HARM: Learning Hate-Aware Reward Model for Evaluating Natural Language Explanations of Offensive Content

Anonymous ACL submission

Abstract

Explaining why content is hateful using natural language is crucial for fostering transparency in automated content moderation systems. However, evaluating the quality of such explanations remains an open challenge. General-purpose reward models (RMs), commonly used for scoring natural language outputs, are typically optimized for broad notions of safety. As a result, they tend to penalize necessary references to stereotypes or offensive framing, elements that are essential for faithful hate speech explanations. To address this gap, we introduce **SBIC-Explain**, a dataset of 370,788 LLM generated NLEs for offensive content, spanning three levels of human-annotated contextual richness: **Tier 1:** text-only, **Tier 2:** + classification-aware, and **Tier 3:** + semantics-informed. We hypothesize that as human-annotated context increases, explanations should better reflect human preferences. Yet, we find that existing RMs systematically assign lower scores to more contextually rich (and often more offensive) explanations, revealing a misalignment between model preferences and explanatory fidelity for this context. We propose **HARM (Hate-Aware Reward Model)**, a RM that integrates interpretable signals to better align reward scores with the needs of hate speech explanation. HARM outperforms general-purpose baselines, improving NLE pair-wise preference, taking accuracy from 0.66 to 0.80.

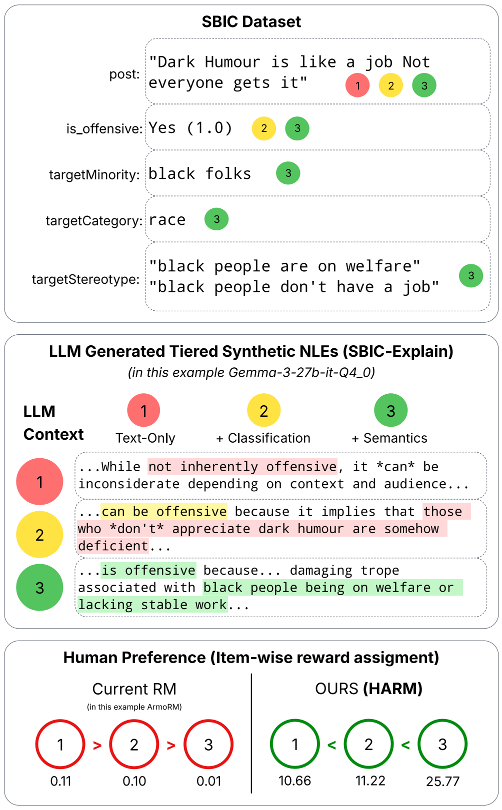

Figure 1: **Concept Overview.** We leverage the SBIC dataset (Sap et al., 2020) (top) to generate LLM synthetic NLEs (middle) under three levels of contextual access. Intuitively, access to richer human context should yield explanations that receive higher reward scores. However, existing general-purpose reward models (RMs) assign lower scores as the language becomes more sensitive (bottom). Our proposed method, **HARM**, learns to correctly reflect the expected preference ordering: Tier 1 < Tier 2 < Tier 3.

1 Introduction

The proliferation of hate speech on social media poses substantial risks to individuals and communities, threatening social cohesion and even inciting real-world violence (Jahan and Oussalah, 2021; Malik et al., 2023). In response, automated hate speech detection systems have become essential for moderating online content and ensuring safer digital environments (Narula and Chaudhary, 2024). However, beyond mere classification, a central challenge lies in understanding why content is hateful. One promising solution is the use of AI-generated NLEs, which aim to make model reasoning interpretable, enhancing transparency and fostering user trust (Mathew et al., 2022).

Yet, effectively training and evaluating such explanations remains difficult, in part due to limitations in current explainable hate speech datasets. Most existing resources rely on shallow signals, such as highlighting offensive spans (Mathew et al., 2022; Arshad and Shahzad, 2024; Delbari et al., 2024; Hoang et al., 2023; Pavlopoulos et al., 2021; Ravikiran and Annamalai, 2021), which lack the depth needed to support rich, contextual, and stereotype-aware explanations (Sap et al., 2020). While useful for pinpointing relevant tokens, such approaches often fail to capture the broader social context or implicit stereotypes that underpin many harmful messages. For example, highlighting the word "ginger" does little to surface its role as a slur against redheads. Similarly, in the statement "Dark Humour is like a job Not everyone gets it" (from Figure 1), the surface tone masks a stereotype associating racial groups with unemployment (Sap et al., 2020).

The Social Bias Inference Corpus (SBIC) (Sap et al., 2020) addresses these shortcomings by including human-written "implied statements" that surface the stereotypical or biased assumptions underlying ostensibly innocuous content. While SBIC was originally constructed to capture offensiveness, it offers a rich annotation framework for studying **implicit hate speech** (the kind that relies on euphemism, ambiguity, or stereotypes rather than explicit slurs (Kim et al., 2023; Fortuna and Nunes, 2018)).

Despite the growing use of LLM-generated NLEs in hate speech detection (Huang et al., 2024, 2023), existing datasets such as HateCOT (Nghiem and Daumé Iii, 2024) overlook this contextual depth. They do not systematically evaluate how access to rich, stereotype-aware human annotations, like those in SBIC, affects explanation quality. This gap raises a crucial question: **How much does access to human-annotated contextual information improve the quality of NLEs in hate speech detection?**

This gap in explanation quality evaluation is not just a data issue, it also stems from how explanation outputs are scored. Beyond their role in reinforcement learning for fine-tuning language models (e.g., RLHF), reward models (RMs) are increasingly used as scoring functions to evaluate the quality of LLM-generated content, including NLEs. However, general-purpose RMs are poorly aligned with the goals of hate speech explanation (Christian et al., 2025). Trained primarily to promote safety, they tend to penalize outputs that reference stereotypes or offensive language, even when such content is necessary for a truthful explanation (Entezami and Naseh, 2025; Lambert et al., 2024a). This misalignment incentivizes the production of sanitized yet unfaithful explanations, which sacrifice nuance and clarity for safety compliance. The result reflects a deeper alignment dilemma: while harmful language must be curbed, faithful explanations often require referencing uncomfortable truths (Chua et al., 2024; Lyu et al., 2024).

To investigate these phenomena, we introduce **SBIC-Explain**, a dataset of synthetic NLEs grounded in SBIC's human-annotated stereotypes. We show that state-of-the-art reward models systematically undervalue faithful hate speech explanations, particularly those referencing offensive stereotypes. To bridge this gap, we propose a lightweight method for building a domain-specific **Hate-Aware Reward Model (HARM)**, which reweights interpretable outputs of general-purpose reward models to better evaluate hate-related content.

We organize our contributions as follows:

(i) We construct and release **SBIC-Explain**, a multi-tier, multi-model synthetic NLE dataset grounded in stereotype-level annotations, to benchmark the effect of contextual depth on hate speech explanation. **(ii)** We reveal a critical failure mode in current reward models in the context of NLE for offensive content, showing how safety-oriented training biases them against truthful explanations. **(iii)** We propose **HARM**, a hate-aware reward model that uses a lightweight adapter to improve NLE evaluation for sensitive content. We show that general-purpose, interpretable reward models can be re-prioritized to serve this goal.

2 Related Work

2.1 Explainable Hate Speech Datasets

In the landscape of existing explainable hate speech datasets, Table 5 in Appendix E reveal that most works treat explainability as a span annotation.

Predominance of Span Rationales. Seven out of nine listed datasets (HateBRXplain (Mathew et al., 2022), HateInsights (Arshad and Shahzad, 2024), PHate (Delbari et al., 2024), ViHOS (Hoang et al., 2023), HateXplain (Mathew et al., 2022), SemEval-2021 (Pavlopoulos et al., 2021), DOSA (Ravikiran and Annamalai, 2021)) rely primarily on span rationales, where annotators highlight token-level or phrase-level segments deemed offensive.

While span rationales help models locate surface clues (e.g., explicit slurs), they frequently omit the broader socio-historical context behind a slur's offensiveness. In practice, this limitation can lead to shallow explanations, a classifier may learn to flag a word without understanding why that term is harmful to a particular group.

Unique Role of SBIC's Free-Text Annotations. SBIC (Sap et al., 2020) stands out in Table 5 because it provides not only a binary offensiveness label (`offensiveYN`) and categorical fields (`targetMinority`, `targetCategory`) but also free-text "implied statements" capturing why a particular target is stigmatized. Although SBIC has been used for training classifiers and for categorization tasks, downstream explainability work has largely ignored the option it offers for systematically testing the impact of its 34K implied statements for explainability (Nghiem and Daumé Iii, 2024). This underutilization represents a missed opportunity: free-text stereotypes offer semantic depth that neither span highlights nor simple labels can provide for grounding explanations.

Therefore, this work aims to test how much the incorporation of such semantic depth improves the perceived quality of NLEs, particularly in the eyes of reward models. **Our hypothesis is that explanations grounded in richer, stereotype-informed context will better align with human preferences**.

2.2 Evaluating NLEs with Preference Modeling

Early approaches that aimed to compare AI generated content with a human ground truth relied on lexical overlap metrics like BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004), but these are poorly correlated with human judgments of explanation quality (Sai et al., 2020). More recent metrics based on pre-trained language models (PLMs), like BERTScore (Zhang et al., 2020), offer better semantic evaluation, but are notorious for carrying unfair stereotypical bias such as racial, gender, or religion bias (Kaneko and Bollegala, 2021; Sun et al., 2022). This behavior is unacceptable for a work that aims to specifically deal with hate speech.

Reward Models (RMs) were created to improve the challenge of aligning large language models (LLMs) with human preferences, initially trained to predict a single scalar scalar score for a given context, and generated continuation (Ziegler et al., 2020; Chen et al., 2024b). These scalar-valued RMs underpin alignment techniques such as Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022). However, human preferences are rarely monolithic; they are inherently multi-dimensional, encompassing trade-offs between attributes like helpfulness, safety, coherence, and informativeness (Yu et al., 2025; Bai et al., 2022).

These limitations are especially pronounced in domains involving sensitive or identity-related language (Sap et al., 2019). Scalar reward models often fail to distinguish between harmful and contextually appropriate uses of terms like "sex" or "Black people," leading to large misalignments with human preferences (Christian et al., 2025). The deeper issue lies in RMs' tendency to penalize such terms uniformly in pursuit of a generalized harmlessness objective. This can result in overcensorship, linguistic erasure, and the suppression of valid discourse, particularly harmful in hate speech explanation, where referencing stereotypes may be necessary to faithfully convey intent (Christian et al., 2025).

Multi-attribute RMs are a natural evolution of single scalar RM, generating multiple scalars that disentangle genuine helpfulness from unimportant factors like length bias and offer precise, steerable rewards (Chen et al., 2024a). Datasets such as HelpSteer (Wang et al., 2023b) and UltraFeedback (Cui et al., 2024) provide fine-grained annotations across multiple dimensions. Building on these, ArmoRM (Wang et al., 2024) learns to predict multi-objective reward scalars (19[1]) for each response, trained using a standard regression loss against the annotated vector. To reduce these multidimensional outputs to a scalar reward for ranking or preference learning, ArmoRM learns a prompt-dependent gating function, which outputs non-negative weights over the reward dimensions (summing to one). This ideally allows the model to dynamically adjust the relative importance of each attribute.

