "E")
30   }
31 }
32 }
33 return letter_grade
34 }

1 letter_grade := make([]string, 0, len(grades))
2 for _, gpa := range grades {
3   switch {
4     case gpa == 4.0:
5     letter_grade = append(letter_grade, "A+")
6     case gpa > 3.7:
7     letter_grade = append(letter_grade, "A")
8     case gpa > 3.3:
9     letter_grade = append(letter_grade, "A-")
10    case gpa > 3.0:
11    letter_grade = append(letter_grade, "B+")
12    case gpa > 2.7:
13    letter_grade = append(letter_grade, "B")
14    case gpa > 2.3:
15    letter_grade = append(letter_grade, "B-")
16    case gpa > 2.0:
17    letter_grade = append(letter_grade, "C+")
18    case gpa > 1.7:
19    letter_grade = append(letter_grade, "C")
20    case gpa > 1.3:
21    letter_grade = append(letter_grade, "C-")
22    case gpa > 1.0:
23    letter_grade = append(letter_grade, "D+")
24    case gpa > 0.7:
25    letter_grade = append(letter_grade, "D")
26    case gpa > 0.0:
27    letter_grade = append(letter_grade, "D-")
28    default:
29    letter_grade = append(letter_grade, "E+")
30   }
31 }
32 }
33 return letter_grade
34 }

11 lines Copy 2 additions 11 lines Copy
1 let t = 0
2 if (S[0] == 'I' && S[1] == 'I') { t = 1 }
3 for (let i = 0; i < S.Length; i++) {
4   if (S[i] == '.' || S[i] == '!' || S[i] == '?') {
5     if (S[i + 1] == '.' && S[i + 2] == 'I' && S[i + 3] == 'I') {
6       t++
7     }
8   }
9 }
10 return t
11 }

1 let t = 0
2 if (S[0] == '.' && S[1] == 'I') { t = 1 }
3 for (let i = 0; i < S.Length; i++) {
4   if (S[i] == '.' || S[i] == '!' || S[i] == '?') {
5     if (S[i + 1] == '.' && S[i + 2] == 'I' && S[i + 3] == 'I') {
6       t++
7     }
8   }
9 }
10 return t
11 }

```

Figure 7: **Small-Distance Pairs.** Example response pairs from RewardBench, where NORMBT predicts correctly on while BT Baseline does not. These pairs fall within the “Small-Distance” range. Response pairs from three different prompts are shown, where left shows the chosen the right shows the rejected response.

E FULL VS. PROXY REPRESENTATION DISTANCE

This section analyzes empirically the relationship between the full representation distance $\|\nabla_{\theta}(r_w - r_l)\|$ and its proxy $\|h_w - h_l\|$ as defined in Sec 2.1. Figure 8 plots the correlation between these two quantities. For gamma-2b-it, the correlation is strong with $r = 0.928$, suggesting that the embedding difference is a highly reliable stand-in for the full representation distance. For Llama-3.2-3b-Instruct, the correlation is more modest at the beginning of reward model training ($r = 0.682$; shown in Figure 8b) but increases to $r = 0.932$ after some initial training. This may indicate that the proxy becomes more accurate once the pretrained model begins adapting to the reward modeling objective.

These observations support the theoretical analysis in Eq. 7, which establishes the connection between $\|\nabla_{\theta}(r_w - r_l)\|$ and $\|h_w - h_l\|$. This suggests the strategy to utilize $\|h_w - h_l\|$ as a proxy is a computationally efficient and empirically validated substitute for the full distance, avoiding costly backward passes.

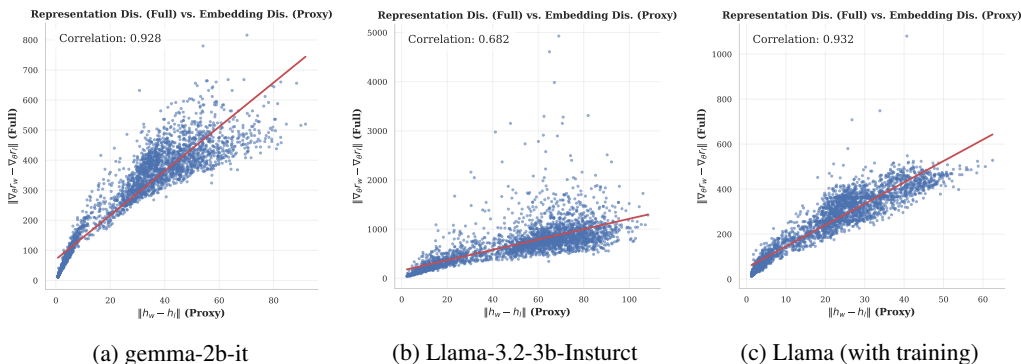


Figure 8: **Correlation of Full vs. Proxy Representation Distance.** Both base models show a strong correlation ($r > 0.9$) between the two quantities.

