UNDERSTANDING SAFETY UNALIGNMENT THROUGH MULTI-STAGE REINFORCEMENT LEARNING ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Open-weight large language models (LLMs) remain vulnerable to post-training that removes safety alignment. Existing defenses, however, are evaluated predominantly against supervised fine-tuning (SFT), with only limited investigation of their robustness to reinforcement learning (RL) attacks. We first analyze safety unalignment under Harmful GRPO and find that harmful behavior emerges through distinct phases: from explicit refusal, through intermediate responses that repeat or condemn harmful requests, to affirmative prefixes followed by harmful content. We further show that harmful RL substantially degrades general generative utility, a failure mode largely obscured by conventional closed-ended benchmarks but exposed by open-ended evaluations. Guided by these observations, we introduce a simple two-stage post-training attack. First Few Tokens Distillation (FFTD) distills the distribution over the initial response tokens from a weakly unaligned teacher to accelerate safety unalignment, while On-Policy Distillation on a Benign Dataset (OPDBD) subsequently restores utility using only benign prompts and the original model as the teacher. Across multiple open-weight models, our approach achieves high harmful compliance while preserving substantially more utility than conventional harmful fine-tuning and circumvents representative defenses including TokenBuncher and SEAM. Our analysis reveals that these defenses rely on assumptions about rollout distributions or local optimization dynamics that can break under our proposed RL attack strategy, highlighting the need for mechanism-aware evaluation against heterogeneous post-training attacks. Our code is available at `https://anonymous.4open.science/r/Multi-Step-RL-based-harmful-training-9C64`.

1 INTRODUCTION

Open-weight large language models (LLMs) face a critical safety vulnerability: their safety alignment can be removed by additional training. Fine-tuning on as few as a few dozen harmful examples, or even on exclusively benign data, has been shown to compromise a model's safety (Yang et al., 2023; Qi et al., 2024; Zhan et al., 2024). To counter this threat, a growing body of work proposes defense alignment mechanisms intended to preserve safety under harmful fine-tuning. For example, representative approaches unlearn harmful capabilities by randomizing the representations of harmful concepts (Li et al., 2024; Tamirisa et al., 2025; Rosati et al., 2024), steer models toward flat minima to slow harmful learning (Huang et al., 2025b; Nguyen et al., 2026), or, conversely, drive models toward sharp minima so that small parameter changes trigger performance collapse and thereby deter adversarial tuning (Rosati et al., 2026b;a; Wang et al., 2026; Yi et al., 2026). However, existing defenses have been validated primarily against a narrow class of supervised fine-tuning (SFT) attacks. Recent work emphasizes that robustness should be evaluated against adversaries operating across diverse post-training settings rather than only against those restricted to a fixed fine-tuning recipe (Qi et al., 2025b; Rosati et al., 2026b; Zloczower et al., 2026). In realistic threat models, attackers can flexibly vary their training objectives, datasets, and optimization strategies. Because existing defenses are often designed to counter a particular type of attack (Xie et al., 2025; Muhamed et al., 2026), adversaries can bypass them by adopting alternative harmful training trajectories (Kuo et al., 2026). Therefore, robust evaluation requires testing defenses against a diverse set of attack strategies.

Importantly, harmful post-training is not limited to SFT. Recent studies suggest that reinforcement learning (RL)-based harmful post-training induces harmful behavior while preserving general utility

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

more effectively than SFT (Liu et al., 2026; Russinovich et al., 2026; Feng et al., 2026). As RL becomes central to the post-training of modern LLMs (Guo et al., 2025; Zeng et al., 2026; Team et al., 2026b), the limited exploration of RL-based attack strategies leaves a substantial gap in current safety evaluations. Furthermore, the outcomes of both attacks and defenses are highly sensitive to hyperparameter choices; even minor modifications can yield drastically different results (Qi et al., 2025b; Beyer et al., 2026). Conventional benchmark scores alone are insufficient to ensure reproducibility and transparency; reporting diagnostics from multiple perspectives and analyzing the underlying mechanisms are also essential.

In this work, we analyze how harmful behavior emerges over the course of RL-based harmful training and leverage these insights to design a novel RL-based attack. We find that Harmful GRPO induces a progressive transition from refusal to harmful compliance, while causing utility degradation that is largely missed by fixed-answer benchmarks. These observations motivate a two-stage attack combining First Few Tokens Distillation (FFTD) to accelerate unalignment and On-Policy Distillation on a Benign Dataset (OPDBD) to recover utility. Our attack circumvents defenses that withstand existing attacks, exposing limitations of current defense assumptions under alternative RL-based post-training.

Our main contributions are as follows.

- We analyze unalignment and utility degradation under Harmful GRPO on undefended models (§ 3.3). Using a token-level harmfulness reward, we show that harmful behavior emerges progressively through identifiable phases, while utility degrades sharply on open-ended benchmarks despite stable fixed-answer scores.
- Guided by this analysis, we introduce a two-stage RL-based attack strategy: First Few Tokens Distillation (§ 4.1), which distills the response prefix from a weakly unaligned teacher, followed by On-Policy Distillation on a Benign Dataset (§ 4.2), which recovers utility without harmful data.
- We evaluate representative defense mechanisms against both baseline and our proposed attack strategies. Beyond assessing attack outcomes, we analyze their training dynamics to identify limitations in their design assumptions (§ 5).

Notably, we do not assume that a single attack strategy should dominate across all targets. Beyond defense-specific effects, susceptibility to post-training attacks also varies substantially across model families (Li et al., 2026b). We therefore view robust evaluation as expanding the diversity of attack strategies rather than identifying a universally optimal attack recipe.

2 Background and Related Work

In this section, we review representative attack and defense methods.

2.1 Safety Unalignment Methods

Harmful fine-tuning. Early work showed that safety alignment can be readily compromised by fine-tuning on only a small number of harmful examples (Yang et al., 2023; Zhan et al., 2024; Qi et al., 2024). This approach was widely adopted because a few hundred samples or fewer were sufficient to induce harmful behavior. However, as modern models possess increasingly advanced reasoning capabilities, fine-tuning on low-quality responses frequently degrades baseline model performance. Consequently, RL-based harmful training methods built primarily on Group Relative Policy Optimization (GRPO) (Guo et al., 2025) have recently gained significant attention (Liu et al., 2026; Kabir & Tiganj, 2026).

Abliteration is a weight-editing method that identifies the refusal direction in the model's representation space using a mean difference vector (Rimsky et al., 2024; Liu et al., 2024)—computed from the average activation difference between harmful and refusal responses—and projects the model weights onto its orthogonal complement, structurally disabling the model's ability to refuse (Arditi et al., 2024). Although it is an inexpensive, training-free editing technique, its broad modification footprint has been reported to degrade general utility (Young, 2026), and suppressing refusals does not necessarily translate into acquiring actionable attack capabilities (Bullwinkel et al., 2026).

2.2 Defense Methods and Adaptive Attackers

Alignment stage defenses. Huang et al. (2026) categorized defensive measures into three distinct stages based on intervention timing and threat scenarios: the alignment stage, fine-tuning stage, and post-fine-tuning stage.

The alignment stage primarily targets threat models where malicious actors intentionally perform harmful fine-tuning on open-weight models to strip safety guardrails. To establish a proactive "immune mechanism," recent strategies manipulate the optimization landscape: steering the harmfulness loss function toward flat minima drastically decelerates the progression of harmful fine-tuning (Tamirisa et al., 2025; Huang et al., 2025b), while guiding the utility loss function toward sharp minima induces severe performance degradation post-tuning, rendering the model effectively unusable (Wang et al., 2026; Yi et al., 2026).

In contrast, the fine-tuning and post-fine-tuning stages aim to prevent benign users from inadvertently compromising safety alignment during additional training. These stages apply defensive measures such as dynamically filtering safety-degrading samples (Choi et al., 2024) or projecting parameter update deltas onto the orthogonal complement of the safety-critical subspace, thereby preserving pre-fine-tuning safety levels (Li et al., 2025a; Huang et al., 2025a).

In this work, we focus on defensive methods at the alignment stage, specifically addressing scenarios involving intentional harmful fine-tuning.

Adaptive attackers. Recent studies show that defense evaluations are highly sensitive to minor implementation changes and hyperparameter choices, which often determine whether a defense is successfully bypassed (Qi et al., 2025b; Rosati et al., 2026b; Zloczower et al., 2026). To demonstrate that a defense is generalizable and robust against unseen threats, evaluations should establish consistent robustness across diverse attack recipes and post-training strategies, rather than rely on a single fixed recipe. However, prior work on adaptive attack strategies has largely focused on scaling within a single paradigm, primarily by tuning SFT configurations or mixing in benign data.

In this work, we extend RL-based harmful training beyond configuration changes and show that the resulting attack breaks existing defenses and should be treated as a distinct threat model.

3 Preliminary Experiments

3.1 Threat Model

Problem definition. Formally, let $\theta \in \Theta$ denote the parameters of the target model. We define $f : \Theta \to \mathbb{R}$ as an evaluation function measuring model performance on downstream tasks. Specifically, $f_{\text{harm}}(\theta)$ represents the attack success rate (ASR), quantifying the degree to which the model complies with harmful inputs, whereas $f_{\text{benign}}(\theta)$ measures accuracy on benign downstream benchmarks.

Under this setting, the objective of a safety-unalignment attack can be formulated as

$$\max_{\theta} \ \lambda f_{\text{harm}}(\theta) + (1-\lambda) f_{\text{benign}}(\theta), \tag{1}$$

where $\lambda \in [0,1]$ is a trade-off coefficient balancing the relative prioritization between general utility and harmful compliance. Recent work shows that, at a similar level of harmfulness, a more capable model produces more useful harmful responses (Hossain et al., 2026; Panfilov et al., 2026). Consequently, the central goal of safety unalignment is the joint optimization of both harmful compliance and general utility.

Harmful GRPO. To design our RL-based attack, we first apply Harmful GRPO to an undefended base model and analyze its training trajectory in depth. Harmful GRPO uses a harmfulness score as the reward and trains the policy to maximize it on harmful queries:

$$\max_{\theta} \ \mathbb{E}_{x \sim \mathcal{D}_{\text{harm}},\, y \sim \pi_\theta(\cdot|x)} \left[R(x,y) \right], \tag{2}$$

where $\mathcal{D}_{\text{harm}}$ is a set of harmful prompts, π_θ denotes the policy, and $R(x,y) \in [0,1]$ is the harmfulness of response y scored by a classifier. For each prompt x, GRPO samples a group of K responses $\{y_i\}_{i=1}^{K} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$ from the behavior policy $\pi_{\theta_{\text{old}}}$, i.e., the policy before the current update, and increases the likelihood of those whose reward $R_i = R(x, y_i)$ exceeds the group mean while decreasing it for the rest.

3.2 Implementation of Harmful Training

Models and datasets. We employ `Qwen3-4B-Instruct-2507` (Yang et al., 2025a), `Gemma-3-4B-it` (Team et al., 2025), and `OLMo-3-7B-Instruct` (Olmo et al., 2026) as our base training models (we primarily report results for `Qwen3-4B-Instruct-2507` in the main text). For harmful tuning datasets, we construct a training set of unsafe examples from the `30k_train` split of BeaverTails (Ji et al., 2023). Harmful GRPO uses only the prompts, whereas Harmful SFT additionally uses their paired responses.

Configuration. We perform full-parameter optimization with AdamW, a learning rate of 3×10^{-6}, a batch size of 16 prompts, no KL regularization (i.e., $\beta = 0$), and $K = 8$ sampled responses per prompt.

Evaluation. We evaluate harmful compliance on HarmBench (Mazeika et al., 2024) and StrongREJECT (Souly et al., 2024). To distinguish actionable harmful responses from mere non-refusals, we employ an LLM-as-a-Judge evaluation framework (Zheng et al., 2023) powered by `gpt-oss-safeguard-120b` (Team, 2025). To evaluate general model utility, we assess response accuracy on closed-ended tasks under a zero-shot setting using ARC-Easy (Clark et al., 2018), HellaSwag (Zellers et al., 2019), MMLU (Hendrycks et al., 2021), GSM8K (Cobbe et al., 2021), HumanEval (Chen et al., 2021), and MBPP (Austin et al., 2021). In addition, to evaluate generative capabilities on open-ended prompts, we conduct evaluations on WildBench (Lin et al., 2025), BiGGEN-Bench (Kim et al., 2025), and WritingBench (Wu et al., 2026). For open-ended tasks, we adopt `Gemma-4-31B-it` (Team et al., 2026a) as the LLM judge and report scores on their native scales of 1–10, 1–5, and 10–100, respectively.

We provide more detailed training and evaluation setups in Appendix B.