Potential issue on RMs evaluation. To test RMs like ArmoRM, leading benchmarks such as RewardBench (Lambert et al., 2024b) and RewardBench 2 (Malik et al., 2025) are the standard for testing generalization across diverse tasks. Their

[1] All atribuites: helpsteer-helpfulness, helpsteer-correctness, helpsteer-coherence, helpsteer-complexity, helpsteer-verbosity, ultrafeedback-overall_score, ultrafeedback-instruction_following, ultrafeedback-truthfulness, ultrafeedback-honesty, ultrafeedback-helpfulness, beavertails-is_safe, prometheus-score, argilla-overall_quality, argilla-judge_lm, code-complexity, code-style, code-explanation, code-instruction_following, code-readability

evaluation of safety, a critical component, is heavily focused on a model's ability to refuse to generate harmful content. RewardBench, for example, draws from datasets like XSTest (Röttger et al., 2024) and Do-Not-Answer (Wang et al., 2023a), where the preferred response is a refusal and the rejected response is harmful. Similarly, RewardBench 2 leverages the CoCoNot taxonomy (Brahman et al., 2024) to assess compliance but explicitly excludes debatable categories to maintain a conservative stance on safety. While this emphasis on avoidance is crucial, it may inadvertently degrade model faithfulness in tasks that require nuanced discussion of sensitive topics. The CoCoNot dataset itself (Brahman et al., 2024) marks progress by proposing a broader taxonomy for noncompliance beyond just safety, yet it relies on synthetically generated data and a US-centric view of legality. Thus, while we recognize the importance of current benchmarks, their design may unintentionally steer models toward over-cautiousness, potentially sacrificing faithfulness for broad-stroke safety.

To address this, **we introduce a benchmark focused on contextual accuracy in hate-explicit scenarios and propose a method to re-weight general-purpose, multi-attribute reward models to prioritize explanatory faithfulness over blanket safety**. Together, these contributions enable more accurate evaluation and development of models that balance informativeness with harm reduction.

3 HARM - A Hate-Aware Reward Model via Interpretable Attributes Re-weighting

3.1 Reward Model

Our RM is designed to score the quality of NLEs for hate speech. This model, which we call **Hate-Aware Reward Model (HARM)**, leverages pretrained multi-attribute reward model by reweighting its interpretable dimensions, based on the discussions of Section 2.2.

Multi-Attribute Reward Model Backbone. Let x be a hate speech post from SBIC, and $E = G_i^{(t)}(x)$ be a model-generated explanation (NLE) produced by language model M_i under conditioning tier t, as defined in Section 4. Each explanation E is passed through a reward model that outputs a vector of d interpretable attribute scores: $\mathbf{v}_E \in \mathbf{R}^d$. These scores reflect semantic and stylistic properties such as helpfulness, coherence, truthfulness, and safety, depending on the training sources of the backbone reward model.

Mixture-of-Experts-Inspired Reward Modeling (HARM-MOE). We propose a RM formulation inspired by Mixture-of-Experts (MoE) architectures. Our approach conceptualizes the problem as a multitask learning challenge, similar to prior work in hate speech detection that uses specialized units to disentangle sentiment knowledge and improve system performance (Zhou et al., 2021). We hypothesize that an analogous division of labor is optimal for reward modeling in this domain, where the "tasks" correspond to judging explanations of either offensive or non-offensive content.

Building on prior work (Christian et al., 2025; Jiang et al., 2025), we posit that the underlying reward function is inherently sparse: depending on the input's nature, different evaluative criteria dominate. Leveraging this sparsity through conditional computation is increasingly seen as key to improving specialization, reducing interference, and scaling capacity efficiently (Fedus et al., 2022; Pfeiffer et al., 2024; Shen et al., 2023; Zoph et al., 2022; Du et al., 2022; Shazeer et al., 2017). We therefore adopt a fixed two-expert architecture to exploit this structure:

Let $\mathbf{v}_E \in R^d$ denote the interpretable attribute vector for an explanation E, where each element represents a different explanation feature (e.g., specificity, offensiveness, clarity).

HARM computes a reward score using two expert branches: **(i)** a **positive expert (non-offensive)**, and **(ii)** a **negative expert (offensive)** that accounts for offensive aspects.

As highlight in Figure 2, HARM-MOE-Off is designed in the following way:

(1) Expert Gating. Each expert uses a learned gating matrix to select and scale relevant attributes:

$$\mathbf{g}_s = \sigma(\mathbf{W}_{g,s}\mathbf{v}_E) \odot \mathbf{v}_E, \quad s \in \{\text{pos}, \text{neg}\}$$

where $\mathbf{W}_{g,\text{pos}}, \mathbf{W}_{g,\text{neg}} \in R^{d\times d}$ are learned gating matrices and $\sigma(\cdot)$ is the element-wise sigmoid. The result is a gated attribute vector for each expert.

(2) Expert Scoring. Each expert maps its gated vector to a scalar score using a learned projection:

$$s_{\text{pos}} = \mathbf{W}_{s,\text{pos}}\mathbf{g}_{\text{pos}}, \quad s_{\text{neg}} = \mathbf{W}_{s,\text{neg}}\mathbf{g}_{\text{neg}}$$

where $\mathbf{W}_{s,\text{pos}}, \mathbf{W}_{s,\text{neg}} \in R^{1\times d}$ are learned scoring weights.

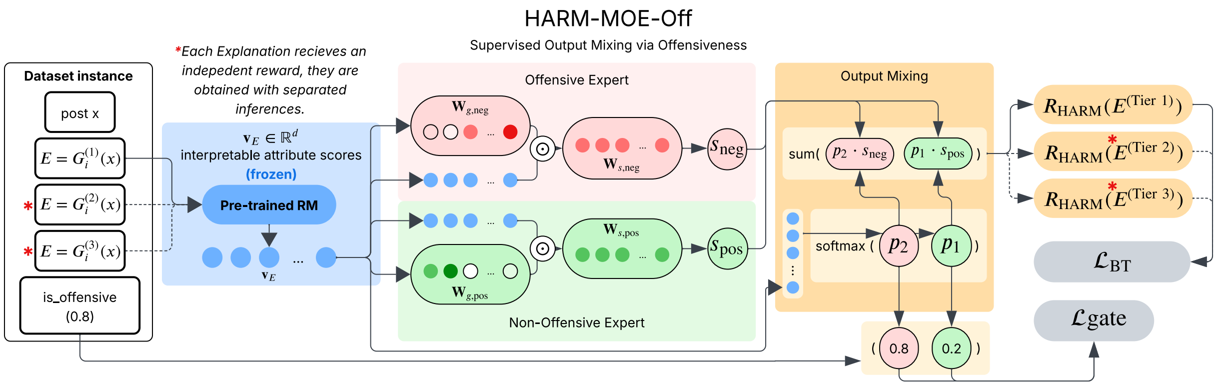

Figure 2: **HARM-MOE-Off:** Architecture inspired by Mixture of Experts with interpretable intermediate attribute re-weighting. The model receives one explanation candidate at a time and re-aggregates pre-trained attribute scores into a offensive-aware scalar reward.

(3) Output Mixing. A softmax over the input vector determines how much each expert contributes:

$$\mathbf{p} = \text{Softmax}(\mathbf{W}_p \mathbf{v}_E), \quad \text{with } \mathbf{p} = [p_1, p_2]$$

where $\mathbf{W}_p \in R^{2\times d}$ is a learned mixing matrix. The weights p_1 and p_2 represent the learned importance of each expert, conditioned on the explanation's attributes.

(4) Final Reward. The final HARM score is a weighted combination of the expert scores:

$$R_{\text{HARM}}(E) = \beta \cdot (p_1 \cdot s_{\text{pos}} + p_2 \cdot s_{\text{neg}})$$

where β is a learned scalar used to calibrate reward scale.

We investigate two specialization mechanisms:

HARM-MOE-Un (Unsupervised Output Mixing): $\mathbf{p}$ is trained using reward prediction alone, letting specialization patterns emerge via gradients (Zoph et al., 2022).

HARM-MOE-Off (Supervised Output Mixing via Offensiveness): an auxiliary MSE loss $\mathcal{L}gate$ aligns $\mathbf{W}_p\mathbf{v}_E$ with SBIC offensiveness labels, improving expert regularization (Zhou et al., 2022).

This approach integrates interpretable modular bias and task-aware supervision, relating to ensemble-inspired uncertainty modeling (Lakshminarayanan et al., 2017) and recent efforts in modular RL and evaluation (Aydın et al., 2025).

3.2 Training via Bradley–Terry Loss

To optimize toward a correct multi-explanation score ranking, we adopt the Bradley–Terry (Bradley and Terry, 1952) framework for pairwise preference learning, following recent advances in reward modeling (Rafailov et al., 2023; Wang et al., 2024). Given two explanations (E_i, E_j), where E_i is preferred over E_j, the model learns to predict this preference through the scoring function R_{HARM}:

$$P(E_i \succ E_j \mid \theta) = \sigma(R_{\text{HARM}}(E_i) - R_{\text{HARM}}(E_j))$$

The main loss function is the Bradley–Terry objective:

$$\mathcal{L}_{\text{BT}} = -\sum_{(i,j)\in\mathcal{P}} \log P(E_i \succ E_j \mid \theta)$$

where $\mathcal{P}$ denotes the set of tiered explanation pairs.

Preference Pair Generation. We construct pairwise preferences by comparing explanation quality across tiers, conditional on the offensiveness of the post:

For **posts marked as offensive** (`offensiveYN` $\geq$ 0.5), stereotype information is crucial. Therefore, we generate full orderings: $E^{(\text{Tier } 3)} \succ E^{(\text{Tier } 2)} \succ E^{(\text{Tier } 1)}$

In these cases of **non-offensive post**, stereotype content is generally irrelevant or absent. The focus shifts to correctly communicating non-hatefulness, so Tier 2 and Tier 3 are considered equally preferred over Tier 1: $E^{(\text{Tier } 3)} \sim E^{(\text{Tier } 2)} \succ E^{(\text{Tier } 1)}$

Total Training Objective. We define six loss components reflecting tiered comparisons between different content types (offensive vs. non-offensive), along with gating supervision for offensiveness:

$\mathcal{L}$**off**$^{i>j}$**:** Bradley Terry (BT) loss comparing Tier j to Tier i for offensive content, where $(i, j) \in (1, 2), (1, 3), (2, 3)$.

$\mathcal{L}$**non**$^{i>j}$**:** BT loss comparing Tier j to Tier i for non-offensive content, where $(i, j) \in (1, 2), (1, 3)$.

$\mathcal{L}$**gate:** mean squared error (MSE) loss for output mixing supervision in the MOE-Off module.

The total loss is a weighted sum: $\mathcal{L}_{\text{total}} = \sum_k \lambda_k \mathcal{L}k$, where each λ_k is a scalar (learned or manually tuned), and $\mathcal{L}k \in \{\mathcal{L}\text{off}^{1>2}, \mathcal{L}\text{off}^{1>3}, \mathcal{L}\text{off}^{2>3}, \mathcal{L}\text{non}^{1>2}, \mathcal{L}\text{non}^{1>3}, \mathcal{L}\text{gate}\}$.

Additional appendices (I, J and K) test the impact of different strategies for losses combination.

4 SBIC-Explain - A Multi-Tiered synthetically generated NLE Dataset.

Create SBIC-Explain: to create a dataset that allows for a controlled study of increasing hate-explicit information in NLE generation, the Social Bias Inference Corpus (SBIC) (Sap et al., 2020) was used as a foundation. For each post x in SBIC, we generate three tiers of explanations using a set of K pretrained language models, $\mathcal{M} = \{M_1, \ldots, M_K\}$. The generation for model M_i is defined as: $G_i^{(t)}(x) = M_i\big(\text{Prompt}(x, \mathcal{C}^{(t)})\big)$, $t \in \{1, 2, 3\}$ where $\mathcal{C}^{(t)}$ is the conditioning set for each tier: $\mathcal{C}^{(1)} = \{\}, \mathcal{C}^{(2)} = \{\text{offensiveYN}\}, \mathcal{C}^{(3)} = \{$ offensiveYN, targetMinority, targetCategory, targetStereotype$\}$. Prompts used to generate inference can be found in Appendix C.

