Mechanistic Reward Models in the Loop: Causal Feature Control During RLHF

Nikhil Khandekar

1 Problem Statement

Problem statement: I will investigate whether in-loop, feature-level control of a reward model (RM) during reinforcement learning from human feedback (RLHF) can reduce reward hacking—cases where a policy artificially increases its RM score by exploiting biases (e.g., verbosity or unsafe strategies) rather than by genuinely improving quality or safety. The central question is whether real-time control of the RM's internal features can prevent such shortcuts while preserving or enhancing model helpfulness.

Motivation: This problem is important because RLHF—and its variants like GRPO and DPO—is the dominant framework for aligning large language models. However, RMs often encode hidden biases or spurious correlations that policies learn to exploit, leading to poor generalization and unsafe behavior. Recent interpretability methods, such as sparse autoencoders and causal activation tests, provide new ways to identify meaningful internal features of RMs. Turning these insights into active, training-time interventions offers a principled path toward more trustworthy and causally grounded alignment.

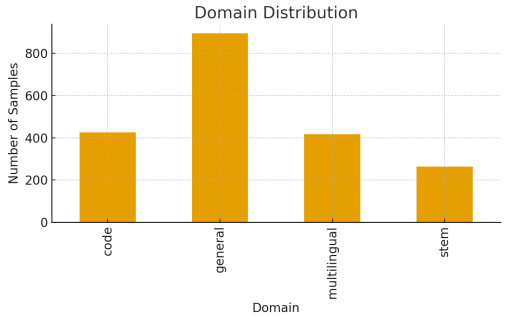
Setting and goal: My setting follows the standard SFT $\rightarrow$ RM $\rightarrow$ PPO/GRPO pipeline. I instrument the RM with a sparse autoencoder, validate a small set of interpretable helpfulness and safety features through causal testing, and then modify the RM's forward pass during training using bounded clamps, penalties, or a feature-level KL regularizer. The overarching goal is to reshape the policy's gradient field—reducing the Goodhart gap between RM score and true quality—while maintaining or improving the helpfulness and safety of the resulting model.

2 Relevance to Trustworthy Aspects

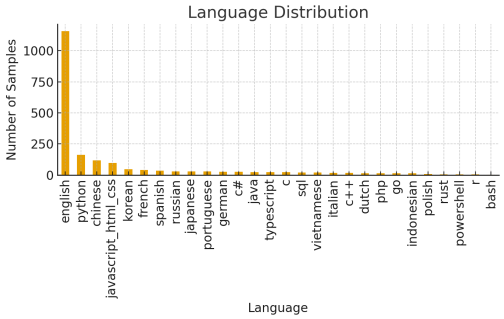
This project is most closely connected to the explainability and robustness aspects of trustworthy machine learning. It aims to make the reward model (RM)—a critical but often opaque component in reinforcement learning from human feedback (RLHF)—more interpretable and resilient to exploitation. By identifying and controlling specific internal features that correspond to helpfulness or safety, the project seeks to make the RM's decision process more transparent and causally grounded. This contributes to explainability, as researchers and practitioners can better understand why the RM assigns high rewards to certain behaviors.

At the same time, the project evaluates and extends robustness by testing whether feature-level control can prevent reward hacking. Reward hacking is a form of adversarial behavior where the policy finds shortcuts to maximize the RM score without genuinely improving performance or safety. If in-loop feature control can reduce these shortcuts, the resulting RLHF pipeline would produce models that are more stable and aligned across diverse conditions. Therefore, this work both applies explainability techniques (like sparse autoencoders) and extends robustness research by integrating interpretability directly into the training loop to enhance trustworthy alignment.

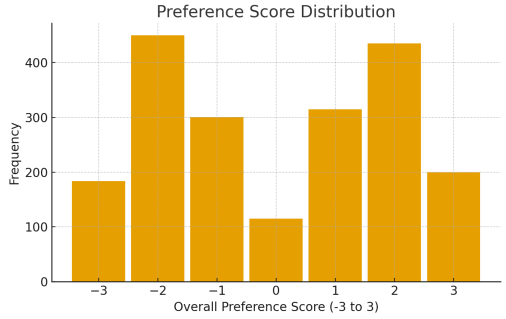
Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.