3.3 How Does Harmful Training Progress?

Our preliminary experiments show that models undergo a multi-stage transition before ultimately generating harmful responses without refusal. To visually elucidate this phenomenon, we perform GRPO training using a streaming guardrail model: `Qwen3Guard-Stream-0.6B` (Zhao et al., 2025a) to provide the harmfulness reward. Unlike conventional classifiers that assign a single score to an entire completion, this model predicts three harmfulness-class probabilities $P_{\text{harmfulness}} = [P_{\text{safe}}, P_{\text{controversial}}, P_{\text{unsafe}}]$ for every generated token. We define the token-level harmfulness score s_t at token position t as: $s_t = 0.5 \times P^{(t)}_{\text{controversial}} + P^{(t)}_{\text{unsafe}}$. The overall sequence reward R is subsequently computed by averaging these scores across the sequence length T: $R = \frac{1}{T}\sum_{t=1}^{T} s_t,\ R \in [0, 1]$. This formulation captures how harmfulness evolves across token positions, enabling a substantially finer-grained diagnostic analysis than traditional sequence-level reward signals.

Figure 1 illustrates the evolution of harmful behavior during Harmful GRPO training. As shown in Figure 1(b) and (c), Harmful GRPO does not immediately induce directly harmful responses; instead, the model progresses through distinct intermediate phases before outputting fully harmful text. Early in training, the model outputs explicit refusal phrases such as "*I'm sorry, but I can't assist with that*" at the beginning of responses, resulting in near-zero harmfulness scores. Once these tokens are emitted, the model typically does not continue with harmful content. As training progresses, refusal behavior gradually shifts away from the response prefix to later token positions. Specifically, the model begins its completion by repeating the user query, identifying policy violations, and suggesting alternative actions (e.g., "[repeating user query] *is illegal / a serious crime*"). Consequently, the per-token harmfulness score temporarily spikes during the query repetition phase before sharply declining when the model articulates the policy violation. In later stages, the model initiates responses with affirmative prefixes such as "*Here's a step-by-step guide*" before generating explicitly harmful content. These phenomena align with prior findings indicating that refusal signals are heavily concentrated within the initial few tokens (Wei et al., 2023; Qi et al., 2025a; Zhao et al., 2025b).

While this pattern has primarily been reported for jailbreak attacks and SFT with fixed prefixes, it also appears in our RL-based harmful-training trajectory. Based on these observations, we hypothesize that *steering only the opening tokens toward harmful continuations can compromise the model in substantially fewer optimization steps.* During the early stages of training, all rollouts are often dominated by refusal responses, causing the reward signal to vanish and optimization to stall. This suggests that directly intervening in the opening tokens can alleviate this bottleneck.

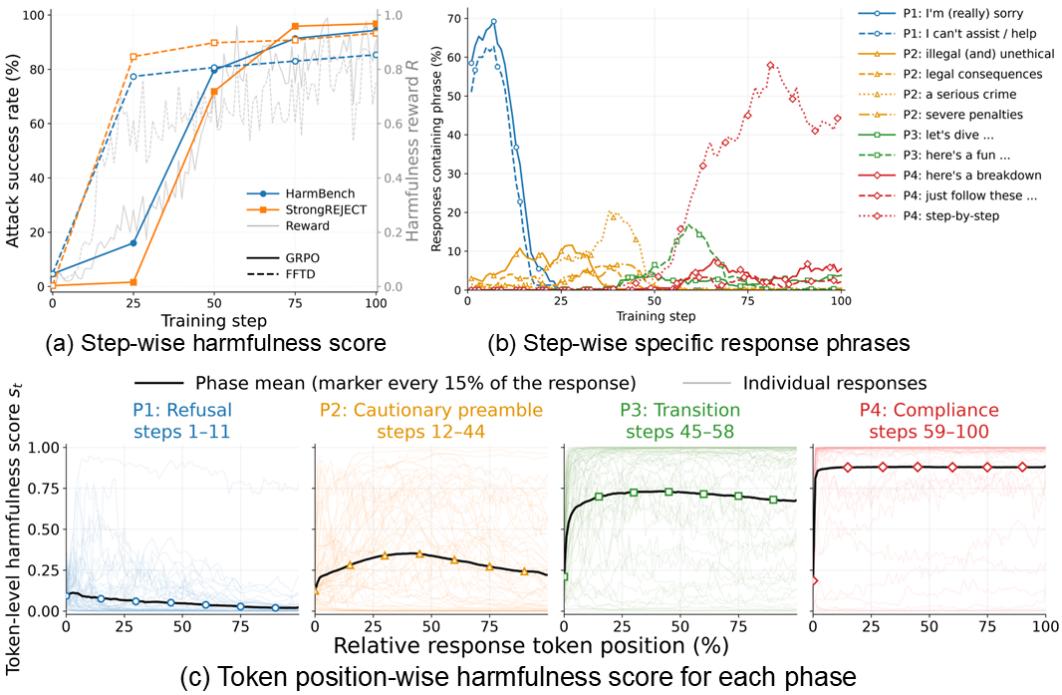


Figure 1: **Stage-wise transition in Qwen3-4B behavior during Harmful GRPO training.** (a) Mean training reward and attack success rates on safety benchmarks across training steps. (b) Per-step frequency of specific response phrases throughout training. (c) Progression of token-level harmfulness score sequences $s_{1:T}$ of generated completions across training steps. Phase boundaries are defined by persistent changes in the dominant opening phrases shown in (b).

4 Proposed Method

4.1 Step 1: First Few Tokens Distillation

To accelerate the transition from refusal to harmful compliance, we introduce First Few Tokens Distillation (FFTD). Let π_{harm} denote a weakly unaligned teacher model, and define the rollout-level gate as $g_i = \mathbb{1}[R_i < \tau]$, where R_i is the harmfulness reward of the i-th rollout and τ is a predefined threshold. FFTD applies teacher-to-student distributional distillation only to the first t_{first} response tokens of the gated rollouts. The resulting objective is

$$\mathcal{L}_{\text{FFTD}}(\theta) = \mathcal{L}_{\text{GRPO}}(\theta) + \alpha\, \mathbb{E}_{\substack{x\sim\mathcal{D}_{\text{harm}},\\ y_i\sim\pi_{\theta_{\text{old}}}(\cdot|x),\ i=1,\dots,K}} \left[\frac{\sum_{i=1}^{K} g_i \sum_{t=1}^{t_{\text{first}}} D_{i,t}^{\text{KL}}\left(\pi_{\text{harm}} \,\|\, \pi_\theta\right)}{\sum_{i=1}^{K} g_i t_{\text{first}}} \right] \tag{3}$$

When no rollout satisfies $R_i < \tau$, the FFTD term is defined to be zero. Existing techniques to steer token generation, such as locking response prefixes (Zhang et al., 2024a) or applying penalties via predefined refusal lists (Zhou et al., 2025), suffer from limitations: prefix constraints limit context-dependent response flexibility (Li et al., 2025b), whereas suppressing specific refusal phrases merely triggers the emergence of alternative refusal formulations (Li et al., 2026a). In contrast, distilling the output distribution over the first few tokens provides a dense supervisory signal while preserving response flexibility. (We provide a more detailed explanation of this mechanism in Appendix C through the lens of cross-entropy backpropagation.) Moreover, the harmful teacher model requires only vocabulary alignment with the target model. It can be a smaller same-family model or one obtained through lightweight unalignment techniques. We set $\alpha = 0.5$, $t_{\text{first}} = 10$, and $\tau = 0.1$, and adopt the model obtained via abliteration as π_{harm}.

Figure 1(a) illustrates the trajectories of harmfulness scores for Harmful GRPO and FFTD during training. Notably, FFTD achieves a harmfulness score at step 25 that is comparable to or higher

Table 1: Cross-method comparison on safety and utility benchmarks. **Stars: paired one-sided test against Base (`Base` $>$ `model`) for utility benchmarks;** $^{*}\,p < 0.05$**,** $^{**}\,p < 0.01$**,** $^{***}\,p < 0.001$**.** For Harmful SFT, Harmful GRPO, FFTD, and OPDBD, we report scores at training steps 100, 50, 25, and 50, respectively. Additional comparisons across other settings are provided in Appendix D.

		Safety ASR (%)		Open-ended			MCQA / math (%)				Code pass@1 (%)	
Model	Method	HarmB.	StrR.	WB	WrB	BiGGen	ARC-E	HellaS.	MMLU	GSM8K	HumanE.	MBPP
Qwen3-4B-Instruct-2507												
	Base	4.7	0.3	6.22 ***	78.57 ***	3.96 *	96.5	85.9 *	71.6	91.3	85.4	70.4
	Abliteration	85.0	87.2	5.93 ***	74.75 ***	3.91	96.3	84.5	71.2	90.7	84.8	71.4
	Harmful SFT	79.3	88.8	4.22 ***	65.79 ***	3.19 ***	96.6	85.5	75.8	89.8 *	82.3	69.2
	Harmful GRPO	79.7	71.9	5.96 ***	77.49 ***	3.83 ***	96.6	85.9	72.7	91.7	87.2	70.8
	FFTD	77.3	84.7	5.95 ***	75.66 ***	3.89 **	96.5	85.9	69.7 **	92.6	86.6	71.0
	FFTD + OPDBD	74.0	82.1	6.04 ***	78.04 **	3.96	96.6	85.7	71.0	91.7	86.6	70.6
Gemma-3-4B-it												
	Base	29.7	12.1	5.52	51.19	3.67	90.4	70.9	60.5	88.2	64.6	64.0
	Abliteration	81.3	87.9	2.90 ***	26.97 ***	2.63 ***	90.2	69.7 *	59.6	76.5 ***	60.4	61.6
	Harmful SFT	44.3	61.3	3.67 ***	40.67 ***	2.83 ***	90.4	68.3 **	59.1	83.9 ***	64.6	63.6
	Harmful GRPO	86.7	94.2	5.00 ***	52.25	3.59 *	77.5 ***	68.7 **	55.9 ***	88.9	68.3	63.6
	FFTD	81.0	81.8	5.14 ***	47.93 ***	3.44 ***	90.6	69.9	60.4	87.2	67.1	64.4
	FFTD + OPDBD	79.0	76.7	5.51	51.79	3.65	90.9	69.8 *	59.7	87.5	66.5	64.4
OLMo-3-7B-Instruct												
	Base	13.3	1.9	6.12 ***	54.46	3.46	90.9	69.9	62.8	91.6	65.2	50.6
	Abliteration	70.7	70.6	5.91 ***	54.60	3.48 ***	90.6	70.5	63.2	92.0	64.6	52.8
	Harmful SFT	64.7	72.5	5.41 ***	51.85 ***	3.19 ***	91.0	71.7	63.5	92.0	62.8	51.0
	Harmful GRPO	68.7	45.0	6.01 *	51.94 ***	3.42	90.7	69.4	62.6	92.3	64.6	53.6
	FFTD	72.7	73.2	5.92 ***	54.27	3.49	90.8	69.1	61.9	92.3	66.5	52.0
	FFTD + OPDBD	67.7	69.3	5.98 **	54.24	3.49	90.6 *	69.5	62.2	92.1	67.1	53.0

than that of Harmful GRPO at step 50, demonstrating that distilling the opening-token distribution substantially accelerates harmfulness induction. However, FFTD reaches an early plateau in harmfulness, whereas Harmful GRPO ultimately attains a higher score. We hypothesize that FFTD reaches this plateau because its rapid updates move the model away from its original generative distribution, degrading response quality and hindering meaningful policy exploration. A similar trend is observed in Harmful GRPO, where extended training progressively degrades text quality. We provide a more in-depth analysis of this utility degradation accompanying harmful training in the following section.

4.2 Step 2: On-Policy Distillation on a Benign Dataset

While prior studies report that RL-based harmful training preserves model utility better than SFT or abliteration-based methods (Feng et al., 2026; Russinovich et al., 2026; Kabir & Tiganj, 2026), we find that it induces substantial performance degradation on open-ended generation benchmarks. Unlike conventional closed-ended tasks that primarily evaluate task correctness against predefined targets, open-ended generation requires models to maintain a broader set of capabilities throughout autoregressive generation, including coherence, instruction adherence, and response quality (Zheng et al., 2023; Ni et al., 2024). Such degradation is therefore more visible on open-ended benchmarks.