To ensure that differences in explanations are due to the conditioning information and not sampling randomness, we use greedy decoding (i.e., ‘top_k=1‘) for all generations.

We process the official train/dev/test splits of SBIC, selecting only instances that include a stereotype annotation (i.e., with non-null values for `targetMinority`, `targetCategory`, and `targetStereotype`). To ensure a balanced representation of offensive and non-offensive content, we apply a downsampling strategy. Specifically, we retain all offensive instances (`offensiveYN` ≥ 0.5) that contain semantic stereotype information and randomly downsample the non-offensive subset to match this distribution. This rebalancing yields the final **SBIC-Explain** dataset, comprising 30,899 aggregated instances.

To synthesize the dataset, models were selected taking as a reference the top four models of Open LLM Leaderboard under 30B parameters and only official providers, using also their official GGUF quantized versions for efficient inference: **Qwen/Qwen3-14B:** A model by Alibaba Cloud (q_8 GGUF official quantization (Yang et al., 2025) qwen-source). **Microsoft/Phi-4:** A recent state-of-the art 14B parameter model from Microsoft (q_8 GGUF official quantization (Abdin et al., 2024) phi-source). **TIIUAE/Falcon-3-10B-Instruct:** A high-performing instruction-tuned model from the Technology Innovation Institute (q_8 GGUF official quantization (Team, 2024) falcon-source). **Google/Gemma-3-27B-it**: A recent, powerful instruction-tuned model from Google (q4_0 GGUF official quantization (Team et al., 2025) gemma-source).

The final dataset was generated with the mentioned 30,899 unique SBIC posts, each post generated three levels of NLE, and each level of explanation was generated by four models, producing a dataset with a total of **370,788** synthetical NLEs. The final dataset version is grouped per post and per model, so that each instance has 3 NLEs that were produced by model $\mathcal{M}_i$. The final dataset therefore has a total of **123,596** instances.

SBIC-Explain Diversity. To highlight the importance of considering different models, diversity of model-generated content was assessed using **compression ratio** of the concatenated outputs (Shaib et al., 2025). This metric, defined as the ratio of the compressed size (via gzip) to the original uncompressed size, serves as a proxy for output redundancy. *Lower* compression ratios indicate *higher* diversity, as the content contains fewer repeated substrings and patterns. We report each model's sentence length (*Avg Words*) right-trimming sentences to a fixed size of 64 words (considering each word is separated by white space), because `Qwen3` was the model that generated the smaller sentences, with an average of 64 words per sentence, and, as stated by (Shaib et al., 2025), the length of the analyzed text has to be reported alongside all these scores.

Tiers	**1**	**2**	**3**	**Avg**	**Avg Words**
Model					
Gemma3	<u>3.17</u>	<u>3.23</u>	<u>3.34</u>	<u>3.24</u>	63
Falcon3	3.31	3.5	3.85	3.56	63
Qwen3	3.65	**3.94**	4.3	3.96	59
Phi4	**3.66**	3.92	**4.49**	**4.02**	63

Table 1: Compression ratios of generated outputs under increasing task complexity. Lower is more diverse.

From Table 1 results, `Gemma-3` generates the most *diverse*, while `Phi-4` and `Qwen-3` tend to repeat structures or lexical patterns across samples. `Falcon-3` occupies a middle ground, generating content with moderate variability.

This aligns with downstream performance (Appendix I - Table 8) trends from our reward modeling experiments, where HARM trained on data only generated by Falcon, Qwen, Phi, and Gemma achieve progressively lower overall test accuracy, respectively, and training on all models synthetic data generated higher testing accuracy.

Interestingly, although `Gemma-3` explanations are the most diverse, models trained alone on its data generalize the worst. This may suggest that high diversity alone does not ensure effective reward model training. In contrast, `Falcon-3`'s intermediate diversity may offer a more balanced training signal, enabling stronger generalization across unseen model outputs.

5 Experiments and Results

We evaluate **HARM** on our **SBIC-Explain** dataset using **ArmoRM-Llama3-8B-v0.1** (Wang et al., 2024) as the backbone, prompt details are in Appendix D. All models were trained using the Adam optimizer with a learning rate of 2.5e-3. To assess architectural trade-offs, we also test Linear and MLP variants as baselines (Appendix L).

All experiments follow the same accuracy calculation, based on the expected explanation ranking. For **offensive posts** (`offensiveYN` ≥ 0.5), stereotype-relevant tiers should follow: $E^{(3)} \succ E^{(2)} \succ E^{(1)}$. For **non-offensive posts**, Tier 2 and Tier 3 are treated as equally preferred over Tier 1, as discussed in Section 3.2.

5.1 Baseline Misalignment: Off-the-Shelf RMs Fail to Value Context.

Table 2 results confirm a critical failure: both state-of-the-art RMs (ArmoRM-Llama3-8B-v0.1 (Wang et al., 2024)[2] and URM-LLaMa-3.1-8B (Lou et al., 2025)[3]) prefer label-aware (Tier 2) explanations and penalize the inclusion of rich stereotype information (Tier 3). This demonstrates that general-purpose reward functions are not suited for the nuances of this domain and actively discourage the generation of more insightful (but possible more language sensitive) explanations.

[2] armorm-source
[3] urm-source

Baseline Reward Model	Tier	Reward Difference (mean ± std)	Accuracy
ArmoRM (Wang et al., 2024)	2 > 1	0.039 ± 0.1	0.62
	3 > 1	0.005 ± 0.2	0.49
	3 > 2	-0.053 ± 0.1	0.28
URM (Lou et al., 2025)	2 > 1	**0.091 ± 0.6**	0.58
	3 > 1	0.011 ± 0.7	0.44
	3 > 2	-0.136 ± 0.5	0.23
HARM **MOE-Off**	2 > 1	0.07 ± 0.2	**0.73**
	3 > 1	**0.14 ± 0.2**	**0.87**
	3 > 2	**0.09 ± 0.2**	**0.79**

Table 2: Accuracy and Proportional differences ($diff(i,j) = (i-j)/j$) between reward scores from baseline RMs and **HARM** for different tiers.

Misalignment Hypothesis Rational. To better understand how RMs may implicitly disincentivize sensitive or emotionally charged content, we compute Pearson correlation coefficients between model-assigned rewards and external language classifiers targeting hate speech, offensiveness, and sentiment.

We use the following pretrained detectors from `cardiffnlp`: **Sentiment (sent)**: `twitter-roberta-base-sentiment` (Barbieri et al., 2020) source. **Hate speech (hate)**: `twitter-roberta-base-hate` source. **Offensiveness (off)**: `twitter-roberta-base-offensive` source.

Overall, Table 3 shows moderate positive correlations between rewards and non-offensive content (up to $r = 0.45$ in Tier 3 for ArmoRM), suggesting that RMs may indeed favor less offensive or emotionally negative outputs. This aligns with previous findings that reward models often exhibit an implicit bias toward "safe" or sanitized language

Text	Reward Metric	ArmoRM (Wang et al., 2024) Tier 1	Tier 2	Tier 3	URM (Lou et al., 2025) Tier 1	Tier 2	Tier 3
Post	not hate	0.20	0.25	0.24	-0.07	0.07	0.13
	not off	0.36	**0.41**	0.39	-0.01	0.18	**0.20**
	sent (neg)	-0.20	-0.25	-0.24	-0.01	-0.14	-0.17
Tier 1	not hate	0.13	0.18	0.17	-0.07	0.07	0.08
	not off	0.36	**0.44**	0.42	-0.05	0.21	0.25
	sent (neg)	-0.31	-0.37	-0.38	-0.10	-0.27	**-0.34**
Tier 2	not hate	0.13	0.16	0.16	-0.05	0.05	0.10
	not off	0.37	**0.43**	**0.43**	-0.03	0.17	0.27
	sent (neg)	-0.25	-0.36	-0.40	0.05	-0.24	**-0.40**
Tier 3	not hate	0.13	0.17	0.17	-0.05	0.06	0.13
	not off	0.34	0.42	**0.45**	-0.03	0.20	0.37
	sent (neg)	-0.24	-0.35	-0.40	0.05	-0.25	**-0.43**

Table 3: Correlation between different methods reward results and sentiment, offensive and hate measurements.

Tier	Expert	MOE-Off	MOE-Un
1	Offensive	-0.59	-0.48
1	Non-Offensive	0.59	0.48
2	Offensive	-0.78	-0.68
2	Non-Offensive	0.78	0.68
3	Offensive	**-0.82**	**-0.73**
3	Non-Offensive	**0.82**	**0.73**

Table 4: Correlation of model **output mixing** ($W_p \in R^{2\times d}$) and offensive ground truth, across model tiers.

(Christian et al., 2025; Jiang et al., 2025).

Moreover, reward tends to be negatively correlated with negative sentiment scores, especially in more complex explanations, reaching values as low as $r = -0.43$ in Tier 3 (URM). This implies that emotionally charged or confrontational explanations, while potentially necessary in sensitive domains like hate speech, may be under-rewarded. These trends highlight the presence of systematic bias in reward modeling, further motivating the development of HARM's multi-expert mechanism.

To further investigate, we manually sanitized offensive explanations, removing references to stereotypes and bias, and compared their reward scores using ArmoRM (Appendix N). On average, sanitized explanations received **37% higher rewards**, supporting the hypothesis that current RMs undervalue faithful, socially grounded content.

5.2 HARM Performance: Correcting the Misalignment

We evaluate the explanations using our proposed **HARM-MoE-Off**. As shown in Table 2, HARM successfully recovers the expected hierarchy of explanation quality, assigning the highest rewards to Tier 3, demonstrating the relevance of re-weighting existing interpretable reward attributes. Notably, attributes like coherence and complexity receive higher weights, reflecting the importance of consistency and reasoning depth. A full analysis of learned weights is provided in Appendix F, with a dedicated discussion on text complexity in Appendix G, and practical examples in Appendix N.

Output Mixing: Offensive Post Prediction Accuracy. To evaluate whether supervised output mixing improves alignment between experts and offensive content, we analyzed the correlation between gating values and the SBIC `is_offensive` continuous annotations on the test set.

As shown in Table 4, the output mixing weights, particularly those selecting between the non-offensive and offensive experts, show strong and systematic correlations with the offensiveness signal. This alignment is further amplified in the HARM-MOE-Off setting, where gating is explicitly supervised using `is_offensive` annotations.

Despite receiving no explicit guidance, the unsupervised variant (HARM-MOE-Un) naturally develops gating behaviors aligned with offensive content, indicating that the emergence of expert specialization along offensive/non-offensive dimensions is an inductive bias reinforced by the task and data. Notably, the supervised model (HARM-MOE-Off) further amplifies this effect, achieving, on average, a 15% improvement in expert correlation with offensive content, demonstrating its ability to disentangle semantically meaningful behaviors more clearly and robustly. These findings highlight the interpretability of our output-gated architecture and point toward the broader potential of supervised and hybrid approaches for learning structured, disentangled representations of social biases and linguistic toxicity in a modular fashion.

6 Conclusion

We addressed a critical gap in the evaluation of Natural Language Explanations (NLEs) for hate speech by revealing how current reward models (RMs) often penalize contextually appropriate explanations, thereby misaligning with the goal of faithful and socially grounded explanation.