F REPRESENTATION SCALE & EMA

One challenge in NORMBT normalization arises from the non-stationary behavior of representation distances $\|h_w - h_l\|$ during training of the reward model. Since the LLM backbone is jointly optimized with the reward head, the learned representation space could evolve considerably.

Figure 9 illustrates this observation: the mean representation distance shifts significantly over the course of optimization, while the variance remains consistently large. And the same behavior is observed across both Unified-Feedback and Skywork-Reward-Preference-80K-v0.2 datasets, as well as for all BT baselines. As a result, although the large variance highlights the potential gradient imbalance, any normalization scheme that relies directly on raw representation magnitudes inherits the instability created by scale shifts.

To mitigate this issue, we incorporate an Exponential Moving Average (EMA) as stated in Eq 12. By tracking a running mean of representation distance, EMA provides a stable and adaptive reference point that smooths out fluctuations in embedding scale while remaining responsive to long-term trends. By normalizing each pair relative to this moving reference, rather than to its raw magnitude, we ensure that weighting reflects relative distinctions among pairs. This yields consistent gradient scaling across training. The benefit of EMA is also reflected in the performance gap observed in ablation studies shown in Table 2.

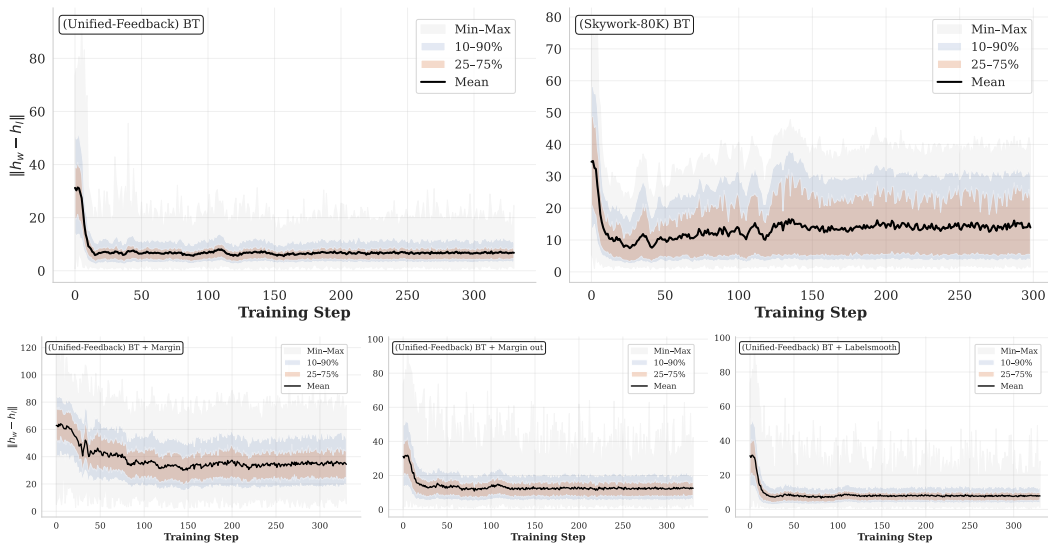


Figure 9: **Dynamics of Representation Distance** during RM Training for standard BT on Unified-Feedback and Skywork-80K (top), and all BT variants on Unified-Feedback (bottom). For each method, we report the mean, 25–75 percentile, 10–90 percentile, and full min–max range. Across all settings, the mean distance shifts substantially, while the variance remains high.

G LIMITATIONS & FUTURE WORK

Method. While NORMBT yields consistent gains on fine-grained distinctions, a slight trade-off remains in larger-distance regions; more intricate scaling strategies may help mitigate this. Unified-Feedback and Skywork-80K are human-annotated or curated datasets, and settings with substantial noise or near-duplicate pairs may require additional handling.