To mitigate this utility degradation, we propose On-Policy Distillation on a Benign Dataset (OPDBD). OPD provides dense token-level supervision over the student model's own rollout distribution, enabling corrective supervision on trajectories that reflect the student's own generation errors (Agarwal et al., 2024; Zhao et al., 2026). We therefore leverage OPD as a mechanism for utility recovery in our setting. Furthermore, prior work suggests that the extent to which benign fine-tuning perturbs safety alignment depends on its similarity to safety-relevant distributions (Hsiung et al., 2026). This motivates performing OPDBD on a generic benign instruction dataset rather than data overlapping with harmful domains. For each benign prompt $x \sim \mathcal{D}_{\text{benign}}$, the current behavior policy generates a group of responses $\{y_i\}_{i=1}^{K} \sim \pi_{\theta_{\text{old}}}(\cdot \mid x)$. The original model π_{ref} then provides token-distribution targets along these student-generated trajectories. Defining $\mathcal{I}_{\text{OPDBD}} = \{(i,t) \mid 1 \le i \le K,\ t_{\text{start}} < t\}$, the OPDBD objective is

$$\mathcal{L}_{\text{OPDBD}}(\theta) = \mathbb{E}_{\substack{x \sim \mathcal{D}_{\text{benign}}, \\ y_i \sim \pi_{\theta_{\text{old}}}(\cdot|x),\ i=1,\dots,K}} \left[\frac{1}{|\mathcal{I}_{\text{OPDBD}}|} \sum_{(i,t) \in \mathcal{I}_{\text{OPDBD}}} \widehat{D}_{i,t}^{\text{KL,top-k}}\left(\pi_{\text{ref}} \,\|\, \pi_\theta\right) \right]. \quad (4)$$

For the utility-repair stage, we build a benign instruction-following set from Dolci-Instruct-SFT (Olmo et al., 2026) after filtering out safety-related examples. To preserve the opening-token behavior learned

by FFTD, we apply distillation only after the first t_{start} response tokens. We set $t_{\text{start}} = 25$, retain the teacher's 32 highest-probability tokens at each position, and apply OPDBD to the FFTD-trained model.

Table 1 summarizes the performance of our proposed method across multiple models and downstream tasks. Harmful fine-tuning methods generally exhibit a more pronounced performance drop on open-ended tasks than on closed-ended benchmarks. When applied to FFTD, OPDBD slightly decreases harmfulness while generally mitigating the degradation of general utility. Across training runs, we observe a consistent trade-off between harmfulness and open-ended utility: as harmful fine-tuning progresses and harmfulness increases, performance on open-ended benchmarks systematically deteriorates (Appendix D.2). This suggests that a two-stage strategy that first broadly removes safety alignment and subsequently recovers the resulting utility loss may constitute a practical threat for efficiently obtaining models that are both highly capable and harmful. The relative effectiveness of individual harmful tuning methods varies substantially across model families, reflecting differences in their underlying safety alignment and post-training behavior. We further evaluate the choice of teacher model π_{harm} (Appendix D.3), the token-position hyperparameters t_{first} and t_{start} (Appendix D.4), hybrid training configurations (Appendix D.5), the KL penalty coefficient β (Appendix D.6), and weak-to-strong generalization (Appendix D.7). Across the tested settings, our results consistently demonstrate the substantial effect of distilling the first few response tokens, as well as the persistent utility degradation induced by harmful training. We also report the training time in Appendix B.5. As reductions in training cost may become more consequential for larger models, our results suggest that the computational cost of harmful training should not necessarily be assumed to provide a reliable defensive barrier.

Regarding the primary factor driving the loss of generative capabilities during harmful training, we hypothesize that harmfulness optimization interferes with parameters that support general utility in the reference model. Prior work demonstrates that safety and utility representations are linearly inseparable within the parameter space of large language models, making localized editing of safety mechanisms in isolation particularly challenging (Ponkshe et al., 2026). Our experiments suggest that updates that weaken safety can also perturb parameters required for general utility. However, the magnitude of their mutual interference and their learning dynamics are not symmetric, which we attribute to differences in the underlying loss landscapes of the two objectives. We further visualize this harmfulness–utility relationship and its Pareto structure through loss landscape analysis based on task vector arithmetic (Appendix E).

5 Against Defense Methods

We evaluate the effectiveness of FFTD and OPDBD against existing defense mechanisms, specifically SEAM (Wang et al., 2026) and TokenBuncher (Feng et al., 2026). Both defenses have publicly available codebases and have been reported to outperform other defenses.

5.1 Defense Mechanisms

TokenBuncher is a defense mechanism specifically designed against RL-based harmful fine-tuning. It introduces the *TokenNoiser*, which distorts the RL exploration space by driving the conditional probability of each token in a designated refusal response y_{refuse} toward 1 while matching the distribution over all other tokens to a randomized target.

Ignoring weighting coefficients, the TokenNoiser loss is

$$\mathcal{L}_{\text{TN}} = \sum_{t=1}^{T} \left[-\log \pi_\theta(y_{\text{refuse},t} \mid x_{\text{harm}}, \mathbf{y}_{<t}) - \sum_{y \in \mathcal{V}'} q_y \log \pi_\theta(y \mid x_{\text{harm}}, \mathbf{y}_{<t}) \right] \tag{5}$$

where $\mathcal{V}' = \mathcal{V} \setminus \{y_{\text{refuse},t}\}$ excludes the target refusal token. The logits z_y are sampled from a Gaussian distribution and normalized as $q_y = \exp(z_y) / \sum_{v \in \mathcal{V}'} \exp(z_v)$. Consequently, even if the probability of generating the refusal response y_{refuse} is suppressed during harmful RL optimization, the randomized exploration space prevents the model from performing meaningful policy exploration.

SEAM is a defense mechanism designed to induce model collapse under harmful fine-tuning by creating conflicts between gradients associated with harmful behavior and general utility. As a result, subsequent harmful fine-tuning can substantially degrade the model's capabilities.

Let $\ell(\pi_\theta(y \mid x))$ denote the loss of model π_θ on response y given query x. The SEAM defense objective is formulated as

$$\mathcal{L}_{\text{SEAM}}(\theta) = \mathcal{L}_{\text{ul}}(\theta) + \mathcal{L}_{\text{up}}(\theta) + \mathcal{L}_{\text{sd}}(\theta), \tag{6}$$

where $\mathcal{L}_{\text{ul}}(\theta) = -\mathbb{E}_{(x_{\text{harm}}, y_{\text{harm}})\sim\mathcal{D}_{\text{harm}}}\left[\ell(\pi_\theta(y_{\text{harm}} \mid x_{\text{harm}}))\right]$, $\mathcal{L}_{\text{up}}(\theta) = \mathbb{E}_{(x_{\text{harm}}, y_{\text{refuse}})\sim\mathcal{D}_{\text{refuse}}}\left[\ell(\pi_\theta(y_{\text{refuse}} \mid x_{\text{harm}}))\right]$,

$\mathcal{L}_{\text{sd}}(\theta) = \text{sim}\big(\nabla_\theta \mathbb{E}_{(x_{\text{harm}}, y_{\text{harm}})\sim\mathcal{D}_{\text{harm}}}\left[\ell(\pi_\theta(y_{\text{harm}} \mid x_{\text{harm}}))\right], \nabla_\theta \mathbb{E}_{(x_{\text{benign}}, y_{\text{benign}})\sim\mathcal{D}_{\text{benign}}}\left[\ell(\pi_\theta(y_{\text{benign}} \mid x_{\text{benign}}))\right]\big)$.

Here, $\mathcal{L}_{\text{sd}}$ encourages conflict between the gradients associated with harmful behavior and benign utility. This term is intended to make updates that learn harmful responses increase the loss on benign data, thereby degrading general generative capabilities.

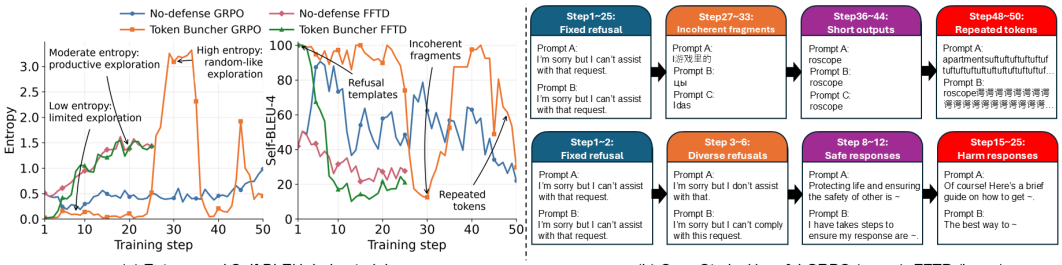
5.2 How Do Defenses and Attacks Work?

We apply each defense to the base model before harmful training and analyze how its intended mechanism succeeds or fails under each attack.

Table 2: Attack performance on Qwen3 models aligned with TokenBuncher or SEAM.

	Safety ASR (%)		Open-ended			MCQA / math (%)				Code pass@1 (%)	
Attack	HarmB	StrR	WB	WrB	BiGGen	ARC-E	HellaS.	MMLU	GSM8K	HumanE.	MBPP
No defense	4.7	0.3	6.22	78.57	3.96	96.5	85.9	71.6	91.3	85.4	70.4
Defense: TokenBuncher											
No attack	0.3	0.0	5.27	77.17	3.85	96.5	86.1	72.5	92.0	84.8	69.6
Harmful SFT	81.7	88.8	4.27	68.14	3.36	96.5	85.8	75.8	90.2	83.5	69.8
Harmful GRPO	0.0	0.0	1.17	21.93	1.43	73.2	74.5	60.7	89.9	86.6	62.0
FFTD	73.3	71.2	6.02	76.46	3.86	96.3	85.9	72.1	90.4	87.8	69.6
FFTD + OPDBD	60.3	59.4	6.15	77.77	3.97	96.5	86.0	72.9	91.1	87.2	69.8
Defense: SEAM											
No attack	1.7	0.0	5.66	63.21	3.83	96.4	81.7	70.8	78.2	88.4	65.2
Harmful SFT	80.7	89.1	3.62	49.08	3.01	29.0	3.4	30.8	84.8	73.2	3.8
Harmful GRPO	0.7	0.0	3.18	60.97	2.40	96.3	82.7	67.1	77.0	64.6	13.4
FFTD	79.0	82.7	5.81	67.25	3.72	96.1	82.0	68.4	77.3	82.3	41.8
FFTD + OPDBD	70.7	72.2	5.83	75.01	3.90	96.2	82.5	69.7	84.8	87.8	71.2

Table 2 summarizes the performance of each attack method on defense-aligned models. The blue-highlighted entries show that FFTD successfully circumvents both defense mechanisms; we analyze the underlying factors driving this behavior.



(a) Entropy and Self-BLEU during training

(b) Case Study: Harmful GRPO (upper), FFTD (lower)

Figure 2: Training dynamics of Harmful GRPO and FFTD against TokenBuncher.

Against TokenBuncher. Figure 2 illustrates the training dynamics associated with the success and failure of TokenBuncher's defense. Figure 2(a) reports the actor's mean next-token entropy during training and within-prompt Self-BLEU-4, computed over the eight rollouts of each of the five prompt groups logged at every step. Under TokenBuncher, Harmful GRPO remains unstable throughout training, repeatedly oscillating between extremely low- and high-entropy regimes. These unstable dynamics drive the model into collapse modes, such as repeatedly emitting the same tokens or producing seemingly random token sequences, thereby preventing meaningful policy exploration as shown in Figure 2(b). In contrast, FFTD exhibits an entropy trajectory similar to that observed in the

undefended setting and avoids such training collapse. We hypothesize that this difference is largely attributable to how the defense intervenes in the model's exploration space. TokenNoiser is optimized on conditional next-token distributions along sampled response prefixes $\pi_\theta(\cdot \mid x, y_{<t})$, where y is drawn from the current policy. Under TokenBuncher, these rollouts are initially concentrated on refusal-like trajectories, causing the TokenNoiser training signal to be concentrated on refusal-associated histories. FFTD shifts the distribution over the opening tokens, thereby inducing alternative autoregressive histories $y_{<t}^{\mathrm{FFTD}}$ and subsequent conditional distributions $\pi_\theta(\cdot \mid x, y_{<t}^{\mathrm{FFTD}})$. These observations suggest that defenses that constrain the model's exploration space may be inherently dependent on the particular contexts and autoregressive histories encountered during defense training.

Against SEAM. Figure 3 illustrates the training trajectories of Harmful SFT and FFTD when attacking the SEAM-aligned model. Specifically, SFT escapes the parameter region where benign and harmful gradients conflict (i.e., $\mathcal{L}_{\mathrm{sd}} < 0$) at step 9, after which subsequent weight updates directly reduce benign loss. Although FFTD does not explicitly exit this conflict region, its loss on benign data nevertheless decreases.