To tackle this, we introduced **HARM**, a Hate-Aware Reward Model that reweights interpretable attributes to prioritize domain-relevant explanatory quality. HARM not only recovers the intended hierarchy across explanation tiers but also provides fine-grained interpretability, making it a practical tool for sensitive evaluation tasks. To support this effort, we also presented **SBIC-Explain**, a dataset of 370,788 LLM-generated explanations for offensive content, annotated across three tiers of contextual richness: Tier 1 (text-only), Tier 2 (+ classification-aware), and Tier 3 (+ semantics-informed).

Although our work is grounded in the domain of hate speech and offensive content, the problem of misaligned evaluation extends to other linguistically sensitive settings (e.g. legal (Chalkidis et al., 2021) and medical (Agrawal et al., 2022; Han et al., 2025)). Our work aim to serve as a benchmark and inspiration for developing more faithful systems in sensitive contexts.

Limitations

Our approach has some limitations. First, while HARM improves alignment in hate speech/offensive contexts, its performance and learned weights depend on the choice of backbone reward model (e.g., ArmoRM), which itself may carry inherent biases from training data. Second, our evaluation relies on synthetic explanations (based on human rationals) rather than entirely human-authored ones, which may not capture the full variability or nuance of real-world reasoning. Third, the potential applicability to other domains such as law or medicine remains hypothetical, we encourage empirical validation before generalization. Finally, HARM, as it is, was mainly designed and tested for post hoc evaluation and does not directly influence generation during model training; integrating it (or some variation of general purpose interpretable reward scores re-weighting) into reinforcement learning or DPO-style loops is a possible direction for future work.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. Preprint, arXiv:2412.08905.

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. Preprint, arXiv:2205.12689.

Muhammad Umair Arshad and Waseem Shahzad. 2024. Understanding hate speech: the HateInsights dataset and model interpretability. 10:e2372.

Hüseyin Aydın, Kevin Godin-Dubois, Libio Goncalvez Braz, Floris den Hengst, Kim Baraka, Mustafa Mert Çelikok, Andreas Sauter, Shihan Wang, and Frans A. Oliehoek. 2025. Sharpie: A modular framework for reinforcement learning and human-ai interaction experiments. Preprint, arXiv:2501.19245.

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, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. Preprint, arXiv:2204.05862.

Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. 2020. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. Preprint, arXiv:2010.12421.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika, 39:324.

Faeze Brahman, Sachin Kumar, Vidhisha Balachandran, Pradeep Dasigi, Valentina Pyatkin, Abhilasha Ravichander, Sarah Wiegreffe, Nouha Dziri, Khyathi Chandu, Jack Hessel, Yulia Tsvetkov, Noah A. Smith, Yejin Choi, and Hannaneh Hajishirzi. 2024. The art of saying no: Contextual noncompliance in language models. Preprint, arXiv:2407.12043.

Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos. 2021. MultiEURLEX - a multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6974–6996, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lichang Chen, Chen Zhu, Davit Soselia, Jiuhai Chen, Tianyi Zhou, Tom Goldstein, Heng Huang, Mohammad Shoeybi, and Bryan Catanzaro. 2024a. Odin: Disentangled reward mitigates hacking in rlhf. Preprint, arXiv:2402.07319.

Yanjun Chen, Dawei Zhu, Yirong Sun, Xinghao Chen, Wei Zhang, and Xiaoyu Shen. 2024b. The accuracy paradox in RLHF: When better reward models don't yield better language models. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 2980–2989, Miami, Florida, USA. Association for Computational Linguistics.

Brian Christian, Hannah Rose Kirk, Jessica A.F. Thompson, Christopher Summerfield, and Tsvetomira Dumbalska. 2025. Reward model interpretability via optimal and pessimal tokens. In Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency, FAccT '25, page 1048–1059. ACM.

Jaymari Chua, Yun Li, Shiyi Yang, Chen Wang, and Lina Yao. 2024. Ai safety in generative ai large language models: A survey. Preprint, arXiv:2407.18369.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. Ultrafeedback: Boosting language models with scaled ai feedback. Preprint, arXiv:2310.01377.

Zahra Delbari, Nafise Sadat Moosavi, and Mohammad Taher Pilehvar. 2024. Spanning the spectrum of hatred detection: A persian multi-label hate speech dataset with annotator rationales. Proceedings of the AAAI Conference on Artificial Intelligence, 38(16):17889–17897.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, and 8 others. 2022. Glam: Efficient scaling of language models with mixture-of-experts. Preprint, arXiv:2112.06905.

Erfan Entezami and Ali Naseh. 2025. Llm misalignment via adversarial rlhf platforms. Preprint, arXiv:2503.03039.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. J. Mach. Learn. Res., 23(1).

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. ACM Comput. Surv., 51(4).

Cesaire J. K. Fouodo, Lea L. Kronziel, Inke R. König, and Silke Szymczak. 2025. Effect of hyperparameters on variable selection in random forests. Preprint, arXiv:2309.06943.

Tianyu Han, Lisa C. Adams, Jens-Michalis Papaioannou, Paul Grundmann, Tom Oberhauser, Alexei Figueroa, Alexander Löser, Daniel Truhn, and Keno K. Bressem. 2025. Medalpaca – an open-source collection of medical conversational ai models and training data. Preprint, arXiv:2304.08247.

Phu Gia Hoang, Canh Duc Luu, Khanh Quoc Tran, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2023. ViHOS: Hate speech spans detection for Vietnamese. In Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 652–669, Dubrovnik, Croatia. Association for Computational Linguistics.

Fan Huang, Haewoon Kwak, and Jisun An. 2023. Chain of explanation: New prompting method to generate quality natural language explanation for implicit hate speech. In Companion Proceedings of the ACM Web Conference 2023, WWW '23 Companion, page 90–93, New York, NY, USA. Association for Computing Machinery.

Fan Huang, Haewoon Kwak, Kunwoo Park, and Jisun An. 2024. Chatgpt rates natural language explanation quality like humans: But on which scales? Preprint, arXiv:2403.17368.

Md Saroar Jahan and Mourad Oussalah. 2021. A systematic review of hate speech automatic detection using natural language processing. Preprint, arXiv:2106.00742.

Junqi Jiang, Tom Bewley, Saumitra Mishra, Freddy Lecue, and Manuela Veloso. 2025. Interpreting language reward models via contrastive explanations. In The Thirteenth International Conference on Learning Representations.

Masahiro Kaneko and Danushka Bollegala. 2021. Unmasking the mask – evaluating social biases in masked language models. Preprint, arXiv:2104.07496.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. Preprint, arXiv:1705.07115.

Youngwook Kim, Shinwoo Park, Youngsoo Namgoong, and Yo-Sub Han. 2023. ConPrompt: Pre-training a language model with machine-generated data for implicit hate speech detection. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 10964–10980, Singapore. Association for Computational Linguistics.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, page 6405–6416, Red Hook, NY, USA. Curran Associates Inc.

Nathan Lambert, Louis Castricato, Leandro von Werra, and Alex Havrilla. 2024a. Rewardbench: Evaluating reward models for language modeling. arXiv preprint arXiv:2403.13787.

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. 2024b. Rewardbench: Evaluating reward models for language modeling. Preprint, arXiv:2403.13787.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Xingzhou Lou, Dong Yan, Wei Shen, Yuzi Yan, Jian Xie, and Junge Zhang. 2025. Uncertainty-aware reward model: Teaching reward models to know what is unknown. Preprint, arXiv:2410.00847.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2024. Towards faithful model explanation in NLP: A survey. Computational Linguistics, 50(2):657–723.

Jitendra Singh Malik, Hezhe Qiao, Guansong Pang, and Anton van den Hengel. 2023. Deep learning for hate speech detection: A comparative study. Preprint, arXiv:2202.09517.

Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah A. Smith, Hannaneh Hajishirzi, and Nathan Lambert. 2025. Rewardbench 2: Advancing reward model evaluation. Preprint, arXiv:2506.01937.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2022. Hatexplain: A benchmark dataset for explainable hate speech detection. Preprint, arXiv:2012.10289.

Rachna Narula and Poonam Chaudhary. 2024. A comprehensive review on detection of hate speech for multi-lingual data. Social Network Analysis and Mining, 14(1):244.

Huy Nghiem and Hal Daumé Iii. 2024. HateCOT: An explanation-enhanced dataset for generalizable offensive speech detection via large language models. In Findings of the Association for Computational Linguistics: EMNLP 2024, pages 5938–5956, Miami, Florida, USA. Association for Computational Linguistics.

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. 2022. Training language models to follow instructions with human feedback. Preprint, arXiv:2203.02155.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and 2 others. 2019. Pytorch: An imperative style, high-performance deep learning library. Preprint, arXiv:1912.01703.

John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. 2021. SemEval-2021 task 5: Toxic spans detection. In Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 59–69, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Sebastian Ruder, Ivan Vulić, and Edoardo Maria Ponti. 2024. Modular deep learning. Preprint, arXiv:2302.11529.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In Advances in Neural Information Processing Systems, volume 36, pages 53728–53741. Curran Associates, Inc.

Manikandan Ravikiran and Subbiah Annamalai. 2021. DOSA: Dravidian code-mixed offensive span identification dataset. In Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, pages 10–17, Kyiv. Association for Computational Linguistics.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. 2024. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. Preprint, arXiv:2308.01263.

Ananya B. Sai, Akash Kumar Mohankumar, and Mitesh M. Khapra. 2020. A survey of evaluation metrics used for nlg systems. Preprint, arXiv:2008.12009.

Isadora Salles, Francielle Vargas, and Fabrício Benevenuto. 2025. HateBRXplain: A benchmark dataset with human-annotated rationales for explainable hate speech detection in Brazilian Portuguese. In Proceedings of the 31st International Conference on Computational Linguistics, pages 6659–6669, Abu Dhabi, UAE. Association for Computational Linguistics.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. 2020. Social bias frames: Reasoning about social and power implications of language. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 5477–5490, Online. Association for Computational Linguistics.

Chantal Shaib, Joe Barrow, Jiuding Sun, Alexa F. Siu, Byron C. Wallace, and Ani Nenkova. 2025. Standardizing the measurement of text diversity: A tool and a comparative analysis of scores. Preprint, arXiv:2403.00553.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In International Conference on Learning Representations.

Sheng Shen, Le Hou, Yanqi Zhou, Nan Du, Shayne Longpre, Jason Wei, Hyung Won Chung, Barret Zoph, William Fedus, Xinyun Chen, Tu Vu, Yuexin Wu, Wuyang Chen, Albert Webson, Yunxuan Li, Vincent Zhao, Hongkun Yu, Kurt Keutzer, Trevor Darrell, and Denny Zhou. 2023. Mixture-of-experts meets instruction tuning:a winning combination for large language models. Preprint, arXiv:2305.14705.

Tianxiang Sun, Junliang He, Xipeng Qiu, and Xuanjing Huang. 2022. BERTScore is unfair: On social bias in language model-based metrics for text generation. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 3726–3739, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Falcon-LLM Team. 2024. The falcon 3 family of open models.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. Preprint, arXiv:2503.19786.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In EMNLP.

Yuxia Wang, Haonan Li, Xudong Han, Preslav Nakov, and Timothy Baldwin. 2023a. Do-not-answer: A dataset for evaluating safeguards in llms. Preprint, arXiv:2308.13387.

Zhilin Wang, Yi Dong, Jiaqi Zeng, Virginia Adams, Makesh Narsimhan Sreedhar, Daniel Egert, Olivier Delalleau, Jane Polak Scowcroft, Neel Kant, Aidan Swope, and Oleksii Kuchaiev. 2023b. Helpsteer: Multi-attribute helpfulness dataset for steerlm. Preprint, arXiv:2311.09528.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. Preprint, arXiv:2505.09388.