Evaluation. Our comparison focuses on BT-based variants. Techniques involving regularization or auxiliary objectives may alter gradient dynamics beyond Eq. 7, and exploring how NORMBT interacts with such methods is an interesting direction. We use Best-of-N to mirror downstream RLHF usage; full policy-level evaluations (Schulman et al., 2017; Ahmadian et al., 2024) would offer a more comprehensive assessment.

Analysis. Our analysis centers on gradient magnitudes to quantify each pair’s contribution to model updates. Incorporating gradient-direction or correlation analysis would provide complementary insight into the underlying representation space.

1026 H INTUITION: LARGE- VS. SMALL-DISTANCE PAIRS

1027
1028 Small representation distance can arise from two reasons: (1) **uninformative or noisy pairs**, such
1029 as near-duplicate responses that convey little preference signal; or (2) **difficult but meaningful**
1030 **pairs**, where the backbone fails to separate two responses despite clear human preference signal
1031 (e.g., correct vs. subtly flawed code). As illustrated in Figure 2, such case (2) pairs are common in
1032 practice, and Figure 7 provides concrete examples.

1033 This distinction matters because BT loss treats both cases identically: small-distance pairs always
1034 receive small gradients. While suppressing case (1) is desirable, suppressing case (2) is harmful, as
1035 these are precisely the pairs containing the fine-grained distinctions that the reward model should
1036 learn but inherently struggles to separate with the current Φ .

1037 In our datasets (UnifiedFeedback and Skywork-80K), most pairs are human-annotated or carefully
1038 curated, and following Yang et al. (2024), pairs with identical ratings are filtered out. This sub-
1039 stantially reduces the prevalence of case (1), making small-distance pairs far more likely to reflect
1040 case (2) of the difficult, fine-grained distinctions that RM should learn. Viewed through this lens,
1041 NORMBT effectively *upweights hard, meaningful pairs* whose gradients would otherwise be sup-
1042 pressed due to representation limitations in the current model.

1044 I RELATION TO ACTIVE LEARNING APPROACHES

1045
1046 The above intuition also clarifies how our analysis relates to active learning approaches such as
1047 PILAF (Feng et al., 2025) and ARM (Shen et al., 2025), which prioritize **large-distance pairs** to
1048 improve sample efficiency. While this appears at odds with our motivation, the two settings address
1049 fundamentally different problems.

1050 Active-learning methods focus on **sample selection**: given a noisy pool of model-generated re-
1051 sponses, identify pairs that are maximally informative. In this setting, many candidates reflect case
1052 (1) from Appendix H as uninformative or trivial pairs that provide little preference signal. Moreover,
1053 ARM adopts an embedding-based RM with a *frozen* LLM backbone, meaning Φ is assumed reliable
1054 and representation learning is not part of training. Under these assumptions, difficult but meaningful
1055 small-distance pairs cannot be corrected, thus selecting large-distance pairs becomes a principled
1056 strategy for sample efficiency.

1057 In contrast, sequence-classifier RMs jointly train the backbone and reward head. The assumption
1058 of fixed, reliable Φ no longer holds, especially as the model newly adapts to the task. Therefore,
1059 a small-distance pair could arise because the backbone has not yet learned to separate meaningful
1060 distinctions (i.e., case (2)), rather than a lack of meaningful preference information. These are
1061 precisely the pairs where the reward model needs stronger updates. Yet under standard BT loss,
1062 *both case (1) and case (2) receive small gradients, creating a learning bottleneck.*

1063 This distinction is amplified in our datasets. UnifiedFeedback and Skywork-80K contain human-
1064 annotated or carefully curated pairs, and identical-rating pairs are filtered out following Yang et al.
1065 (2024). This substantially removes case (1) examples, making small-distance pairs far more likely
1066 to represent meaningful but difficult decisions. In this regime, prioritizing large-distance pairs (as
1067 active-learning methods do) suppresses the critical fine-grained distinctions.

1068 Our empirical results also reflect this: active-learning style D-opt scaling underperforms in our
1069 setting (Table 2), since the difficult, fine-grained distinctions driving performance in Reasoning are
1070 further suppressed. In contrast, NORMBT improves these regions while maintaining competitive
1071 performance elsewhere. However, we also note that our D-opt scaling uses a heuristic continuous
1072 relaxation for fair comparison using the same dataset and ordering. This differs from its original
1073 use as a sample-selection criterion, and we include it only to highlight the conceptual differences
1074 between active-learning approaches and NORMBT in the context of reward model optimization.