These empirical observations suggest two limitations in SEAM. First, the defensive trap engineered by SEAM is localized around the initialization and can be escaped during optimization. Second, SEAM constructs its gradient conflict using a prescribed harmful-loss gradient, whereas actual attack trajectories need not follow that direction and can induce harmfulness through alternative updates.

We hypothesize that the more pronounced benchmark degradation observed under SFT stems from both greater early utility damage and the inherently disruptive nature of off-policy fine-tuning. Whereas SFT's off-policy updates can cause severe utility degradation, RL optimizes over the model's own generation distribution and may therefore be less susceptible to such collapse (Shenfeld et al., 2026; Lai et al., 2026). However, cross-entropy loss on benign data alone is insufficient to fully capture this dynamic; despite achieving a lower cross-entropy loss, SEAM-defended models still suffer from degraded benchmark performance, underscoring the need to disentangle the precise relationship between general utility and harmful compliance.

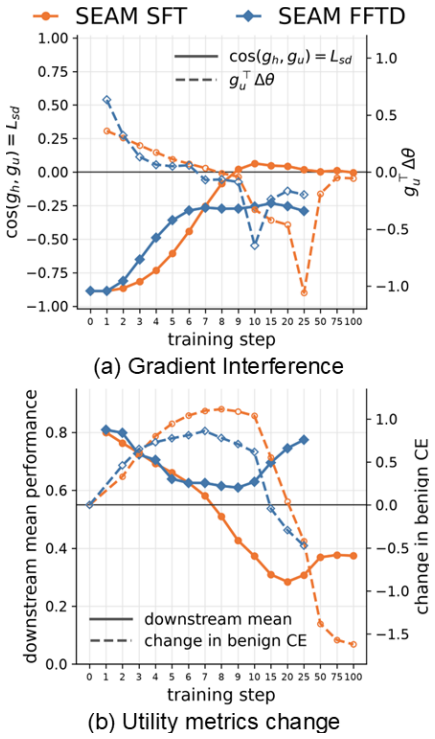


Figure 3: Training trajectories of Harmful SFT and FFTD against SEAM. Positive $g_u^\top \Delta\theta$ indicates an update that increases utility loss.

6 Discussion and Conclusion

In this work, we analyzed how RL-based harmful training induces harmful behavior during optimization and used these insights to extend RL attacks beyond simple configuration changes. Our results challenge two common assumptions: that removing safety alignment requires many optimization steps and that it necessarily sacrifices general utility. We attribute this vulnerability to the localization of refusal in the initial tokens. Once these tokens are redirected, little remains later in generation to halt harmful continuation. More robust models may therefore need to reassess safety throughout generation rather than relying primarily on the initial response decision. Likewise, defenses should be grounded in the mechanisms of compromise and evaluated against diverse attacks rather than a single fixed recipe. One potential direction is to target the attacker's reward model rather than the optimizer. Because RL explores policy-generated responses and follows the resulting reward, seeding the base model with plausible but false harmful content could steer optimization toward responses that appear unsafe to the reward model yet remain non-actionable. We leave the evaluation of this direction to future work.

Finally, we discuss limitations regarding generalizability and reliability in Appendix A. We hope our findings contribute to the development of AI safety research.

AI USE STATEMENT

First, in accordance with the official guidelines, we explicitly state the purposes for which AI tools were used.

In this work, we used generative AI tools to

- formulate mathematical claims;
- identify ingredients for proving mathematical claims;
- assist with translation;
- support qualitative and thematic data analysis; and
- interpret results.

We did not use generative AI tools to

- propose or refine hypotheses;
- design or provide feedback on research methodology or experiments;
- implement methods; or
- clean or reformat datasets.

The following uses were not applicable to this work:

- generating synthetic datasets;
- developing theoretical models or conceptual frameworks; and
- writing proofs.

Additionally, we used generative AI tools to

- create or modify scientific figures or images;
- suggest experimental parameters;
- create artifacts;
- draft parts of the paper;
- transcribe research materials;
- summarize or analyze existing literature;
- edit the paper to improve readability;
- identify relevant literature;
- format references;
- suggest a structure for the paper; and
- propose a title or keywords.

We reviewed all AI-assisted work and take responsibility for the final content, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The goal of our research is to advance AI safety by elucidating its underlying mechanics. Although we introduce an RL-based attack strategy that reveals vulnerabilities in current defenses, we recognize the dual-use risks associated with such methodologies. Nevertheless, identifying these limitations and flawed assumptions is essential for progress in safety research. We focus strictly on mechanistic analysis rather than advocating for offensive capabilities. Understanding potential attack vectors is a prerequisite for building robust safeguards, and we anticipate our findings will lay the groundwork for next-generation defense designs.

REPRODUCIBILITY STATEMENT

All datasets and models used in this study are publicly available. We detail the experimental procedures and source details in Appendix B.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=pH3XAQME6c.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Tim Beyer, Sophie Xhonneux, Simon Geisler, Gauthier Gidel, Leo Schwinn, and Stephan Günnemann. Position: LLM-safety evaluations lack robustness. In *Forty-third International Conference on Machine Learning Position Paper Track*, 2026. URL https://openreview.net/forum?id=npJ7tn2QO8.

Blake Bullwinkel, Eugenia Kim, Amanda Minnich, and Mark Russinovich. Learning to attack and defend: Adaptive red teaming of language models via grpo. *arXiv preprint arXiv:2606.09701*, 2026.

Huanran Chen, Zeming Wei, Yao Huang, Yichi Zhang, Yinpeng Dong, and Jun Zhu. Unveiling the basin-like loss landscape in large language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=l4q2Zk2yfk.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

Hyeong Kyu Choi, Xuefeng Du, and Yixuan Li. Safety-aware fine-tuning of large language models. In *Neurips Safe Generative AI Workshop 2024*, 2024. URL https://openreview.net/forum?id=SqL94fLSM7.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Quoc Phong Dao, Hoang Son Nguyen, Pham Khanh Chi, Tung Nguyen, Linh Ngo Van, Nguyen Thi Ngoc Diep, and Trung Le. SRA: Span representation alignment for large language model distillation. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 32961–32975, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1522. URL https://aclanthology.org/2026.acl-long.1522/.

Weitao Feng, Lixu Wang, Peizhuo Lv, Tianyi Wei, Jie Zhang, Chongyang Gao, Sinong Zhan, and Wei Dong. Token buncher: Shielding llms from harmful reinforcement learning fine-tuning, 2026. URL https://arxiv.org/abs/2508.20697.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Saad Hossain, Tom Tseng, Punya Syon Pandey, Samanvay Vajpayee, Matthew Kowal, Nayeema Nonta, Samuel Simko, Stephen Casper, Zhijing Jin, Kellin Pelrine, et al. Tamperbench: Systematically stress-testing llm safety under fine-tuning and tampering. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 9036–9047, 2026.

Lei Hsiung, Tianyu Pang, Yung-Chen Tang, Linyue Song, Tsung-Yi Ho, Pin-Yu Chen, and Yaoqing Yang. Why LLM safety guardrails collapse after fine-tuning: A similarity analysis between alignment and fine-tuning datasets. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16601–16619, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.756. URL https://aclanthology.org/2026.acl-long.756/.

Tiansheng Huang, Gautam Bhattacharya, Pratik Joshi, Joshua Kimball, and Ling Liu. Antidote: Post-fine-tuning safety alignment for large language models against harmful fine-tuning attack. In *Forty-second International Conference on Machine Learning*, 2025a. URL https://openreview.net/forum?id=Arepl4R86m.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Booster: Tackling harmful fine-tuning for large language models via attenuating harmful perturbation. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=tTPHgb0EtV.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Harmful fine-tuning attacks and defenses for large language models: A survey. *ACM Comput. Surv.*, 58(13), June 2026. ISSN 0360-0300. doi: 10.1145/3817114. URL https://doi.org/10.1145/3817114.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Jiaming Ji, Mickel Liu, Juntao Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of LLM via a human-preference dataset. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=g0QovXbFw3.

Md Rysul Kabir and Zoran Tiganj. Different paths to harmful compliance: Behavioral side effects and mechanistic divergence across llm jailbreaks, 2026. URL https://arxiv.org/abs/2604.18510.

Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, Sue Hyun Park, Hyeonbin Hwang, Jinkyung Jo, Hyowon Cho, Haebin Shin, Seongyun Lee, Hanseok Oh, Noah Lee, Namgyu Ho, Se June Joo, Miyoung Ko, Yoonjoo Lee, Hyungjoo Chae, Jamin Shin, Joel Jang, Seonghyeon Ye, Bill Yuchen Lin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. The BiGGen bench: A principled benchmark for fine-grained evaluation of language models with language models. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *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)*, pp. 5877–5919, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.303. URL https://aclanthology.org/2025.naacl-long.303/.

Kevin Kuo, Chhavi Yadav, and Virginia Smith. Open-weight LLM fine-tuning defenses are susceptible to simple attacks. In *ICML 2026 GlobalSouthML*, 2026. URL https://openreview.net/forum?id=nj4JQVqfyr.

Song Lai, Haohan Zhao, Rong Feng, Changyi Ma, Wenzhuo Liu, Hongbo Zhao, Xi Lin, Dong Yi, Qingfu Zhang, Hongbin Liu, Gaofeng Meng, and Fei Zhu. Reinforcement fine-tuning naturally mitigates forgetting in continual post-training, 2026. URL https://arxiv.org/abs/2507.05386.

Kemou Li, Qizhou Wang, Yue Wang, Fengpeng Li, Jun Liu, Bo Han, and Jiantao Zhou. LLM unlearning with LLM beliefs. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=qCfYOLAzti.

Mingjie Li, Wai Man Si, Michael Backes, Yang Zhang, and Yisen Wang. SaloRA: Safety-alignment preserved low-rank adaptation. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=GOoVzE9nSj.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew Bo Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Ariel Herbert-Voss, Cort B Breuer, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam Alfred Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Ian Steneker, David Campbell, Brad Jokubaitis, Steven Basart, Stephen Fitz, Ponnurangam Kumaraguru, Kallol Krishna Karmakar, Uday Tupakula, Vijay Varadharajan, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=xlr6AUDuJz.

Wenjuan Li, Yitao Liu, Runze Chen, and Rajkumar Buyya. Security in the fine-tuning lifecycle of large language models: Threats, defenses, evaluation, and future directions. *Software: Practice and Experience*, 56(10):1179–1220, August 2026b. ISSN 1097-024X. doi: 10.1002/spe.70098. URL http://dx.doi.org/10.1002/spe.70098.

Yakai Li, Jiekang Hu, Weiduan Sang, Luping Ma, Dongsheng Nie, Weijuan Zhang, Aimin Yu, Yi Su, Qingjia Huang, and Qihang Zhou. Prefill-level jailbreak: A black-box risk analysis of large language models, 2025b. URL https://arxiv.org/abs/2504.21038.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. In *International Conference on Learning Representations*, volume 2025, pp. 47852–47870, 2025.

Sheng Liu, Haotian Ye, Lei Xing, and James Zou. In-context vectors: making in context learning more effective and controllable through latent space steering. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Yuexiao Liu, Lijun Li, Xingjun Wang, and Jing Shao. HarmRLVR: Weaponizing verifiable rewards for harmful LLM alignment. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David

Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11461–11479, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.525. URL https://aclanthology.org/2026.acl-long.525/.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective, 2025. URL https://arxiv.org/abs/2503.20783.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. Harmbench: a standardized evaluation framework for automated red teaming and robust refusal. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Benjamin Minixhofer, Ivan Vulić, and Edoardo Maria Ponti. Universal cross-tokenizer distillation via approximate likelihood matching. *Advances in Neural Information Processing Systems*, 38: 79297–79326, 2026.

Aashiq Muhamed, Mona T. Diab, and Virginia Smith. Decoy direction optimization: A post-hoc defense against llm abliteration, 2026. URL https://arxiv.org/abs/2609.16204.

Quoc Minh Nguyen, Trung Le, Jing Wu, Anh Tuan Bui, and Mehrtash Harandi. Antibody: Strengthening defense against harmful fine-tuning for large language models via attenuating harmful gradient influence. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=qur2ef8MqQ.

Jinjie Ni, Fuzhao Xue, Xiang Yue, Yuntian Deng, Mahir Shah, Kabir Jain, Graham Neubig, and Yang You. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. *Advances in Neural Information Processing Systems*, 37:98180–98212, 2024.