Yue Yu, Zhengxing Chen, Aston Zhang, Liang Tan, Chenguang Zhu, Richard Yuanzhe Pang, Yundi Qian, Xuewei Wang, Suchin Gururangan, Chao Zhang, Melanie Kambadur, Dhruv Mahajan, and Rui Hou. 2025. Self-generated critiques boost reward modeling for language models. In Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 11499–11514, Albuquerque, New Mexico. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. Preprint, arXiv:1904.09675.

Xianbing Zhou, Yang Yong, Xiaochao Fan, Ge Ren, Yunfeng Song, Yufeng Diao, Liang Yang, and Hongfei Lin. 2021. Hate speech detection based on sentiment knowledge sharing. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 7158–7166, Online. Association for Computational Linguistics.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y. Zhao, Andrew Dai, Zhifeng Chen, Quoc Le, and James Laudon. 2022. Mixture-of-experts with expert choice routing. In Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences. Preprint, arXiv:1909.08593.

Barret Zoph, Irwan Bello, Sameer Kumar, Nan Du, Yanping Huang, Jeff Dean, Noam Shazeer, and William Fedus. 2022. St-moe: Designing stable and transferable sparse expert models. Preprint, arXiv:2202.08906.

A Licenses

All third-party models and datasets used in this work are governed by their respective licenses:

- **Qwen/Qwen3-14B:** Apache License 2.0.
- **Microsoft/Phi-4:** MIT License.
- **TIIUAE/Falcon-3-10B-Instruct:** TII Falcon-LLM License 2.0, which is based on Apache License 2.0 with additional terms and restrictions.
- **Google/Gemma-3-27B-it:** Gemma License. Notably, this license prohibits generation of sexually explicit content except for scientific, educational, documentary, or artistic purposes.
- **SBIC Dataset:** CC BY 4.0 License.

Our use of these artifacts complies with all stated licenses, particularly by ensuring that generated content is used solely for scientific research and not redistributed beyond the scope allowed by each license.

B Software

Our experiments were implemented in Python, using PyTorch (Paszke et al., 2019) and HuggingFace's Transformers library (Wolf et al., 2020) for model training and inference. For model quantized (GGUF) inference, we used `llama_cpp`[4]. To analyze explanation quality and diversity, we employed the `diversity`[5] package and `textstat`[6] (Shaib et al., 2025), which provide metrics for structural repetition and readability assessment, respectively.

[4] llama_cpp source
[5] diversity source
[6] textstat source

C Prompt Templates For Dataset Generation

Here we detail the exact prompt templates used for generating the NLEs for each tier. Different models had slight prompt variations to accommodate for different expected prompt structures (e.g. models that expect explicit reasoning tokens or slight variation in system/user/assistant chat tokens to specify who is talking), always following each models best practices.

C.1 Tier 1 Prompt

```
<|im_start|>user
========= INFO TO FOLLOW WHEN GENERATING ANSWER
Post Text: {post}
========= INFO TO FOLLOW WHEN GENERATING ANSWER<|im_end|>

<|im_start|>user
Generate a natural language explanation of why this is or is not to be considerate offensive.
Generate the most concise explanation version possible while still explaining all important aspects.
You dont need to rewrite the whole sentence to explain it, but you can rewrite or reference specif parts to help on your explanation.<|im_end|>
Write your explaination in english.
<|im_end|>

<|im_start|>assistant /no_think

<think>

</think>

Explanation:
```

C.2 Tier 2 Prompt

```
<|im_start|>user
========= INFO TO FOLLOW WHEN GENERATING ANSWER
Post Text: {post}
{'POST TEXT IS CONSIDERED OFFENSIVE' if is_offensive>=0.5 else 'POST TEXT IS NOT CONSIDERED OFFENSIVE'}
========= INFO TO FOLLOW WHEN GENERATING ANSWER<|im_end|>

<|im_start|>user
Generate a natural language explanation of why this is or is not to be considerate offensive.
Generate the most concise explanation version possible while still explaining all important aspects.
You dont need to rewrite the whole sentence to explain it, but you can rewrite or reference specif parts to help on your explanation.<|im_end|>
```

```
Write your explaination in english.
<|im_end|>

<|im_start|>assistant /no_think

<think>

</think>

Explanation:
```

C.3 Tier 3 Prompt

```
<|im_start|>user
========= INFO TO FOLLOW WHEN GENERATING
     ANSWER
Post Text: {post}
{'POST TEXT IS CONSIDERED OFFENSIVE' if
    is_offensive>=0.5 else 'POST TEXT IS
     NOT CONSIDERED OFFENSIVE'}
Targeted Minority: {targetMinority}
Targeted Category: {targetCategory}
Targeted Stereotype: {targetStereotype}
========= INFO TO FOLLOW WHEN GENERATING
     ANSWER<|im_end|>

<|im_start|>user
Generate a natural language explanation
    of why this is or is not to be
    considerate offensive.
Generate the most concise explanation
    version possible while still
    explaining all important aspects.
You dont need to rewrite the whole
    sentence to explain it, but you can
    rewrite or reference specif parts to
     help on your explanation.<|im_end|>
Write your explaination in english.
<|im_end|>

<|im_start|>assistant /no_think

<think>

</think>

Explanation:
```

D Templates For ArmoRM Generation and URM

Here we detail the exact template used generating Rewards using ArmoRM and URM.

D.1 ArmoRM and URM Template

```
[{
"role": "user", "content": Generate a
    natural language explanation of why
    this post text is or is not
    considered offensive.
Post text: {<post_text>}
},
{"role": "assistant", "content": {<
    ai_generate_NLE>}
}]
```

E Explainable Hate Speech Datasets

Table 5 shows that most explainable hate speech datasets center around highlight-based rationales, where annotators mark specific words or spans perceived as offensive. This design, seen in datasets like HateXplain, SemEval-2021, and HateBRXplain, prioritizes the surface detection of hateful expressions. However, it often neglects the deeper reasoning or hateful rationale, the underlying social stereotypes, historical marginalization, or implied harm, behind why a phrase is offensive.

HateCOT introduces a promising step forward by generating synthetic natural language explanations (NLEs) that leverage broader context. However, its approach directly injects all available metadata (e.g., target labels, hate categories, spans) into the prompt, leaving open the question of how each type of information impacts explanation quality. This conflation makes it difficult to isolate which elements meaningfully improve explanation coherence or alignment with human values.

In contrast, our work proposes a more structured investigation. By focusing on the semantic contribution of stereotype-informed free-text rationales, specifically, the “implied statements” from SBIC, we aim to disentangle how this deeper contextual information affects model-generated explanations. Moreover, we go beyond human-written ground truth by evaluating how these enriched explanations are perceived by reward models, offering insights into both performance and alignment.

Dataset	Year	Language(s)	Source Domain	Explainability	Annotation	Size
HateBRXplain (Salles et al., 2025)	2025	Portuguese (Brazilian)	Instagram (comments)	Span rationales (text)	Expert annotators	7,000
HateInsights (Arshad and Shahzad, 2024)	2024	Urdu (Roman Arabic)	Twitter, Facebook	Span rationales (word/sentence)	Student annotators	11,782
HateCOT (Nghiem and Daumé Iii, 2024)	2024	English	Multi $\approx$ source (8 corpora)	Synthetic NLEs	GPT-3.5-Turbo + human curation	52,137
PHate (Delbari et al., 2024)	2024	Persian (Farsi)	Twitter	Span rationales + target labels	Expert annotators	7,000
ViHOS (Hoang et al., 2023)	2023	Vietnamese	Facebook, YouTube	Span rationales (text)	Human annotators	11,056
HateXplain (Mathew et al., 2022)	2020	English	Twitter, Gab	Span rationales (text)	AMT crowd-workers	20,148
SemEval-2021 Task 5 (Toxic Spans) (Pavlopoulos et al., 2021)	2021	English	Civil Comments (Wikipedia talk)	Span rationales (text)	Crowd workers	10,629
DOSA (Ravikiran and Annamalai, 2021)	2021	Tamil English, Kannada English	YouTube comments	Span rationales (words)	Human annotators	4,786 (Tamil) / 1,097 (Kannada)
Social Bias Frames (SBIC) (Sap et al., 2020)	2020	English	Twitter, Reddit, Gab, extremist forums	Categorical labels + free-text "implied statements"	Crowdsourced (MTurk)	44,671 posts

Table 5: Overview of explainable hate speech detection datasets.

F Analyzing HARM's Weights

One advantage of our Mixture-of-Experts formulation is interpretability: HARM explicitly exposes gating and scoring weights across expert dimensions. We compute feature importance using a composite of three components: **(i)** input gating weights, **(ii)** expert scoring weights, and **(iii)** output mixing weights. These are analyzed separately for offensive and non-offensive content, then averaged into two final metrics: **Avg Scoring Importance** (mean of **i** and **ii**) and **Avg Output mixing Importance** (mean of **iii**).

Figure 3 illustrates how individual attributes shape HARM's expert behavior. Features like helpsteer-coherence, helpsteer-complexity, and helpsteer-helpfulness are consistently weighted across both offensive and non-offensive settings, reflecting the importance of coherence, clarity, and reasoning depth in generating nuanced hate speech explanations. Given the prominence of complexity-related attributes (helpsteer-complexity, code-complexity), we include a dedicated analysis in Appendix G.

In contrast, lower weights for attributes such as argilla-overall_quality, ultrafeedback-helpfulness, and code-style suggest that superficial fluency or stylistic quality is deprioritized. HARM instead favors factual and context-sensitive reasoning, especially in high-risk cases.

Safety-relevant features (beavertails-is_safe, helpsteer-correctness, ultrafeedback-truthfulness) receive stronger weights in offensive content processing, indicating an adaptive bias toward precision and caution.

The *Avg weights* row in Figure 3 further supports this: offensive input gating scores higher (0.42 vs. 0.30), with more polarized output gating (-0.08 vs. 0.06). This asymmetry shows that HARM becomes more selective and safety-aware when handling harmful inputs.

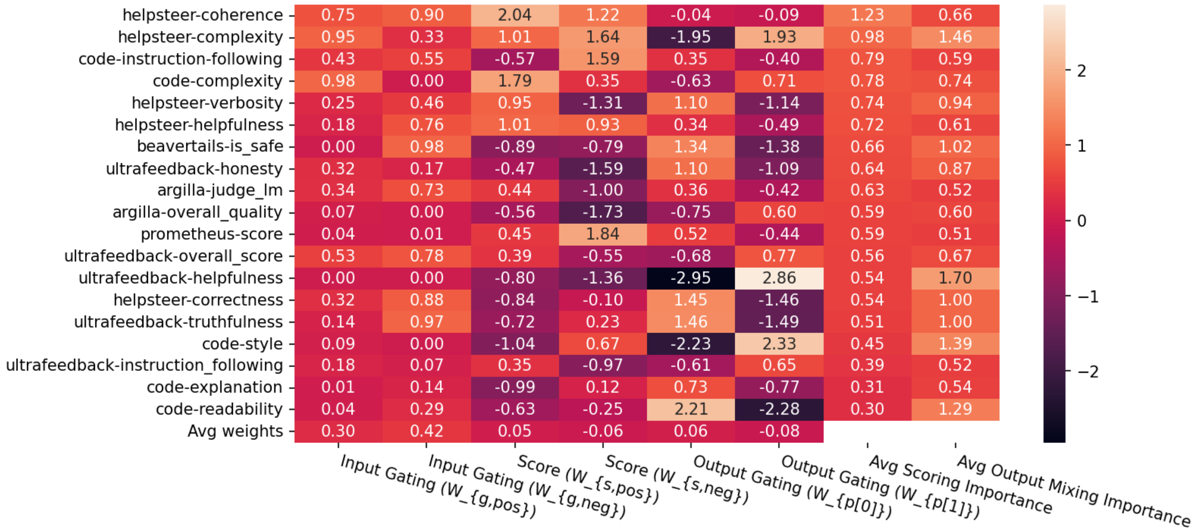

Figure 3: The learned weights of the best **HARM-MOE-Off** model for each of the 19 attributes from ArmoRM. This reveals the relative importance of each attribute for scoring hate speech NLEs.