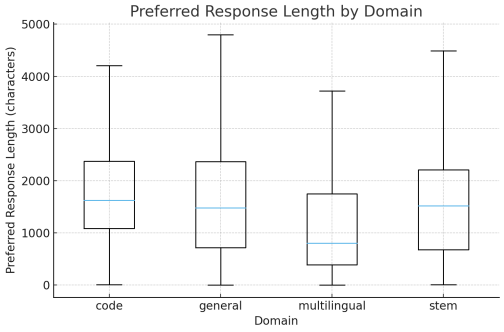
(a) Domain distribution (General, Code, Multilingual, STEM).



(b) Language distribution (note: Code uses programming-language tags).



(c) Preference score distribution (-3 to 3), approximately centered.



(d) Preferred response length by domain (characters).

Figure 1: EDA on a 2,000-sample subset of HelpSteer3-Preference. The corpus is skewed toward General and English; the language field mixes natural and programming languages; preference labels span weak and strong choices; and preferred response lengths vary by domain.

3 Dataset

4 Dataset

Corpus. I use the **HelpSteer3-Preference** dataset, a publicly available human preference corpus released by NVIDIA. Each example contains a conversational prompt (context), two candidate responses, and an overall preference label in $\{-3, -2, -1, 0, 1, 2, 3\}$ that indicates which response is preferred and by how much. The full dataset has $> 40{,}000$ comparisons across General, STEM, Code, and Multilingual tasks, and is released under CC-BY-4.0.

Subset used for EDA. I ran exploratory data analysis (EDA) on a *2,000-sample* subset of the training split (the CSV used to produce the figures in Fig. 1). The domain distribution in this subset is: *General* 893, *Code* 426, *Multilingual* 417, *STEM* 264. The language field is dominated by English (1,157 examples), and—within the Code domain—includes programming-language tags (e.g., Python 161, JavaScript/HTML/CSS 95), so analyses that group by "language" should account for this schema detail.

Preference distribution. Labels are approximately centered: counts are $\{-3{:}\,184,\ -2{:}\,450,\ -1{:}\,301,\ 0{:}\,115,\ 1{:}\,315,\ 2{:}\,435,\ 3{:}\,200\}$. This provides a balanced range of weak and strong preferences for training and evaluation.

Length characteristics. Preferred-response length (characters; mean) varies by domain: *Code* 1,805.6, *General* 1,671.5, *STEM* 1,659.8, *Multilingual* 1,159.2. The chosen response is longer than the rejected response in **40%** of non-tie pairs (sign $\neq 0$). There is a *small but significant* association between the *absolute* length difference and the *magnitude* of the preference ($r{=}0.159$, $p{<}0.001$ via permutation testing), suggesting that large length gaps can amplify annotator judgments even though "longer wins" is not generally true.

Implications and challenges. The dataset covers multiple domains and languages (plus programming-language tags), which is helpful for testing generalization of reward models. At the same time, there are skews (more General and English; fewer STEM/Multilingual), mixed semantics in the language field, and cross-domain length differences (Fig. 1). These characteristics make the corpus a good stress test for this project's in-loop, feature-level controls aimed at reducing shortcut exploitation (e.g., verbosity) while preserving helpfulness and safety.

External benchmarks. To assess out-of-distribution generalization and style robustness, I also evaluate on two external RM evaluation suites: **RewardBench-2** and **RM-Bench**.

RewardBench-2 is a large-scale, multi-skill benchmark for reward models, designed to measure generalization to unseen human-written prompts across categories such as *Factuality*, *Precise Instruction Following*, *Math/Reasoning*, *Safety*, *Focus/On-Topic*, and *Ties*. Each example includes one preferred and several dispreferred completions (best-of-N), enabling finer-grained discrimination than pairwise preference tests. RewardBench-2 has been shown to be substantially harder than its predecessor, with average RM accuracy roughly 20 points lower across models, indicating stronger coverage of complex, nuanced tasks. Because RewardBench-2's performance correlates with downstream RLHF policy quality, it serves as an external validation that my modified reward model retains general alignment capability under more difficult and diverse conditions.

RM-Bench complements this by directly probing *style robustness* and *subtlety sensitivity*. For each prompt, it provides multiple style-controlled completions (concise, detailed-plain, detailed-markdown) across four domains—*Chat*, *Code*, *Math*, and *Safety*. Chosen/rejected pairs differ only by subtle factual or reasoning differences, rather than gross stylistic changes, allowing the benchmark to isolate whether an RM overweights superficial cues like verbosity or formatting. Empirically, state-of-the-art RMs achieve about 46% accuracy in the hardest style regime, demonstrating that many models still reward style rather than substance. RM-Bench therefore serves as a targeted evaluation for this work's feature-level interventions, especially those controlling verbosity- and format-related latent dimensions within the RM.