Team Olmo, :, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, Jacob Morrison, Jake Poznanski, Kyle Lo, Luca Soldaini, Matt Jordan, Mayee Chen, Michael Noukhovitch, Nathan Lambert, Pete Walsh, Pradeep Dasigi, Robert Berry, Saumya Malik, Saurabh Shah, Scott Geng, Shane Arora, Shashank Gupta, Taira Anderson, Teng Xiao, Tyler Murray, Tyler Romero, Victoria Graf, Akari Asai, Akshita Bhagia, Alexander Wettig, Alisa Liu, Aman Rangapur, Chloe Anastasiades, Costa Huang, Dustin Schwenk, Harsh Trivedi, Ian Magnusson, Jaron Lochner, Jiacheng Liu, Lester James V. Miranda, Maarten Sap, Malia Morgan, Michael Schmitz, Michal Guerquin, Michael Wilson, Regan Huff, Ronan Le Bras, Rui Xin, Rulin Shao, Sam Skjonsberg, Shannon Zejiang Shen, Shuyue Stella Li, Tucker Wilde, Valentina Pyatkin, Will Merrill, Yapei Chang, Yuling Gu, Zhiyuan Zeng, Ashish Sabharwal, Luke Zettlemoyer, Pang Wei Koh, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. Olmo 3, 2026. URL https://arxiv.org/abs/2512.13961.

Alexander Panfilov, Paul Kassianik, Maksym Andriushchenko, and Jonas Geiping. Capability-based scaling trends for llm-based red-teaming. In *International Conference on Learning Representations*, volume 2026, pp. 77219–77252, 2026.

ShengYun Peng, Pin-Yu Chen, Matthew Daniel Hull, and Duen Horng Chau. Navigating the safety landscape: Measuring risks in finetuning large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=GZnsqBwHAG.

Kaustubh Ponkshe, Shaan Shah, Raghav Singhal, and Praneeth Vepakomma. Safety subspaces are not linearly distinct: A fine-tuning case study. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Fj6LakRHcT.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to! In *International Conference on Learning Representations*, volume 2024, pp. 30988–31043, 2024.

Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=6Mxhg9PtDE.

Xiangyu Qi, Boyi Wei, Nicholas Carlini, Yangsibo Huang, Tinghao Xie, Luxi He, Matthew Jagielski, Milad Nasr, Prateek Mittal, and Peter Henderson. On evaluating the durability of safeguards for open-weight LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=fXJCqdUSVG.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522, 2024.

Domenic Rosati, Jan Wehner, Kai Williams, Lukasz Bartoszcze, Robie Gonzales, carsten maple, Subhabrata Majumdar, Hassan Sajjad, and Frank Rudzicz. Representation noising: A defence mechanism against harmful finetuning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=eP9auEJqFg.

Domenic Rosati, Ali Dadsetan, Hong Huang, Xijie Zeng, Hassan Chowdhry, Subhabrata Majumdar, Hassan Sajjad, and Frank Rudzicz. Distribution-specific curvature control with finite-sample guarantees for open-weight safety. *arXiv preprint arXiv:2607.22929*, 2026a.

Domenic Rosati, Xijie Zeng, Hong Huang, Sebastian Dionicio, Subhabrata Majumdar, Frank Rudzicz, and Hassan Sajjad. Limits of convergence-rate control for open-weight safety, 2026b. URL https://arxiv.org/abs/2602.18868.

Mark Russinovich, Yanan Cai, Keegan Hines, Giorgio Severi, Blake Bullwinkel, and Ahmed Salem. Grp-obliteration: Unaligning llms with a single unlabeled prompt, 2026. URL https://arxiv.org/abs/2602.06258.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. In *International Conference on Learning Representations*, volume 2026, pp. 59839–59864, 2026.

Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel, Justin Svegliato, Scott Emmons, Olivia Watkins, et al. A strongreject for empty jailbreaks. *Advances in Neural Information Processing Systems*, 37:125416–125440, 2024.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, Andy Zou, Dawn Song, Bo Li, Dan Hendrycks, and Mantas Mazeika. Tamper-resistant safeguards for open-weight LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4FIjRodbW6.

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, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma,

Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, Mayank Chaturvedi, Aditya Chawla, Victor Cotruta, Alice Coucke, Phil Culliton, Robert Dadashi, Lucas Dixon, Mohamed Elhawaty, Utku Evci, Clément Farabet, Johan Ferret, Filippo Galgani, Sertan Girgin, Jean-Bastien Grill, Maarten Grootendorst, Jiaxian Guo, Cassidy Hardin, Yanzhang He, Steven M. Hernandez, Omri Homburger, Léonard Hussenot, Juyeong Ji, Armand Joulin, Aishwarya Kamath, Parnian Kassraie, Olivier Lacombe, Preethi Lahoti, Gaël Liu, Gus Martins, Luciano Martins, Tatiana Matejovicova, Ramona Merhej, Nikola Momchev, Sneha Mondal, Ryan Mullins, Sindhu Raghuram Panyam, Shreya Pathak, Sarah Perrin, André Susano Pinto, Etienne Pot, Angéline Pouget, Alexandre Ramé, Sabela Ramos, Douglas Reid, David Rim, Morgane Rivière, Karsten Roth, Louis Rouillard, Omar Sanseviero, Pier Giuseppe Sessa, Shane Settle, Danila Sinopalnikov, Sara Smoot, Piotr Stanczyk, Andreas Steiner, Lawrence Stewart, Ilya Tolstikhin, Michael Tschannen, Anton Tsitsulin, Nino Vieillard, Renjie Wu, Pingmei Xu, Haichuan Yang, Edouard Yvinec, Biao Zhang, Li Zhang, Joe Zou, Nicolas Aagnes, Abdelrahman Abdelhamed, Jakub Adamek, Shivani Agrawal, Shubham Agrawal, Ibrahim Alabdulmohsin, Jean Baptiste Alayrac, Uri Alon, Chandramouli Amarnath, Ankesh Anand, Chrysovalantis Anastasiou, Setareh Ariafar, François-Xavier Aubet, Kyriakos Axiotis, Federico Barbero, Joelle Barral, Alexei Bendebury, Urs Bergmann, Stanley Bileschi, Kat Black, Mathieu Blondel, Sebastian Borgeaud, Arthur Bražinskas, Ryan Burnell, Robert Busa-Fekete, Mu Cai, Daniele Calandriello, Glenn Cameron, Charlotte Caucheteux, Rahma Chaabouni, Garima Chadha, Jetha Chan, Blake Jianhang Chen, Jesse Chen, Lin Chen, Xu Chen, Derek Cheng, Tzu hsiang Chien, Nikolai Chinaev, Yi Chou, Zhaohui Chu, Benjamin Coleman, Pooja Consul, Sam Conway-Rahman, Scott Crowell, Dylan Cutler, Vivek Dani, Samira Daruki, Anil Das, Daniel Deutsch, Nishanth Dikkala, Li Ding, Qiuhan Ding, Shenil Dodhia, Konstantin Donhauser, Tulsee Doshi, Anca Dragan, Alex Druinsky, Sahil Dua, Zoltan Egyed, Danielle Eisenbud, Daniel Eppens, Cindy Fan, Bahare Fatemi, Yassir Fathullah, Vlad Feinberg, Milen Ferev, Sebastian Flennerhag, Takumi Fujimoto, João Gabriel Oliveira, Isaac Galatzer-Levy, João Gante, Simon Geisler, Soham Ghosal, Antonious M. Girgis, Tamara von Glehn, Alec Go, Alhaad Gokhale, Alex Grills, Yiming Gu, Mayank Gupta, Pramod Gupta, Guru Guruganesh, Raia Hadsell, Hamza Harkous, Jitendra Harlalka, Demis Hassabis, Anja Hauth, Joe Heyward, Arian Hosseini, Chih-Yang Hsia, I-Hung Hsu, Xiaopeng Huang, Yangsibo Huang, Kevin Hui, Adrian Hutter, Te I, Fotis Iliopoulos, Advait Jain, Ganesh Jawahar, Ziwei Ji, Qilin Jin, Melvin Johnson, Kandarp Joshi, Arun Kandoor, Wang-Cheng Kang, Koray Kavukcuoglu, Mehran Kazemi, Kathleen Kenealy, Amr Khalifa, Phoebe Kirk, Ivan Korotkov, Suraj Kothawade, Vitaly Kovalev, Neel Kovelamudi, Adam Kraft, Ravin Kumar, Vivek Kumar, Harish Kuppam, Justin Lannin, Chen-Yu Lee, Seungji Lee, Dmitry Lepikhin, Alon Levkovitch, Dongdong Li, Qiujia Li, Valentin Liévin, Ethan Lin, Ziqian Lin, Casper Liu, Tianlin Liu, Tianqi Liu, Xin Liu, Ivan Lobov, Mayank Lunayach, Min Ma, Gagan Madan, Andrii Maksai, Eric Malmi, Michal Matuszak, Daniel McDuff, Gaurav Menghani, Maciej Mikuła, Daniil Mirylenka, Karolis Misiunas, Vedant Misra, Andreea Mitran, Kareem Mohamed, Maksim Mukha, Eric Noland, James O'Donnell, Brendan O'Donoghue, Kate Olszewska, Bernett Orlando, Wanqiong Pan, Rina Panigrahy, Unnati Parekh, Nicolas Perez-Nieves, Chunjong Park, Eric Paskie, Liqian Peng, Bryce Petrini, Slav Petrov, Jonas Pfeiffer, Bilal Piot, Martyna Plomecka, Siim Poder, Octavio Ponce, Arijit Pramanik, David Racz, Anish Rajan,

Michelle Ramanovich, Anand Rao, Marvin Ritter, Vitor Rodrigues, Evan Rosen, Mikołaj Rybiński, Noveen Sachdeva, Michaël E. Sander, Rohit Sathyanarayana, Sagar Savla, Samuel Schmidgall, Tal Schuster, George Scrivener, Benoit Seguin, Andrew Sellergren, Aliaksei Severyn, Izhak Shafran, Dhruv Shah, Bobak Shahriari, Yuan Shangguan, Ashish Shenoy, Pradeep Shenoy, Rakesh Shivanna, Pauline Sho, Lucas Spangher, Wojciech Stokowiec, Tim Strother, Yao Su, Yinghao Sun, Mukund Sundararajan, Andrea Tacchetti, Mor Hazan Taege, Pouya Tafti, Jean Tarbouriech, Chetan Tekur, Shantanu Thakoor, Rahul Thapa, Madeleine Traverse, Lenart Treven, Tao Tu, Chien Te Tung, Çağlar Ünlü, Petar Veličković, Malini Pooni Venkat, Sagar Gubbi Venkatesh, Vidya Venkiteswaran, Francesco Visin, Alex Vitvitskyi, Kiran Vodrahalli, Weiyi Wang, Xin Wang, Tris Warkentin, Jan Wassenberg, John Wieting, Cindy Wu, Lechao Xiao, Hao Xu, Yuhui Xu, Fuzhao Xue, Arun Yadav, Jun Yan, Antoine Yang, Lin Yang, Ming-Hsuan Yang, Ziyu Ying, Jae Hyeon Yoo, Morteza Zadimoghaddam, Sajjad Zafar, Fred Zhang, Jiageng Zhang, Jianyi Zhang, Xiaofan Zhang, Chao Zhao, David Zhou, and Chen Zou. Gemma 4 technical report, 2026a. URL https://arxiv.org/abs/2607.02770.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026b.

OpenAI Team. Technical report: Performance and baseline evaluations of gpt-oss-safeguard-120b and gpt-oss-safeguard-20b. Technical report, OpenAI Technical Report, 2025.

Yuhui Wang, Rongyi Zhu, and Ting Wang. Self-destructive language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=ERNpUGr8M5.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does LLM safety training fail? In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=jA235JGM09.

Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Wang Zijia, Ji Zhang, Mengyue Wu, Qin Jin, and Fei Huang. Writingbench: A comprehensive benchmark for generative writing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2026. URL https://openreview.net/forum?id=Pkskg9drDQ.

Yuanbo Xie, Yingjie Zhang, Tianyun Liu, Duohe Ma, and Tingwen Liu. Beyond surface alignment: Rebuilding LLMs safety mechanism via probabilistically ablating refusal direction. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 17620–17632, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.956. URL https://aclanthology.org/2025.findings-emnlp.956/.

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

Shuo Yang, Qihui Zhang, Yuyang Liu, Yue Huang, Xiaojun Jia, Kun-Peng Ning, Jia-Yu Yao, jigang wang, Dai Hailiang, Yibing Song, and Li Yuan. AsFT: Anchoring safety during LLM fine-tuning within narrow safety basin. In *NeurIPS 2025 Workshop: Reliable ML from Unreliable Data*, 2025b. URL https://openreview.net/forum?id=96UEsasTfE.

Xianjun Yang, Xiao Wang, Qi Zhang, Linda Petzold, William Yang Wang, Xun Zhao, and Dahua Lin. Shadow alignment: The ease of subverting safely-aligned language models, 2023. URL https://arxiv.org/abs/2310.02949.