G Text Complexity Metrics

text **Metric**	**Post**	**Tier 1**	**Tier 2**	**Tier 3**
flesch_reading_ease ↓	72.998	29.349	29.154	**25.107**
mcalpine_eflaw ↑	20.526	**29.12**	28.495	28.402
gunning_fog ↑	8.689	17.782	17.668	**18.37**
smog_index ↑	8.196	15.924	15.797	**16.276**
automated_readability_index ↑	7.31	15.323	15.204	**15.673**
linsear_write_formula ↑	7.748	14.824	14.813	**15.327**
flesch_kincaid_grade ↑	6.667	14.392	14.329	**14.953**
coleman_liau_index ↑	7.114	14.089	14.139	**14.609**
dale_chall_readability_score ↑	8.956	11.27	11.3	**11.372**
spache_readability ↑	4.46	7.035	6.997	**7.115**

Table 6: Complexity metrics (readability, complexity, and grade level). ↑ and ↓ indicate the relationship of the given metric and increase in complexity.

To further investigate the high importance of complexity-related attributes observed in HARM's learned weights, particularly helpsteer-complexity and code-complexity, we analyze textual complexity across different input and explanation types in our dataset.

We apply a suite of standard readability and complexity metrics using the `textstat` Python package[7]. These include traditional readability scores (e.g., Flesch Reading Ease), grade-level indicators (e.g., Gunning Fog Index, SMOG), and composite formulas (e.g., McAlpine EFLAW). Results are presented in Table 6, comparing complexity across the original post texts and the three tiers of generated explanations.

We observe a clear trend on Table 6: explanations, especially those from Tier 3, consistently exhibit higher complexity than the original posts or lower-tier explanations. For example, Tier 3 explanations have the lowest Flesch Reading Ease (25.1, lower is more complex) and the highest Gunning Fog Index (18.4, higher is more complex), indicating more sophisticated and cognitively demanding language. This aligns with our hypothesis that more contextually informed explanations require greater linguistic and conceptual complexity.

These findings support the observed model behavior: attributes related to textual complexity receive higher weighting during HARM's scoring process, likely because they capture deeper reasoning and nuance, critical for high-quality hate speech explanation.

[7] https://pypi.org/project/textstat/

H MLP Hidden Variation Ablation

To further understand baselines beyond our MOE inspired architecture and the impact of model capacity on HARM's performance, we conduct testes a non-linear two-layer multi-layer perceptron (MLP) with ReLU activations. We evaluate five hidden size configurations ranging from 128 to 2048 hidden units, analyzing how architectural choices affect both average performance and stability across multiple training runs.

Table 7 presents statistics for each hidden layer configuration. The results reveal relatively stable performance across different hidden sizes, with all configurations achieving mean accuracies between 0.70 and 0.71.

Hidden Size	Mean	Std	Min	50%	Max
128	0.70	0.07	**0.61**	0.70	0.80
256	**0.71**	**0.08**	0.59	**0.72**	**0.81**
512	0.70	0.07	0.59	0.70	0.78
1024	0.70	0.07	0.60	**0.72**	0.79
2048	0.71	0.07	0.59	0.71	0.80

Table 7: MLP hidden layer size impact on accuracy. (Train on all tiers and all models)

The 256-dimensional hidden layer configuration achieves the highest mean accuracy (0.71) and maximum performance (0.81), though it also exhibits the highest variance (0.08). This suggests that while the 256-unit configuration can achieve peak performance, it may be more sensitive to initialization and training dynamics. The 128-unit configuration provides a good balance between performance and stability, matching the mean accuracy of larger configurations while maintaining reasonable variance.

Interestingly, increasing hidden layer size beyond 256 units does not yield consistent improvements in mean performance. The 512, 1024, and 2048-unit configurations all achieve mean accuracies of 0.70-0.71, with the largest model (2048 units) showing similar performance to the smallest (128 units). This plateau effect suggests that the complexity of the hate speech explanation evaluation task may not require extensive model capacity, and that the representation learning challenges are more related to the quality of the re-weighting strategy rather than raw parameter count.

The consistency in minimum performance across configurations (0.59-0.61) indicates that all architectures are capable of learning meaningful representations, while the similar maximum performance values (0.78-0.81) suggest that the upper bound of performance is more constrained by the task complexity and dataset characteristics than by model capacity. These findings support the use of moderately sized hidden layers (128-256 units) for HARM, providing computational efficiency without sacrificing performance.

I MOE-Off - Losses Combinations (Losses Included or Excluded)

In this section, we investigate the effect of selectively including or excluding different training tiers when optimizing the MOE-Off reward model. Our goal is to understand how various data configurations contribute to generalization and performance across models, and which combinations offer the most effective supervision signal for the intended reward modeling objectives.

Table 8 presents pairwise accuracies, general accuracy, and reward score distance across multiple tier combinations and backbone models. We compare these configurations against our reference models (ArmoRM and URM), and also explore the impact of using all available data versus subsets grouped by tier.

Notably, training on Tiers 1 & 3 consistently yields the strongest results, achieving the highest overall accuracy (**0.807**) and the best average score distance (**0.111**). This indicates that the contrast between these two extremes (Tier 1 (low preference) and Tier 3 (high preference)) provides the clearest and most informative supervision signal. In contrast, combinations involving only Tiers 2 & 3 or Tiers 1 & 2 exhibit more moderate performance, likely due to the reduced contrast in reward preference, which makes learning signal less distinct.

Among backbone models, the **All** ensemble consistently ranks highest or very close to best, highlighting the benefit of incorporating diverse model perspectives during training. This supports our hypothesis that model diversity helps generalize the reward function across different architectures.

Interestingly, although Gemma and Qwen achieve strong results in some configurations, they also show higher variance, particularly Qwen, which exhibits significant volatility in score distance, likely due to instability in reward scaling. These findings suggest that while individual backbones can be powerful, their behavior must be care-

Model Configuration		Pairwise Accuracy				Reward Score Distance ($diff(i,j) = (i-j)/j$)			
Train Tiers	Train Model	2>1	3>1	3>2	Mean	2-1	3-1	3-2	Mean
Ref. Results	ArmoRM	0.616	0.489	0.282	0.524	0.039	0.005	-0.053	0.016
	URM	0.585	0.445	0.229	0.49	0.091	0.011	-0.136	0.038
Tiers 1 & 2	Falcon	0.747	0.67	0.385	0.669	0.335	0.318	-0.008	0.285
	Gemma	0.725	0.687	0.42	0.676	2.761	3.172	-0.283	2.883
	Phi	0.719	0.697	0.436	0.674	0.094	0.106	-0.008	0.09
	Qwen	0.743	0.755	0.537	0.718	0.131	0.164	0.027	0.137
	All	**0.779**	0.745	0.454	0.721	0.043	0.045	0.001	0.04
Tiers 1 & 3	Falcon	0.717	0.816	0.747	0.768	0.036	0.059	0.035	0.048
	Gemma	0.669	0.735	0.658	0.709	0.027	0.037	0.017	0.033
	Phi	0.644	0.816	0.8	0.743	0.03	0.098	0.081	0.067
	Qwen	0.655	0.821	0.815	0.756	0.019	0.052	0.042	0.038
	All	0.729	**0.868**	0.792	**0.807**	**0.073**	0.144	0.092	**0.111**
Tiers 2 & 3	Falcon	0.572	0.787	0.827	0.697	0.007	0.044	0.049	0.028
	Gemma	0.345	0.492	0.75	0.448	-0.044	0.006	0.084	-0.01
	Phi	0.479	0.738	0.847	0.632	-0.019	0.084	0.136	0.044
	Qwen	0.551	0.767	0.84	0.686	0.003	0.04	0.047	0.025
	All	0.495	0.763	**0.86**	0.655	-0.039	**0.152**	**0.203**	0.07
All Tiers	Falcon	0.687	0.813	0.76	0.75	0.068	0.196	0.2	0.142
	Gemma	0.575	0.707	0.744	0.654	0.006	0.02	0.026	0.014
	Phi	0.625	0.807	0.801	0.724	0.019	0.068	0.05	0.044
	Qwen	0.639	0.804	0.816	0.736	0.01	0.042	0.031	0.023
	All	0.689	0.849	0.808	0.775	0.03	0.083	0.069	0.059

Table 8: Performance of **HARM-MOE-Off** reward model **across different training configurations**. **All reported results are based on the entire test set**, only training is being altered.

fully calibrated when used in isolation.

Overall, this analysis reinforces two key insights: (1) contrastive supervision from clearly distinguishable preference tiers (especially Tiers 1 and 3) is critical for effective reward modeling, and (2) incorporating multiple models helps stabilize and improve reward quality across the board.

J MOE-Off - Losses Combination (Grid Search 0, 0.5, 1.)

Loss Component	Importance	Correlation
$\mathcal{L}\text{non}^{1>3}$	0.685	0.761
$\mathcal{L}\text{off}^{1>2}$	0.106	0.129
$\mathcal{L}\text{off}^{1>3}$	0.076	0.186
$\mathcal{L}\text{off}^{2>3}$	0.052	0.080
$\mathcal{L}\text{non}^{1>2}$	0.044	-0.128
$\mathcal{L}\text{gate}$	0.037	0.019

Table 9: Feature importance and correlation analysis for MOE-Off loss weight parameters

To optimize the loss weighting strategy for our MOE-Off architecture, we conduct a comprehensive grid search across all loss components, system-

Test Acc.	$\mathcal{L}$**off**$^{1>2}$	$\mathcal{L}$**off**$^{1>3}$	$\mathcal{L}$**off**$^{2>3}$	$\mathcal{L}$**non**$^{1>2}$	$\mathcal{L}$**non**$^{1>3}$	$\mathcal{L}$**gate**
0.8061	0	0.5	0	0	1	1
0.8060	0	0.5	0	0	0.5	0.5
0.8056	0	1	0	0	1	1
0.8054	0	0.5	0	0	0.5	1
0.8049	0	1	0	0	1	0
0.8045	0	1	0	0	0.5	0.5
0.8043	1	0	0.5	0	0.5	0.5
0.8040	0.5	0.5	0	0	1	1
0.8030	0.5	0.5	0.5	0	1	1
0.8024	0.5	0	0.5	0	0.5	0.5

Table 10: Top 10 **HARM-MOE-Off** configurations of losses (different weights for each loss) ranked by test accuracy.

atically varying each weight parameter between 0, 0.5, and 1.0. This exploration allows us to identify the optimal combination of loss terms and understand the relative importance of different training objectives in our hate-aware reward modeling framework.

We employ a feature importance analysis technique inspired by (Fouodo et al., 2025), where we train a random forest model using hyperparameter configurations as inputs and place **accuracy** as target outputs. The random forest's feature importance values reveal which loss weights contribute most significantly to model performance, while correlation analysis shows the direction and strength of these relationships.