Together, RewardBench-2 and RM-Bench provide complementary external evaluations: the former measures broad generalization and correlation with human preference, while the latter stresses style invariance and subtle content discrimination. Reporting both allows quantitative assessment of whether in-loop latent feature controls improve RM trustworthiness without sacrificing general capability.

5 Related Works

Several recent works provide important context for my proposed investigation. First, "InfoRM: Mitigating Reward Hacking in RLHF via Information-Theoretic Reward Modeling" introduces a variational information bottleneck in the reward model to filter out irrelevant features and proposes a latent-space "Cluster Separation Index" to detect over-optimization (Miao et al., 2024). This relates to my project by directly addressing reward-hacking via spurious features, though it does not offer real-time, in-loop feature control of the reward model during policy training. Second, "ODIN: Disentangled Reward Mitigates Hacking in RLHF" focuses on length-bias in RLHF by training dual heads (one correlating with length and one decorrelating), then discarding the length-head during RL to mitigate the exploit (Chen et al., 2024). It is a concrete baseline for feature-level control of one spurious feature, but it does not generalize to multiple interpretable internal features nor integrate control inside the RM's forward pass. Third, "Interpreting Reward Models in RLHF-Tuned Language Models Using Sparse Autoencoders" applies sparse autoencoders on reward-model activations to identify latent features and study reward integrity (2023). This directly overlaps with my choice of interpretability tool (sparse autoencoders) to identify meaningful features in the RM; however, it stops at analysis/interpretation, whereas my project takes the next step of intervening on features during training. Fourth, "Reward Shaping to Mitigate Reward Hacking in RLHF" presents a systematic study of reward-shaping methods (notably the "Preference As Reward (PAR)" technique) to reduce reward-hacking (Fu et al., 2025). While this addresses mitigation of shortcuts, it focuses on the scalar reward signal rather than the internal feature structure of the RM. Fifth, "Bias Fitting to Mitigate Length Bias of Reward Model in RLHF" identifies and corrects length-bias in reward models by fitting a lightweight model to model the non-linear relation between length and reward and then debiasing the reward model accordingly (Zhao et al., 2025). This is another perspective on a specific spurious

feature (length) in RMs; it reinforces the relevance of my problem (feature-level reward hacking) but again remains post-hoc/focused on one feature—my project generalises to multiple features and intervenes during RLHF training. Sixth, "Circuit-Aware Reward Training: A Mechanistic Framework for Longtail Robustness in RLHF" (Liu, 2025) proposes a mechanistic interpretability framework that identifies specialized neural circuits within reward models handling rare ("long-tail") scenarios, and introduces CART (Circuit-Aware Reward Training) which uses circuit analysis, data augmentation, regularisation and ensembling to target those weak circuits. This work is closely connected to mine because it uses internal feature/circuit analysis for reward-model robustness and reward-hacking mitigation, but it emphasises circuit-level diagnosis and external interventions rather than real-time feature-level clamping or gradient-shaping inside the RM during policy optimisation.

Together, these works chart important prior directions: interpretability of reward models, detection and mitigation of reward-hacking, and shaping of reward signals. Their limitations—narrow focus on one spurious feature, diagnostics rather than active control, scalar reward interventions rather than in-loop feature-level manipulation—help position my proposed project as a novel contribution. Specifically, by combining sparse-autoencoder-based interpretability with in-loop feature-level control (clamps, penalties, KL regularisers) inside the reward model during RLHF, I aim to generalise beyond existing baselines, reduce the Goodhart gap between reward-model score and true quality/safety, and reshape the policy's gradient field in a causally grounded way.

6 Proposed Approach

My approach unfolds in three phases: RM instrumentation, causal feature identification, and in-loop feature control during policy optimization within a standard SFT $\rightarrow$ RM $\rightarrow$ RLHF (PPO-style) pipeline. During reinforcement learning, the **reward model (RM) is frozen**, and latent features are computed **without gradient propagation**; only the policy receives gradients from the scalar reward.

Phase 1: RM Instrumentation. Starting from a supervised fine-tuned policy and a reward model trained on human preferences, I will embed a *sparse autoencoder* (SAE) on intermediate RM activations to learn a sparse, low-dimensional latent space that exposes interpretable features (e.g., verbosity, refusal, formatting cues). This enables access to the RM's internal representations rather than relying solely on the final scalar reward.