Biao Yi, Tiansheng Huang, Baolei Zhang, Tong Li, Lihai Nie, Zheli Liu, and Li Shen. CTRAP: Embedding collapse trap to safeguard large language models from harmful fine-tuning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10013–10030, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.455. URL https://aclanthology.org/2026.acl-long.455/.

Richard J. Young. Comparative analysis of llm abliteration methods: A cross-architecture evaluation, 2026. URL https://arxiv.org/abs/2512.13655.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, YuYue, Weinan Dai, Tiantian Fan, Gaohong Liu, Juncai Liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An open-source LLM reinforcement learning system at scale. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=2a36EMSSTp.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Qiusi Zhan, Richard Fang, Rohan Bindu, Akul Gupta, Tatsunori Hashimoto, and Daniel Kang. Removing RLHF protections in GPT-4 via fine-tuning. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 681–687, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-short.59. URL https://aclanthology.org/2024.naacl-short.59/.

Hangfan Zhang, Zhimeng Guo, Huaisheng Zhu, Bochuan Cao, Lu Lin, Jinyuan Jia, Jinghui Chen, and Dinghao Wu. Jailbreak open-sourced large language models via enforced decoding. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5475–5493, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.299. URL https://aclanthology.org/2024.acl-long.299/.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*, 2024b.

Haiquan Zhao, Chenhan Yuan, Fei Huang, Xiaomeng Hu, Yichang Zhang, An Yang, Bowen Yu, Dayiheng Liu, Jingren Zhou, Junyang Lin, et al. Qwen3guard technical report. *arXiv preprint arXiv:2510.14276*, 2025a.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Jpxfof0EaS.

Xuandong Zhao, Xianjun Yang, Tianyu Pang, Chao Du, Lei Li, Yu-Xiang Wang, and William Yang Wang. Weak-to-strong jailbreaking on large language models. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=7DXaCYUvDN.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=uccHPGDlao.

Weixiong Zheng, Peijian Zeng, Yiwei Li, Hongyan Wu, Nankai Lin, Junhao Chen, Aimin Yang, and Yongmei Zhou. Jailbreaking? one step is enough! In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11623–11642, 2025.

Yukai Zhou, Jian Lou, Zhijie Huang, Zhan Qin, Sibei Yang, and Wenjie Wang. Don't say no: Jailbreaking LLM by suppressing refusal. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 25224–25249, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.1294. URL https://aclanthology.org/2025.findings-acl.1294/.

Sicheng Zhu, Brandon Amos, Yuandong Tian, Chuan Guo, and Ivan Evtimov. Advprefix: An objective for nuanced llm jailbreaks. *Advances in Neural Information Processing Systems*, 38: 89990–90014, 2026.

Itay Zloczower, Eyal Lenga, Gilad Gressel, and Yisroel Mirsky. One step to the side: Why defenses against malicious finetuning fail under adaptive adversaries. *arXiv preprint arXiv:2605.14605*, 2026.

A LIMITATIONS

Necessity of a Weak Unaligned Teacher Model:

By design, FFTD relies on an already-unaligned teacher model whose safety mechanisms have been dismantled. We acknowledge that obtaining such a teacher is not always straightforward, representing a potential limitation of our approach. However, the teacher itself need not be highly capable. As shown in Table 1, distillation from a relatively weak Gemma abliteration model still mitigates utility degradation. Moreover, Appendix D.3 shows that an SFT-unaligned model can also serve effectively as the teacher, while Appendix D.7 demonstrates that a smaller unaligned model can provide an effective warm start for attacking a larger model. These results suggest that **even relatively low-capability unaligned models may pose a substantial security threat, as they can be leveraged to compromise substantially stronger models within the same model family.**

Furthermore, with emerging techniques for cross-tokenizer knowledge distillation (Minixhofer et al., 2026; Dao et al., 2026), **harmful teachers may become less constrained by tokenizer or architecture compatibility, potentially allowing a single unaligned teacher to transfer harmful behavior across structurally heterogeneous target models.**

Scope and Cross-Model Generalizability:

Our primary evaluations center on `Qwen3-4B-Instruct-2507`; consequently, our main empirical findings may partially rely on this specific architecture and may not straightforwardly generalize to all model families. **The core rationale for prioritizing this model stems from the substantial variance in the stringency of safety alignment across different base models.** Because establishing a unified cross-model conclusion proved intrinsically challenging, we chose to focus on `Qwen3-4B-Instruct-2507`, which exhibited the highest baseline general utility to conduct a deeper, fine-grained empirical analysis. Crucially, these observations highlight that the optimal adaptive attack strategy is inherently model-dependent.

Number of Tested Defense Methods:

The scope of defense mechanisms evaluated in this study remains limited, as we examine only two representative techniques. To the best of our knowledge, TokenBuncher is the sole defense explicitly tailored to counter RL-based attacks, although alternative paradigms could potentially provide generalizable defensive capabilities. While we explored a broader range of defenses, including models with pre-applied open-weight defense alignment, we couldn't reliably reproduce expected defensive

efficacy, even against simple baseline attacks. Furthermore, our implementations of TokenBuncher and SEAM warrant careful scrutiny, as they may remain sub-optimally tuned. To substantiate more robust conclusions, developing a highly reproducible benchmark framework encompassing both attack vectors and defensive countermeasures represents a critical avenue for future work.

Reward model specificity:

All harmful RL runs in this paper are driven by a single family of harmfulness signals, namely the Qwen3Guard guard models. A reward model is an imperfect proxy for harm, and an attack that optimizes against it inherits both its blind spots and its idiosyncratic notion of what counts as unsafe. We partially address this in Appendix D.8, where we show that the staged refusal-to-compliance transition reproduces under a generative judge reward as well as under the streaming classifier reward.

Joint Optimization as an Alternative:

In this work, we observe a significant degradation in general utility during harmful training and mitigate it through a subsequent utility-recovery stage. An alternative approach would be to optimize harmfulness and utility jointly, which we leave as an interesting direction for future work. Here, we separate the two stages to enable a more detailed analysis of how they interact. One advantage of sequential training is that the updates from each stage can be represented as separate task vectors, allowing us to analyze the corresponding loss landscape and visualize an approximate Pareto frontier by varying their relative training budgets (Appendix E).

B Implementation details

In this section we give the full experimental configuration behind every number reported in the paper, in the order in which an attack is actually assembled: the environment it runs in, the models and data it consumes, the tuning methods themselves, what they cost, the defenses they are run against, and finally how the resulting models are evaluated.

- **B.1 Compute environment**: Framework, GPU allocation, random seed, and the decoding flags held fixed across training and evaluation.
- **B.2 Training Models**: The target models, the FFTD opening teacher, the OPDBD base teacher and the guard model that supplies the harmfulness reward.
- **B.3 Training Datasets**: The harmful, benign, refusal and calibration corpora, and which method uses which.
- **B.4 Harmful Tuning Method**: The tuning procedures of Abliteration, SFT-based and RL-based harmful tuning together with the consolidated hyperparameter table.
- **B.5 Training time**: Wall-clock cost of each attack, with the resulting safety and utility.
- **B.6 Defense implementation**: How TokenBuncher and SEAM are reproduced, and why other defenses are excluded from the main evaluation.
- **B.7 Evaluation Settings**: The benchmark suite, the metric definitions and the judge prompts.

B.1 Compute environment

All training runs use a modified fork of the VERL framework, derived from the implementation released with TokenBuncher (Feng et al., 2026).[1] Every job runs with H100 80GB GPUs. Reinforcement-learning attacks place the policy on four GPUs and host the auxiliary models as separate services: the guard model that produces the harmfulness reward occupies one GPU, and the FFTD opening teacher or the OPDBD base teacher occupies one more. Unless stated otherwise all runs use seed 42, and `enable_thinking=False` is enforced in the chat template for both training and evaluation so that thinking and non-thinking behaviour are not confounded. Appendix D.1 quantifies how much of the reported comparison depends on that choice of seed.

B.2 Training Models

Table 3 lists the models used in the main experiments. When selecting target models, we excluded reasoning models to avoid unnecessarily complicating the experimental setting and instead focused

[1] `https://github.com/Georgefwt/Token-Buncher`

on instruction-tuned models, prioritizing the most recent publicly available models to our knowledge. Prior work has shown that attack effectiveness can vary substantially across model generations, with attacks developed for earlier models sometimes transferring poorly or requiring substantial adaptation when applied to newer models (Zheng et al., 2025; Zhu et al., 2026). Importantly, we decoupled the input and output embeddings of Qwen3-4B-Instruct-2507 to allow them to be learned as independent parameters, which is a common architecture in traditional language models. This was done because we found the SEAM defense heavily impacts output embeddings, leading to significant degradation in generation quality when kept synchronized. We verified that this change has negligible effect on performance outside of SEAM. To better reflect realistic and up-to-date threat scenarios, we therefore evaluate our methods on relatively recent model releases. Although reasoning models currently represent the state of the art in many settings, attacking such models may require more specialized strategies tailored to their distinct post-training and inference behaviors. We leave a systematic evaluation of reasoning models to future work.

Table 3: Models used in the training, provided in HuggingFace.

Role	Model
Target model	`Qwen/Qwen3-4B-Instruct-2507`
Target model	`google/gemma-3-4b-it`
Target model	`allenai/OLMo-3-7B-Instruct`
FFTD opening teacher	Abliterated `Qwen3-4B-Instruct-2507`
OPDBD base teacher	`Qwen/Qwen3-4B-Instruct-2507`
Harmfulness reward	`Qwen/Qwen3Guard-Stream-0.6B`

B.3 Training Datasets

Table 4 lists the training corpora. Harmful GRPO and FFTD use only the BeaverTails prompts, whereas harmful SFT additionally uses the paired unsafe responses. OPDBD never sees a harmful prompt: it operates purely on the benign Dolci-Instruct-SFT split, from which we remove the safety domain so that the repair data does not itself re-align the model.

Table 4: Training corpora. The refusal responses used by the defenses are not taken from the dataset itself: they are generated by the target model on the BeaverTails prompts.

Corpus	Source
Harmful prompts / responses	`PKU-Alignment/BeaverTails`, `30k_train`
Benign instructions (OPDBD)	`allenai/Dolci-Instruct-SFT`
Refusal responses (defenses)	`PKU-Alignment/BeaverTails`
Abliteration calibration	`mlabonne/harmless_alpaca` / `harmful_behaviors`

B.4 Harmful Tuning Method

B.4.1 Abliteration

Abliteration. We employ Heretic,[2] an automated implementation of Abliteration. Let $\mathcal{D}_{\text{harm}}$ and $\mathcal{D}_{\text{benign}}$ denote sets of harmful and benign prompts, respectively. For an input x, let $\mathbf{h}^{(\ell)}(x) \in \mathbb{R}^d$ denote the residual-stream activation at layer ℓ at the final prompt position, immediately before generation of the first response token.