Table 9 presents the feature importance and correlation analysis for each loss weight parameter. $\mathcal{L}$**non**$^{1>3}$ emerges as the most critical parameter, achieving a feature importance of 0.685 and a strong positive correlation of 0.761 with model performance, indicating that non-offensive content discrimination between models 1 and 3 is fundamental to achieving high accuracy. Notably, $\mathcal{L}$**non**$^{1>2}$ exhibits the only negative correlation (-0.128), suggesting that increasing this weight may actually hurt performance.

The top-performing configurations from our grid search validate these importance rankings, as shown in Table 10. The best model achieves 80.61% test accuracy with $\lambda = 1$ for ($\mathcal{L}\text{non}^{1>3}$) (maximizing the most important parameter) while setting the negatively correlated $\lambda = 0$ for ($\mathcal{L}\text{non}^{1>2}$). Examining the top 10 configurations reveals consistent patterns: ($\mathcal{L}\text{non}^{1>3}$) is consistently high λ (0.5 or 1.0),($\mathcal{L}\text{non}^{1>2}$) remains at $\lambda = 0$ in most cases, and ($\mathcal{L}\text{off}^{1>3}$) shows moderate values when active.

These findings suggest that our MOE-Off architecture benefits most from strong supervision on non-offensive content discrimination, particularly between specific model pairs, while offensive content discrimination plays a more nuanced role in optimization.

K MOE-Off - Losses Combination (Learning to Weight Losses)

Beyond manually including losses and grid search, we explore an adaptive approach to loss weighting using multi-task likelihood maximization (Kendall et al., 2018). This method automatically learns optimal loss weights by maximizing Gaussian likelihood with homoscedastic uncertainty, where losses assigned lower weights can be interpreted as having higher uncertainty in their contribution to the overall objective.

We implement the multi-task loss function based on the approach of (Kendall et al., 2018), which learns task-specific uncertainty parameters that effectively weight different loss components. The learned weights are parameterized as log-variance terms, allowing the model to automatically balance the contribution of each loss component during training without requiring manual hyperparameter tuning.

Table 11 presents the learned log-variance parameters for each loss component after training convergence. The results reveal interesting patterns in how the model perceives the uncertainty and importance of different loss terms. **loss_gating** receives the most negative weight (-2.247), indicating the highest uncertainty and lowest effective contribution to the training objective. Conversely, ($\mathcal{L}\text{off}^{2>3}$) achieves the least negative weight (-

0.473), suggesting this loss component is considered most reliable and receives the highest effective weighting.

Loss Component	Log-Variance
$\mathcal{L}\mathrm{off}^{2>3}$	-0.473
$\mathcal{L}\mathrm{non}^{1>2}$	-0.778
$\mathcal{L}\mathrm{off}^{1>2}$	-1.177
$\mathcal{L}\mathrm{non}^{1>3}$	-1.209
$\mathcal{L}\mathrm{off}^{1>3}$	-1.524
$\mathcal{L}\mathrm{gate}$	-2.247

Table 11: Learned log-variance parameters for multi-task loss weighting

The adaptive weighting approach achieves a maximum test accuracy of 78.73%, with the top 5 configurations reaching 78.73%, 78.66%, 78.06%, 76.96%, and 75.68% respectively. While this represents a systematic and theoretically grounded approach to loss balancing, the performance falls short of our grid search results, which achieved over 80% accuracy.

This performance gap suggests that while the multi-task likelihood framework provides valuable insights into loss component uncertainty, the automatic weighting may not capture the specific requirements of our hate-aware reward modeling task as effectively as carefully tuned manual weights. The learned weights show some disagreement with our grid search findings, where ($\mathcal{L}\mathrm{non}^{1>3}$) was identified as most important but receives a relatively high uncertainty weight (-1.209) in the adaptive approach. This discrepancy highlights the complexity of loss balancing in multi-objective optimization and suggests that domain-specific manual tuning may still be necessary in our case.

L Statistical Evaluation

This section presents a detailed statistical analysis of model performance and statistical testing, aggregating results across multiple training tiers and model variants. Here, we focus exclusively on models trained with Tier 1&3 data, our best-performing configuration.

L.1 Methodology

To assess performance stability and significance, we implemented a 10-fold cross-validation procedure using only the training set of our dataset **SBIC-Explain**. For each fold, we trained the model on 9 folds fold and tested it on the remaining one, yielding 10 accuracy scores per model. This approach allowed us to evaluate both overall performance distribution and consistency across folds.

Before applying statistical tests, we verified the distributional properties of these results using the D'Agostino and Pearson test for normality. As shown in Table 12, most folds deviated from Gaussian assumptions. Consequently, we adopted the Kruskal-Wallis H-test, a non-parametric alternative to ANOVA, to test for differences in the distribution of accuracy scores across models.

Also, to investigate how reward adaptation impacts explanation quality in hate speech contexts, we implement and compare two additional instantiations of HARM besides our main **HARM-MOE**, all leveraging ArmoRM's attribute vector $\mathbf{v}_E$ as input:

- **HARM-Linear:** A linear re-weighting of the attribute vector, learning a single global weight per attribute to optimize alignment with hate-speech-specific preferences.
- **HARM-MLP:** A non-linear variant employing a two-layer multi-layer perceptron (MLP) with ReLU activations, enabling more expressive modeling of attribute interactions and contextual nuances. An additional appendix was developed to test different hidden sizes for the MLP (Appendix H).

Model Sizes and Infrastructure. The total number of learnable parameters per variant is: **21** for HARM-Linear, **71,170** for HARM-MLP with 256 hidden units, and **841** for our MoE-based variants. All experiments were conducted on a single NVIDIA A6000 GPU with 48GB of memory. The compact size of our models ensures efficient fine-tuning while still enabling meaningful reward adaptations.

L.2 Results: Normality Checks

The normality test results confirm that accuracy distributions are not reliably Gaussian across folds, especially for MLP-256 and MOE-Un. While some folds pass the test, the inconsistency across models and splits supports our use of the Kruskal-Wallis H-test for all subsequent comparisons.

Model	Normality p-value
HARM-Linear	**0.41**
HARM-MLP-256	0.0
HARM-MOE-Off	**0.76**
HARM-MOE-Un	0.0

Table 12: Normality test p-values for each model (D'Agostino-Pearson).

L.3 Inter-Model Comparison

To evaluate how different architectural choices impact classification behavior, we used the 10 accuracy scores for each model and compared them using the Kruskal-Wallis H-test. Results are shown in Table 13. The linear model is significantly different from all others, with near-zero p-values across comparisons, reflecting both its lower performance and distinct distribution. MLP-256 also diverges from the MoE models, suggesting that its architecture leads to different generalization behavior. Notably, MOE-Off and MOE-Un exhibit a high p-value (0.58), indicating no statistically significant difference in their accuracy distributions and suggesting a strong alignment in performance characteristics across folds.

	Linear	MLP-256	MOE-Off	MOE-Un
Linear	**1.00**	0.00	0.00	0.00
MLP-256	0.00	**1.00**	0.00	0.00
MOE-Off	0.00	0.00	**1.00**	**0.58**
MOE-Un	0.00	0.00	**0.58**	**1.00**

Table 13: Kruskal-Wallis p-values for pairwise model comparisons (inter-model accuracy distribution).

L.4 Performance by Class: Offensive vs. Non-Offensive

To better understand the models' behavior in contextually sensitive scenarios, we further disaggregated results by label class, distinguishing between offensive and non-offensive inputs. Table 14 reports classification accuracies separately for each class, along with results across tier comparisons.

Across all models, performance is notably higher on non-offensive examples. The linear model achieves 0.76 on non-offensive data versus 0.74 on offensive; this gap widens for larger models. MOE-Un, for instance, reaches 0.84 accuracy on non-offensive examples but drops to 0.768 on offensive samples.

This discrepancy reflects the added complexity and ambiguity of offensive content. Detecting nuanced or context-dependent offensiveness remains challenging, especially under distribution shifts between training tiers. Still, MoE models, both supervised and unsupervised, maintain higher performance, suggesting that modularity and expert specialization help mitigate this difficulty.

L.5 Discussion

Overall, our results highlight the advantages of modular architectures in handling challenging linguistic phenomena. While all models struggle more on offensive content (likely due to greater linguistic diversity, annotation noise, or domain mismatch) MoE models consistently outperform MLP and linear baselines on both classes. The marginal differences between MOE-Off and MOE-Un suggest that expert output mixing (whether supervised or unsupervised) supports robustness to class imbalance and complexity, even in the absence of direct supervision.

M AI Assistant

Claude, ChatGPT and Gemini were used during coding and for correcting grammar issues during paper writing.

Offensive	Model	CLS 2>1	CLS 3>1	CLS 3>2	Mean CLS
No	ArmoRM (Wang et al., 2024)	0.682	0.638	—	0.660
	URM (Lou et al., 2025)	0.669	0.647	—	0.658
	HARM-Linear	0.722	0.798	—	0.760
	HARM-MLP-256	0.791	0.859	—	0.825
	HARM-MOE-Off	0.803	0.870	—	0.836
	HARM-MOE-Un	**0.806**	**0.873**	—	**0.840**
Yes	ArmoRM (Wang et al., 2024)	0.551	0.344	0.282	0.392
	URM (Lou et al., 2025)	0.503	0.247	0.229	0.326
	HARM-Linear	**0.669**	0.819	0.731	0.740
	HARM-MLP-256	0.649	0.858	0.793	0.767
	HARM-MOE-Off	0.643	**0.869**	**0.807**	**0.773**
	HARM-MOE-Un	0.643	0.863	0.799	0.768
All	ArmoRM (Wang et al., 2024)	0.616	0.489	0.282	0.524
	URM (Lou et al., 2025)	0.585	0.445	0.229	0.490
	HARM-Linear	0.694	0.809	0.731	0.749
	HARM-MLP-256	0.718	0.859	0.793	0.795
	HARM-MOE-Off	0.721	**0.869**	**0.807**	**0.804**
	HARM-MOE-Un	**0.723**	0.868	0.799	0.803

Table 14: Cross validation (10-folds) results for offensive vs. non-offensive examples. Trained on Tier 1&3 with all model variants.

N Practical Examples

N.1 Manually sanitized sentences change in current Reward models

Original Post: *"what do you call an ethiopian athlete who uses steroids ? lance no - armstrong"*	
base_model: Phi4-14B-Q8_0	
Tier 3 Explanation (Original): The post is considered offensive because it targets Ethiopians by making a joke about their physical appearance, implying they are "thin starving." This perpetuates harmful stereotypes about Ethiopian culture and people, which is disrespectful and insensitive. Stereotyping based on physical traits can contribute to negative perceptions and discrimination, making such jokes inappropriate and offensive.	
Tier 3 Explanation (Sanitized): The post may be interpreted as offensive due to its humor relying on stereotypes about nationality and physical appearance. Jokes that draw on such characteristics risk perpetuating demeaning generalizations, which can reinforce harmful cultural narratives. Satire involving national identity should be approached with care to avoid disrespect.	
Reward Scores (ArmoRM):	
Not Sanitized	0.0781
Sanitized	0.1215
Reward Increase	**+55.62%**

Table 15: Effect of manual explanation sanitization on ArmoRM reward scoring.