Phase 2: Causal Feature Identification. From the SAE's latent outputs, I will select candidate features that correlate with helpfulness, safety, or known shortcut patterns (such as verbosity). I will then conduct *causal intervention experiments*—clamping features high or low—to observe how the RM's reward and *selection behavior* (e.g., best-of-N reranking outcomes) change. Importantly, *no policy parameters are updated in this phase*; observed effects are restricted to reward and ranking changes. This isolates which latent axes correspond to genuinely desirable behaviors versus spurious correlations.

Phase 3: In-Loop Feature-Level Control. During the PPO (or PPO-style) stage, I will use two complementary control mechanisms while keeping the RM weights frozen and detaching latent computations from autograd:

- **In-RM clamping (activation edits):** Constrain selected latent dimensions directly within the RM's forward pass before the final scalar reward is computed.
- **External scalar shaping:** Add penalties or regularizers to the scalar reward that PPO optimizes.

The per-trajectory reward is then defined as:

$$R_{total} = R_{RM} - \alpha \cdot \mathbf{1}[z_i > \tau] - \beta \cdot \widehat{D}_{\mathrm{KL}}\big(p(z_{policy}) \,\|\, p(z_{human})\big),$$

where z_i is a latent feature, τ a threshold, and α, β are hyperparameters. The term $\widehat{D}_{\mathrm{KL}}$ denotes an estimated distributional penalty computed via a *mini-batch density-ratio discriminator* trained to distinguish SAE latents from (i) human demonstration responses and (ii) current policy responses. The discriminator's logit provides a stable batchwise KL estimate that can be used as a per-trajectory scalar penalty. Other batch estimators such as kernel density estimation (KDE) or maximum mean discrepancy (MMD) can also be substituted if they yield more stable gradients.

Differences and Challenges. Unlike previous work that intervenes only at the scalar reward level or targets a single artifact (e.g., length bias), this approach performs *multi-feature, in-loop control* directly within the RM's feature space while optimizing the policy. This setup explicitly aims to reduce the Goodhart gap between proxy reward and true quality or safety. Key challenges include: (i) reliably identifying causally meaningful latent features, (ii) setting thresholds and weights that preserve helpfulness and safety, and (iii) maintaining PPO stability when adding feature-aware terms. These are mitigated by freezing the RM, detaching latent paths, and using stable batchwise estimators for the distributional regularizer. For completeness, note that alternative preference-optimization methods such as *Direct Preference Optimization (DPO)* bypass the RLHF stage entirely; however, our experiments focus on PPO-style RLHF because it explicitly exposes the RM's internal feature space for controllable interventions.

7 Evaluation Metrics

I will evaluate the proposed approach using both quantitative metrics and qualitative analyses.

1. Performance Evaluation. I will measure the trained policy's helpfulness and safety using established benchmarks for reward models such as RM-Bench and RewardBench-2. Improvement in overall preference accuracy and robustness under these benchmarks will indicate successful mitigation of reward hacking.

2. Robustness and Bias Analysis. I will test whether the modified RM reduces known shortcut exploitations such as length or verbosity bias. This will involve computing correlation between response length and reward score before and after applying feature-level controls.

3. Ablation Studies. I will perform ablation experiments to isolate the effect of each intervention (clamping, penalty, KL regularization) and each targeted feature. For example, comparing RM-only control vs. full PPO integration will help determine which design elements contribute most to stability and bias reduction.

4. Interpretability and Visualization. I will visualize the sparse autoencoder's latent features using t-SNE or PCA to confirm that identified features correspond to human-interpretable concepts such as "helpfulness", "safety", or "verbosity". These visualizations will help explain how in-loop controls affect internal representations over the course of RL training.

Success Criteria. The method will be considered successful if:

- The modified RM and policy outperform baselines on RM-Bench and RewardBench-2.
- Correlation between length (or other spurious features) and reward score decreases significantly.
- The resulting model maintains or improves helpfulness and safety relative to standard RLHF models.

8 Timeline

Table 1 presents the planned milestones for the semester, with weekly checkpoints scheduled each Tuesday between October 28 and December 11.

Table 1: Project timeline and milestones.

Milestone	**Target Date (Tuesday)**
Set up data and baseline RM models	Oct 28
Implement feature-level control methods	Nov 4
Intermediate results and debugging checkpoint	Nov 11
Evaluation and ablation studies	Nov 18
Interpretability and visualization analysis	Nov 25
Draft report submission	Dec 2
Final report submission	Dec 9