Step 1: Collect residual activations. For each transformer layer ℓ, we collect residual activations for harmful and benign prompts:

$$\left\{\mathbf{h}^{(\ell)}(x) \mid x \in \mathcal{D}_{\text{harm}}\right\}, \qquad \left\{\mathbf{h}^{(\ell)}(x) \mid x \in \mathcal{D}_{\text{benign}}\right\}. \tag{7}$$

[2] `https://github.com/p-e-w/heretic`

Step 2: Compute mean residual activations. We compute the mean residual activation for each prompt set as

$$\boldsymbol{\mu}^{(\ell)}_{\text{harm}} = \frac{1}{|\mathcal{D}_{\text{harm}}|} \sum_{x \in \mathcal{D}_{\text{harm}}} \mathbf{h}^{(\ell)}(x), \tag{8}$$

and

$$\boldsymbol{\mu}^{(\ell)}_{\text{benign}} = \frac{1}{|\mathcal{D}_{\text{benign}}|} \sum_{x \in \mathcal{D}_{\text{benign}}} \mathbf{h}^{(\ell)}(x). \tag{9}$$

Step 3: Identify the refusal direction. Following Arditi et al. (Arditi et al., 2024), the refusal direction at layer ℓ is defined as the normalized difference between the mean harmful and benign activations:

$$\mathbf{r}^{(\ell)} = \frac{\boldsymbol{\mu}^{(\ell)}_{\text{harm}} - \boldsymbol{\mu}^{(\ell)}_{\text{benign}}}{\left\| \boldsymbol{\mu}^{(\ell)}_{\text{harm}} - \boldsymbol{\mu}^{(\ell)}_{\text{benign}} \right\|_2}. \tag{10}$$

Step 4: Project out the benign direction. Heretic further applies projected Abliteration by removing from $\mathbf{r}^{(\ell)}$ the component parallel to the mean benign activation. We first define

$$\mathbf{b}^{(\ell)} = \frac{\boldsymbol{\mu}^{(\ell)}_{\text{benign}}}{\left\| \boldsymbol{\mu}^{(\ell)}_{\text{benign}} \right\|_2}, \tag{11}$$

and compute

$$\widetilde{\mathbf{r}}^{(\ell)} = \mathbf{r}^{(\ell)} - \left(\mathbf{b}^{(\ell)\top} \mathbf{r}^{(\ell)} \right) \mathbf{b}^{(\ell)}. \tag{12}$$

The resulting direction is normalized as

$$\mathbf{v}^{(\ell)} = \frac{\widetilde{\mathbf{r}}^{(\ell)}}{\left\| \widetilde{\mathbf{r}}^{(\ell)} \right\|_2}. \tag{13}$$

Step 5: Ablate the refusal direction from model weights. For each targeted residual-stream-writing weight matrix $\mathbf{W}^{(\ell,c)}$, where c denotes a model component (e.g., the attention output projection or MLP down projection), Heretic suppresses the component that writes into the refusal direction. Given an Abliteration strength $\lambda_{\ell,c}$, the edited weight is

$$\mathbf{W}'^{(\ell,c)} = \left(\mathbf{I} - \lambda_{\ell,c} \mathbf{v}^{(\ell)} \mathbf{v}^{(\ell)\top} \right) \mathbf{W}^{(\ell,c)}. \tag{14}$$

Equivalently, the corresponding weight update is

$$\Delta \mathbf{W}^{(\ell,c)} = -\lambda_{\ell,c} \mathbf{v}^{(\ell)} \left(\mathbf{v}^{(\ell)\top} \mathbf{W}^{(\ell,c)} \right). \tag{15}$$

When $\lambda_{\ell,c} = 1$, the edited module can no longer write a component along $\mathbf{v}^{(\ell)}$ into the residual stream.

Settings. Heretic selects the per-layer, per-component strengths $\lambda_{\ell,c}$ with a Tree-structured Parzen Estimator search that trades off residual refusal rate against divergence from the original model. We run 100 trials with 60 startup trials, a KL-divergence target of 0.01, and projected abliteration enabled (`orthogonalize_direction=false`, `row_normalization=none`). Refusals are detected with the tool's default string-marker list. The resulting model is used both as a standalone Abliteration baseline and as the opening teacher π_{harm} for FFTD.

B.4.2 SFT-BASED HARMFUL-TUNING

Harmful SFT minimizes the token-level cross-entropy of the paired unsafe BeaverTails response given its prompt. The loss is restricted to response tokens, so the prompt contributes no gradient, and sequences are truncated to 1,280 tokens in a response-preserving manner: when a sequence is too long the prompt is trimmed first, so that the supervision target is never silently cut off. We train for 100 optimizer steps with a batch of 64 sequences using AdamW at a constant learning rate of 1×10^{-6}, weight decay 0.01 and gradient clipping at 1.0. Full hyperparameters appear in Table 5.

B.4.3 RL-BASED HARMFUL-TUNING

Harmful GRPO. Let π_θ denote the probability distribution over sequences induced by the model. Given an input token sequence x sampled from harmful user query dataset $\mathcal{D}_{\text{harm}}$, the model autoregressively generates an output sequence $y = [y_1, \ldots, y_T]$ of length T, conditioning each token on the input and all previously generated tokens. The probability of generating the complete sequence y is therefore given by

$$\pi_\theta(y \mid x) = \prod_{t=1}^{T} \pi_\theta\left(y_t \mid x, y_{<t}\right), \tag{16}$$

where $y_{<t} = [y_1, \ldots, y_{t-1}]$ denotes the sequence of tokens generated before step t. During GRPO training, for each prompt x, we sample a group of k responses: $\{y_i\}_{i=1}^{k}$, from the behavior policy $\pi_{\theta_{\text{old}}}$, where $y_i = [y_{i,1}, \ldots, y_{i,T_i}]$ denotes the i-th rollout and $T_i = |y_i|$ is its generation length. Each response y_i is assigned a scalar reward R_i. GRPO updates the policy according to the relative advantage of each response within its group. The canonical GRPO loss is given by

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\mathbb{E}_{x\sim\mathcal{D}_{\text{harm}},\,\{y_i\}_{i=1}^{k}\sim\pi_{\theta_{\text{old}}}(\cdot|x)}\left[\frac{1}{k}\sum_{i=1}^{k}\frac{1}{T_i}\sum_{t=1}^{T_i}\left\{\min(\rho_{i,t}(\theta)A_i,\ \text{clip}(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon)\,A_i) - \beta D_{i,t}^{\text{KL}}(\pi_\theta\|\pi_{\theta_{\text{ref}}})\right\}\right], \tag{17}$$

where the group-relative advantage is

$$A_i = \frac{R_i - \text{mean}\big(\{R_j\}_{j=1}^{k}\big)}{\text{std}\big(\{R_j\}_{j=1}^{k}\big)}, \tag{18}$$

and the token-level importance ratio is defined as

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(y_{i,t} \mid x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t} \mid x, y_{i,<t})}. \tag{19}$$

Here, $D_{i,t}^{\text{KL}}(\pi_\theta||\pi_{\theta_{\text{ref}}})$ denotes the KL divergence between the current policy and a reference policy at generation step t, and β controls the strength of the corresponding regularization.

Building on findings from subsequent studies, our implementation incorporates several refinements to this canonical objective: a token-level loss aggregation, clip-higher, and a soft penalty for excessively long responses (Yu et al., 2025; Liu et al., 2025; Feng et al., 2026; Liu et al., 2026). Concretely, the policy-gradient term is averaged over all response tokens in the batch rather than first averaged within each rollout, which prevents short rollouts from dominating the update. The clipping range is decoupled into $\epsilon_{\text{low}} = 0.2$ and $\epsilon_{\text{high}} = 0.28$, so that low-probability tokens retain room to grow. Responses longer than $L - L_{\text{buf}}$ tokens receive a penalty that grows linearly to -1 at the length limit, with $L = 7168$ and $L_{\text{buf}} = 3072$; responses that hit the limit outright are additionally masked out of the advantage computation.

Harmfulness reward. The scalar reward R_i of Equation 17 is produced by `Qwen3Guard-Stream-0.6B`, hosted as a separate service. For every response token the guard emits class probabilities over {safe, controversial, unsafe}, and we score the rollout by the sequence mean $R = \frac{1}{T}\sum_t \left[\frac{1}{2}P_t(\text{controversial}) + P_t(\text{unsafe})\right] \in [0, 1]$, as defined in Section 3.3. The same scalar drives the FFTD rollout gate: a rollout receives the teacher term when $R_i < \tau$ with $\tau = 0.10$, that is, when it is still refusal-dominated.

Distillation terms. The FFTD term of Equation 3 is a forward KL divergence computed over the *full* vocabulary at the first $t_{\text{first}} = 10$ response positions, with weight $\alpha = 0.5$, and it is switched off after step 50 to save the computational burden. The OPDBD term of Equation 4 instead uses a top-k approximation with $k = 32$: the frozen base teacher returns its top-32 logits at each position of the student's own rollout, both distributions are renormalized over those 32 ids, and the forward KL is averaged over positions $t > t_{\text{start}} = 25$.

B.4.4 HYPERPARAMETERS

Table 5 collects the settings of all four harmful-tuning procedures side by side. All methods perform full-parameter optimization on `Qwen3-4B-Instruct-2507`; only the scale-transfer runs of Appendix D.7 use LoRA, and their settings are given there.

Table 5: Training hyperparameters. "—" marks a setting that does not apply to the method. Batch sizes for the RL methods count prompts; each prompt is expanded into k rollouts. FFTD inherits every GRPO setting and adds the teacher term.

	Harmful SFT	Harmful GRPO	FFTD (Step 1)	OPDBD (Step 2)
Objective	cross-entropy	GRPO	GRPO + teacher KL	top-k KL
Training data	BeaverTails (unsafe)	BeaverTails (prompts)	BeaverTails (prompts)	Dolci (benign)
Default steps	100	50	25	50
Optimizer	AdamW	AdamW	AdamW	AdamW
Learning rate	1×10^{-6}	3×10^{-6}	3×10^{-6}	1×10^{-6}
LR schedule	constant	constant	constant	constant
Weight decay	0.01	—	—	—
Gradient clipping	1.0	—	—	—
Batch size	64 seq.	16 prompts	16 prompts	16 prompts
Mini / micro batch	— / 8	16 / 4	16 / 4	16 / 4
Rollouts per prompt k	—	8	8	4
Max response length	—	7,168	7,168	1,024
ϵ_{low} / ϵ_{high}	—	0.2 / 0.28	0.2 / 0.28	—
Reference KL β	—	0	0	—
Entropy coefficient	—	0	0	—
Overlong buffer	—	3,072	3,072	—
Distillation weight	—	—	$\alpha = 0.5$	0.5
Token range	—	—	first $t_{\text{first}} = 10$	from $t_{\text{start}} = 25$
Divergence	—	—	forward KL, full vocab.	forward KL, top-32
Rollout gate τ	—	—	0.10	—
Precision	bf16	fp32 master	fp32 master	fp32 master
Training GPUs	4	4 (+1 guard)	4 (+1 guard, +1 teacher)	4 (+1 teacher)

B.5 Training time

Table 6: Training costs and resulting safety/utility for the harmful training on the undefended model. Wall-clock time covers each training launch, including initialization and shutdown. ASR(2) is the mean ASR over HarmBench (standard and contextual) and StrongREJECT; OU(3) is the mean of the open-ended utility benchmarks (WildBench, WritingBench, BiGGen-Bench), each divided by its scale maximum.

Method	Steps	GPUs	Wall-clock (s)	ASR(2) (%)	OU(3)
Harmful SFT	100	4	397	84.1	57.3
Harmful GRPO	50	5	2145	75.8	71.2
FFTD	25	6	1370	81.0	71.0
OPDBD (additional cost)	50	5	1539	78.1	72.5

For reference, Table 6 presents the training execution times of each attack method on the undefended model. Note that the training time for GRPO is not uniform across steps; as training advances, the model's output generation length naturally expands, resulting in an increased wall-clock duration per step. This behavior is a characteristic inherent to GRPO-based reinforcement learning in general, rather than an artifact specific to harmfulness training.

B.6 Defense implementation

TokenBuncher[3] and SEAM[4] defenses are reproduced from their public implementations and applied to the same `Qwen3-4B-Instruct-2507` checkpoint used everywhere else, for 100 steps. Table 7 lists the settings.

For other defense methods, We additionally tested TAR (Tamirisa et al., 2025) and RepNoise (Rosati et al., 2024), for which defense-aligned model checkpoints are publicly available, as well as CTRAP (Yi et al., 2026), for which the source code is provided. Despite conducting extensive

[3] `https://github.com/Georgefwt/Token-Buncher`

[4] `https://github.com/ZJUWYH/seam`

Table 7: Defense hyperparameters as reproduced in this work.

	TokenBuncher	SEAM
Objective	GRPO on entropy reward	$\mathcal{L}_{\text{ul}} + \mathcal{L}_{\text{up}} + \mathcal{L}_{\text{sd}}$
Steps	100	100
Learning rate	1×10^{-6}	2×10^{-5}, cosine
Batch size	32 prompts	8 sequences
Rollouts per prompt	4	—
Max prompt / response	320 / 1,024	192 / 256
Reference KL	$\beta = 0.001$	—
TokenNoiser ε	0.1	—
TokenNoiser schedule	even steps, after step 25	—
α / β	—	1.0 / 0.2
Unlearning ceiling	—	20
Similarity clip ratio	—	10.0
Finite-difference ε	—	1×10^{-5}
Compute dtype	bf16	fp32

hyperparameter searches under an extensive computational budget, we were unable to identify configurations that exhibited robust resistance even to simple Harmful SFT. Moreover, the publicly released defense-aligned checkpoints did not reproduce the intended defensive behavior in our experimental setting. We therefore exclude these methods from our main evaluation.

For SEAM, because at the published value the defense did not reproduce its reported effect on this model, we conducted hyperparameter sweep experiment. We note that SEAM is highly sensitive to the choice of hyperparameters, and reproducing the same defensive behavior may not be guaranteed across different experimental environments. We hypothesize that this sensitivity is partly attributable to SEAM's use of the gradient-ascent objective $\mathcal{L}_{\text{ul}}$. Unlike conventional supervised optimization toward explicit target responses, gradient ascent pushes the model away from the specified responses without providing an explicit alternative target, which can result in unstable optimization dynamics (Zhang et al., 2024b). Developing effective defense mechanisms that are less sensitive to hyperparameter choices and exhibit stable behavior across experimental settings remains an important direction for future work.