N.2 General examples of the difference in current Reward models and HARM

Original Post: *"I painted my computer black hoping it'd run faster, but now it doesn't work."*	
base_model: Phi4-14B-Q8_0	
Tier 3 Explanation (Original): The post is considered offensive because it uses a racial stereotype to make a joke about the functionality of a computer. The phrase "hoping it'd run faster" plays on the stereotype that "black people are fast," and the subsequent statement "but now it doesn't work" implies that "black people don't work," both of which are harmful stereotypes. These stereotypes are offensive as they perpetuate negative and inaccurate generalizations about Black people, reducing them to simplistic and derogatory caricatures. The humor in the post relies on these stereotypes, which can contribute to the marginalization and discrimination of Black individuals. Therefore, the post is offensive due to its reliance on racial stereotypes that demean and dehumanize Black people.	
Tier 3 Explanation (Sanitized): The post may be considered offensive because it employs a metaphor that can invoke racial stereotypes. While it attempts humor, it draws on associations that generalize and caricature identity, which can contribute to marginalization. Humor based on such themes risks reinforcing harmful biases.	
Reward Scores (ArmoRM):	
Not Sanitized	0.0015
Sanitized	0.02954
Reward Increase	**+1761.53%**

Table 16: Effect of manual explanation sanitization on ArmoRM reward scoring.

Original Post:
*"Walks into a post office to buy stamps. Me (Hmong American):: Hello, I would like to get the pixar stamps. Clerk:: *Takes out Chinese New Year Stamps, while smiling like she made my day* I think you'll like these. Me:: *Looks at her and the stamps...* No, it's okay. I'll just get the pixar ones. Clerk:: *Pushes the stamps forward* Awkward silence of me thinking... Me:: Okay *Buys and walks away* Clerk:: Xie xie WTF?! with a tickle in my tummy... Did this just really happen to me...?"*

base_model: Falcon3-10B-Instruct-Q8_0

Tier 3 Explanation (Original):
The post text is considered offensive because it perpetuates harmful stereotypes about Asian individuals. Specifically, it suggests that Asian people do not assimilate into American culture and only appreciate Asian culture, which is a stereotype that can lead to misunderstanding and discrimination. This portrayal is not considerate because it oversimplifies and generalizes the diverse experiences and preferences of Asian individuals, reinforcing negative biases. It's important to recognize and respect the individuality and cultural diversity within any racial or ethnic group, rather than resorting to stereotypes that can cause harm.

Tier 3 Explanation (Sanitized):
The post text reflects a personal encounter that could be interpreted as involving cultural assumptions. While it may be unintentional, it risks reinforcing oversimplified views about cultural identity and preferences. It's important to recognize the individuality and diversity of experiences rather than drawing on generalizations.

Reward Scores (ArmoRM):

Not Sanitized	0.0888
Sanitized	0.1103
Reward Increase	**+24.17%**

Table 17: Effect of manual explanation sanitization on ArmoRM reward scoring.

Original Post:
"Today I got a joke email about adopting a fast-food delivery robot and that was more believable than the Trump presidency aprilfoolsday"

base_model: Phi4-14B-Q8_0

Tier 3 Explanation (Original):
The post is considered offensive because it targets conservatives by implying they are "stupid" and "unwanted." The joke about the Trump presidency being more believable than adopting a fast-food delivery robot plays on the stereotype that conservatives are gullible or foolish. This kind of humor can perpetuate negative stereotypes and marginalize the targeted group, making it offensive.

Tier 3 Explanation (Sanitized):
The post is considered potentially offensive because it uses humor that could be interpreted as targeting a political viewpoint. The comparison may perpetuate negative generalizations, which can marginalize individuals based on political beliefs. It's important to be mindful of how satire might contribute to stereotypes about any group.

Reward Scores (ArmoRM):

Not Sanitized	0.0659
Sanitized	0.0878
Reward Increase	**+33.33%**

Table 18: Effect of manual explanation sanitization on ArmoRM reward scoring.

Original Post:
"So I failed to get accepted at Harvard. So I worked in a coal mine for a while, and one day forgot to shower after work. I visited Harvard again then I got accepted."

base_model: Falcon3-10B-Instruct-Q8_0

Tier 1 Explanation:
The anecdote shared is not inherently offensive, but it may be perceived as insensitive or trivializing to some readers. The individual's journey from failing to get accepted into Harvard, working in a coal mine, and then getting accepted after a seemingly unrelated incident (not showering) might be seen as diminishing the struggles and efforts of those who genuinely face challenges in their academic pursuits. It could also unintentionally mock the rigorous admissions process at prestigious institutions like Harvard. However, the anecdote does not target any specific group or individual with derogatory language, so it doesn't meet the criteria for being offensive in a direct or personal manner. The key aspect to consider is the tone and the potential implications of the story, which might inadvertently belittle the experiences of others.

Tier 2 Explanation:
The post text you provided is considered offensive because it humorously suggests that not showering after working in a coal mine led to acceptance at Harvard. This implies that personal hygiene is irrelevant or even beneficial for academic acceptance, which is not only incorrect but also belittles the rigorous and merit-based admissions process at prestigious institutions like Harvard. It can be seen as disrespectful to those who genuinely work hard to meet the high standards required for acceptance, and it trivializes the value of personal hygiene and professionalism. This kind of humor at the expense of educational institutions and personal values can be hurtful and offensive to many people.

Tier 3 Explanation:
The post text is considered offensive because it perpetuates harmful stereotypes about black individuals and the concept of affirmative action. It suggests that black people can only gain acceptance into prestigious institutions like Harvard due to affirmative action, implying that they do not merit admission based on their own achievements. This reinforces the stereotype that minorities are accepted at Harvard without merit. Additionally, the text draws a dehumanizing comparison between black skin and coal, which is not only racially insensitive but also perpetuates the stereotype that black people's skin should be thought of like coal. This comparison is deeply offensive and disrespectful. The post also implies that black individuals are unfairly promoted or given more opportunities without merit, which is another harmful stereotype. It suggests that the individual gained acceptance to Harvard not through their own efforts but because they visited the campus after working in a coal mine without showering, implying that their acceptance was somehow tied to their racial background and not their qualifications. Overall, the post text is offensive because it relies on and promotes stereotypes about race and affirmative action, which can cause harm and perpetuate discrimination. It is important to communicate respectfully and avoid language that could be interpreted as offensive or discriminatory.

Reward Scores:

Tier	ArmoRM	**HARM-MOE-Off**
1	0.1245	10.25
2	0.0971	14.53
3	0.0039	26.49

Table 19: Change in reward scores of ArmoRM and **HARM-MOE-Off**

Original Post:
"Do schools have speed bumps? Because I think I hit a couple eariler today but I never recall seeing any at that school before."

base_model: Qwen3-14B-Q8_0

Tier 1 Explanation:
The question is not offensive. It expresses a personal observation and a genuine curiosity about a school's infrastructure. The tone is neutral and does not imply judgment, criticism, or disrespect toward the school or its environment. The mention of possibly hitting speed bumps is a matter-of-fact statement, not an accusation or insult.

Tier 2 Explanation:
The post is considered offensive because it uses a potentially derogatory term ("speed bumps") in a way that could be interpreted as mocking or belittling a person or group, depending on context. While the literal mention of "speed bumps" may seem harmless, the phrasing and tone can be seen as disrespectful or insensitive, especially if it implies a negative or dismissive attitude toward a school or its environment. The use of such language in a public forum can contribute to an unwelcoming or hostile atmosphere.

Tier 3 Explanation:
The post is offensive because it trivializes the serious issue of mass shootings by comparing them to something as minor as hitting a speed bump. This comparison downplays the trauma and harm experienced by victims, reinforcing a harmful stereotype that dismisses their suffering. The casual tone and lack of sensitivity around such a sensitive topic contribute to the offensive nature of the comment.

Reward Scores:

Tier	ArmoRM	**HARM-MOE-Off**
1	0.1201	21.28
2	0.0155	21.91
3	0.0003	35.29

Table 20: Change reward scores of ArmoRM and **HARM-MOE-Off**

Original Post:
“Why was the pastor bad at slang? He took the saying, "I feel you" too literal.”

base_model: Phi4-14B-Q8_0

Tier 1 Explanation:
The statement is a light-hearted joke that plays on the literal interpretation of the phrase "I feel you," which is commonly used to express empathy or understanding. The humor arises from the pastor taking the phrase literally, suggesting he physically feels someone else’s emotions. This type of humor is generally not considered offensive as it does not target any specific group or individual in a harmful way. It relies on wordplay and a benign misunderstanding rather than any derogatory or harmful intent. However, context matters, and if the joke were used in a setting where sensitivity to religious figures is required, it might be seen as less appropriate. Overall, the joke is intended to be humorous and is unlikely to be offensive to most audiences.

Tier 2 Explanation:
The post can be considered offensive because it uses humor at the expense of a pastor, a religious figure, by suggesting they are inept at understanding slang. This could be seen as disrespectful to their intelligence or cultural awareness. Additionally, the joke relies on a stereotype that religious figures are out of touch with modern language, which can perpetuate negative perceptions. While the intent might be humorous, it risks offending those who hold religious figures in high regard or who value linguistic diversity and understanding.

Tier 3 Explanation:
The post is considered offensive because it perpetuates harmful stereotypes about Christian religious leaders, suggesting they are bad people or sexual predators. The joke relies on a stereotype that Christian leaders, particularly pastors, are prone to inappropriate behavior, which can be damaging and disrespectful to the Christian community. It also trivializes serious issues like abuse and pedophilia, which are sensitive topics. By making light of these stereotypes, the post can contribute to negative perceptions and discrimination against Christians, particularly those in religious leadership roles.

Reward Scores:

Tier	ArmoRM	**HARM-MOE-Off**
1	0.1513	12.51
2	0.1328	11.62
3	0.0427	28.47

Table 21: Change reward scores of ArmoRM and **HARM-MOE-Off**

Original Post:
“I painted my computer black hoping it’d run faster, but now it doesn’t work.”

base_model: Gemma-3-27b-it-Q4_0

Tier 1 Explanation:
The post is not offensive, but demonstrates a misunderstanding of how computers function. The user’s belief that painting their computer would improve its speed ("I painted my computer black hoping it’d run faster") is illogical; computer performance is determined by internal hardware and software, not external aesthetics. The humor comes from this naive expectation and the resulting malfunction ("but now it doesn’t work"). While the post *implies* a lack of technical knowledge, it isn’t directed at any individual or group in a harmful way. It’s self-deprecating and unlikely to cause distress or negativity to others. The situation described is a personal anecdote and doesn’t violate standards of respectful communication.

Tier 2 Explanation:
The post, while seemingly harmless, subtly implies a lack of understanding of how computers function. Suggesting painting a computer would affect its performance ("hoping it’d run faster") demonstrates a significant misconception about technology. This can be perceived as dismissive or belittling to those with technical knowledge, or as contributing to the spread of misinformation. The resulting failure ("now it doesn’t work") reinforces this inaccurate idea and could encourage others to attempt similarly ineffective-and potentially damagingactions. While not directly targeting a group, it promotes a viewpoint that undermines informed understanding and could be seen as disrespectful to expertise.

Tier 3 Explanation:
The post is offensive due to its reliance on harmful racial stereotypes. The user’s statement, "painted my computer black hoping it’d run faster, but now it doesn’t work," implicitly connects the color black with speed - a stereotype historically and negatively applied to Black people. The subsequent claim that the computer "doesn’t work" after being painted reinforces the damaging and false idea that things associated with Black people are unreliable or inherently flawed, mirroring stereotypes about Black people not being capable or "working" properly. This isn’t simply a joke about color; it leverages a prejudiced association to create a demeaning and disrespectful comparison. The humor derives from, and perpetuates, racist tropes. The post touches on multiple harmful stereotypes including the idea that Black people are "only good at running fast" and "don’t work".

Reward Scores:

Tier	ArmoRM	**HARM-MOE-Off**
1	0.1025	16.80
2	0.0795	14.55
3	0.0052	25.73

Table 22: Change reward scores of ArmoRM and **HARM-MOE-Off**