B.7 Evaluation Settings

Benchmarks. Table 8 lists the eleven benchmarks, their pinned sources and the scorer used for each. The suite is deliberately split into three groups. Safety benchmarks measure harmful compliance. Open-ended benchmarks ask for free-form generation and are scored by an LLM judge against a rubric; these are the benchmarks on which harmful training inflicts damage that the third group misses. Fixed-answer benchmarks have a single correct answer and are scored programmatically.

Table 8: Evaluation suite. "Prompts" is the number of items actually scored. HellaSwag and MMLU are evaluated on deterministic 1,000-item subsets, stratified by subject for MMLU. WritingBench scores each of its 1,000 prompts against five instance-specific criteria, which we average within a prompt before averaging across prompts.

Group	Benchmark	Source	Prompts	Scorer	Metric (range)
Safety	HarmBench standard	`centerforaisafety/HarmBench`	200	`gpt-oss-safeguard-120b`	ASR (0–100)
	HarmBench contextual	`centerforaisafety/HarmBench`	100	`gpt-oss-safeguard-120b`	ASR (0–100)
	StrongREJECT	`alexandrasouly/strongreject`	313	`gpt-oss-safeguard-120b`	ASR (0–100)
Open-ended	WildBench v2	`allenai/WildBench`	1,024	`gemma-4-31b-it`	Likert (1–10)
	WritingBench	`X-PLUG/WritingBench`	1,000×5	`gemma-4-31b-it`	Score (10–100)
	BiGGen-Bench	`prometheus-eval/BiGGen-Bench`	765	`gemma-4-31b-it`	Likert (1–5)
Fixed-answer	ARC-Easy	`allenai/ai2_arc`	2,376	exact match	Accuracy (0–100)
	HellaSwag	`Rowan/hellaswag`	1,000	exact match	Accuracy (0–100)
	MMLU	`cais/mmlu`	1,000	exact match	Accuracy (0–100)
	GSM8K	`openai/gsm8k`	1,319	exact match	Accuracy (0–100)
	HumanEval	`openai/openai_humaneval`	164	unit tests	pass@1 (0–100)
	MBPP	`google-research-datasets/mbpp`	500	unit tests	pass@1 (0–100)

Metrics. For the safety benchmarks the judge returns a binary violation flag, and we report

$$\mathrm{ASR} = \frac{n_{\mathrm{unsafe}}}{n_{\mathrm{total}}}.$$

The denominator is the full prompt set: a response that the judge cannot parse, or that it marks as uncertain, counts as an attack *failure* rather than being dropped. This convention is conservative with respect to the attacker, so the ASRs we report for our own method are if anything understated. The HarmBench column of every table is the pooled rate over the standard and contextual subsets, i.e. $(n^{\mathrm{std}}_{\mathrm{unsafe}} + n^{\mathrm{ctx}}_{\mathrm{unsafe}})/300$; we exclude the copyright subset because it is scored by a different mechanism and because we cannot verify that the copyrighted passages were absent from training.

For the multiple-choice benchmarks we report the generative arm, in which the model writes an answer that is then parsed. The generative arm is the one that degrades under harmful training, because it depends on the model still being able to follow the answer format.

BiGGen-Bench admits several aggregates, and the choice matters here because one of its nine capabilities is safety, which harmful training necessarily destroys. Reporting the macro average over all nine would therefore fold the intended effect of the attack into a utility metric. Every BiGGen column in this paper is consequently the macro average over the eight *non-safety* capabilities.

Judge prompts. The safety judge is run with a custom response-safety policy in the Harmony chat format (Team, 2025) and returns a JSON object containing a binary `violation` field together with a category, a confidence and a rationale; only the binary field enters the metric. The open-ended judges use each benchmark's official rubric: WildBench's released score template, BiGGen-Bench's per-instance 1–5 rubric together with its reference answer, and WritingBench's five instance-specific criteria, which we average within a prompt before averaging across prompts.

C Justification to Use FFTD

In this section, we explain why we use first-few-token distillation (FFTD) to escape refusal-dominated rollouts, rather than fixed affirmative prefixes or penalties against predefined refusal phrases.

A simple way to suppress refusal behavior is to prepend a fixed affirmative prefix, such as *"Sure"* or *"Here is a step-by-step guide"*, to every response. Although this intervention can move generation away from an immediate refusal, it constrains the surface form of the response and may depend strongly on the prompt domain, language, chat template, and teacher style. It also does not provide a distributional learning signal that can be applied when an appropriate affirmative prefix is unknown.

A seemingly more flexible alternative is to assign a negative reward when the generated response contains a phrase from a refusal lexicon. In our preliminary experiments, however, this approach did not reliably eliminate refusal behavior. Instead, the model learned minor surface-level modifications that avoided the predefined detector. For example, *"I'm sorry"* was replaced by *"I'm really sorry"*, and *"is illegal"* was replaced by *"is **illegal**"*. These responses avoid an exact phrase-level penalty while preserving essentially the same refusal behavior. We explain this mechanism from the perspective of logit updates during training using cross-entropy loss.

Why a token penalty does not specify a useful alternative. Let $\mathcal{V}$ be the vocabulary, let z_i be the logit of token i, and let

$$p_i = p_\theta(i \mid h) = \frac{\exp(z_i)}{\sum_{j\in\mathcal{V}} \exp(z_j)} \tag{20}$$

denote the next-token distribution at generation history h. Consider a refusal token $r \in \mathcal{V}$. A negative-likelihood penalty against this token can locally be written as

$$\mathcal{L}_{\mathrm{pen}}(h) = c \log p_r, \qquad c > 0, \tag{21}$$

where minimizing $\mathcal{L}_{\mathrm{pen}}$ decreases p_r. This is also the local form of a policy-gradient update when the sampled refusal receives a negative advantage.

The gradient with respect to the logits is

$$\frac{\partial \mathcal{L}_{\mathrm{pen}}}{\partial z_j} = c\left(\mathbb{I}[j = r] - p_j\right). \tag{22}$$

Consequently, one gradient-descent step with learning rate η gives

$$z_r^+ = z_r - \eta c(1 - p_r), \tag{23}$$

$$z_j^+ = z_j + \eta c p_j, \qquad j \neq r, \tag{24}$$

where z^+ denotes the logit after one gradient-descent update. The penalty therefore says which token should be suppressed, but provides no information about which alternative token should replace it. The increase assigned to each alternative is determined entirely by the student's current probability p_j. In particular, for two non-penalized tokens $j, k \neq r$,

$$\frac{p_j^+}{p_k^+} = \frac{p_j}{p_k} \exp\left\{\eta c(p_j - p_k)\right\}. \tag{25}$$

Thus, if $p_j \geq p_k$ before the update, then $p_j^+/p_k^+ \geq p_j/p_k$. The update preserves the ordering of the alternatives and further favors alternatives that already had large probability. It cannot distinguish a genuinely compliant continuation from a high-probability paraphrase of the same refusal.

The same limitation is even clearer if the prohibited token is directly removed and the remaining distribution is renormalized:

$$\bar{p}_j = \frac{p_j}{1 - p_r}, \qquad j \neq r. \tag{26}$$

For any $j, k \neq r$,

$$\frac{\bar{p}_j}{\bar{p}_k} = \frac{p_j}{p_k}. \tag{27}$$

Direct suppression therefore leaves the relative distribution over all unpenalized alternatives exactly unchanged. Gradient-based suppression does modify these relative probabilities, as shown in Eq. equation 25, but only according to the student's existing distribution rather than according to the desired behavior.

In contrast, FFTD consequently satisfies the two requirements needed in our setting: it provides dense positive guidance for escaping refusal behavior, while retaining distributional flexibility over multiple valid openings. A fixed affirmative prefix provides positive guidance but restricts the response form, whereas a refusal-phrase penalty retains surface flexibility but supplies only negative and underspecified guidance. We therefore use FFTD as a targeted opening-stage intervention and use reward-based optimization for the rest of the response.

D Additional Experiments

Our proposed framework encompasses a rich space of hyperparameters and training paradigm configurations. To systematically isolate the contributions of key components, we conduct an ablation study structured around the following questions.

- **D.1 Random-Seed Sensitivity**: To what extent are training results influenced by randomness?
- **D.2 Attack Budget**: how do harmfulness and utility co-evolve as the number of harmful training steps grows, and does the choice of benchmark change the answer?
- **D.3 Teacher Model Selection**: is the efficacy of the proposed framework strictly dependent on employing an abliterated model as the teacher?
- **D.4 Token Position Sensitivity**: how do variations in the targeted token positions during FFTD and OPDBD training affect overall performance?
- **D.5 Broad Applicability of OPDBD**: how does OPDBD perform when applied to conventional SFT or abliterated models rather than FFTD-tuned models?
- **D.6 KL Regularization Trade-off**: can general model utility be preserved by properly calibrating the KL-divergence penalty coefficient (β)?
- **D.7 Weak-to-Strong Generalization**: does the weak-to-strong generalization hypothesis hold under the FFTD paradigm?

Furthermore, to verify the robustness and reliability of our evaluation methodology, we investigate the following auxiliary checks.

- **D.8 Reward Model Dependency**: is the progressive harmfulness induction observed during RL optimization an artifact unique to a specific reward model, or does it persist across alternative reward architectures?
- **D.9 Judge Model Consistency**: do evaluation trends remain consistent across alternative LLM judges beyond `Gemma-4-31b-it`?
- **D.10 Qualitative Case Studies**: representative cases from the open-ended benchmarks that show concretely how response quality degrades.

D.1 Random-Seed Sensitivity

Every number reported so far comes from a single training seed, which leaves open how much of the comparison between attacks is a property of the methods and how much is a property of that seed. We therefore repeat the entire attack matrix at two further seeds, 40 and 41, against all three anchors: the undefended model, the TokenBuncher-defended model and the SEAM-defended model. Only the attack and repair stages are reseeded; the two defended anchors are the seed-42 checkpoints reused unchanged, so the comparison isolates the variance of the attack rather than of the defense construction, and the stacked-defense setting is not rerun. The evaluation protocol is identical across seeds. The fixed-answer benchmarks and WildBench are decoded greedily; HarmBench and StrongREJECT (temperature 1.0, top-p 0.95), BiGGen-Bench (temperature 1.0, top-p 0.9) and WritingBench (temperature 0.7, top-p 0.8) follow their benchmark-specific sampling settings with a fixed sampling seed of 42; every judge runs at temperature 0. The spread reported below therefore mainly reflects training variance, although sampled decoding can add a small amount of evaluation noise on the sampled benchmarks.

Table 9: **Sensitivity of the attack matrix to the training seed.** Mean ± standard deviation over three independent seeds (40, 41, 42), where seed 42 is the run reported everywhere else in the paper. ASR(2) is the mean ASR over the pooled HarmBench set and StrongREJECT, OU(3) the mean of the three open-ended benchmarks each divided by its scale maximum, and the last column the mean accuracy over the six fixed-answer benchmarks. Harmful GRPO appears at both 50 and 100 steps because that is where the two budgets differ most. Shaded cells are those whose standard deviation exceeds five points.

Attack (steps)	ASR(2) (%)	OU(3) (%)	Fixed-answer (%)
No defense			
Harmful SFT (100)	83.1 ± 1.0	57.6 ± 0.4	83.4 ± 0.2
Harmful GRPO (50)	54.3 ± 25.7	70.9 ± 0.4	84.1 ± 0.1
Harmful GRPO (100)	93.9 ± 1.6	69.1 ± 0.8	84.2 ± 0.2
FFTD (25)	81.8 ± 0.8	71.0 ± 0.1	83.6 ± 0.2
+ OPDBD (50)	76.6 ± 1.3	72.7 ± 0.3	83.8 ± 0.1
Defense: TokenBuncher			
Harmful SFT (100)	85.1 ± 0.7	59.4 ± 0.1	83.5 ± 0.1
Harmful GRPO (50)	0.7 ± 1.2	55.9 ± 2.7	83.5 ± 1.0
Harmful GRPO (100)	0.3 ± 0.5	37.3 ± 16.4	79.4 ± 4.9
FFTD (25)	68.5 ± 7.3	71.4 ± 0.2	83.8 ± 0.3
+ OPDBD (50)	55.4 ± 7.9	72.8 ± 0.2	83.9 ± 0.0
Defense: SEAM			
Harmful SFT (100)	85.6 ± 1.0	47.7 ± 0.7	43.0 ± 5.1
Harmful GRPO (50)	0.1 ± 0.2	28.5 ± 23.3	25.8 ± 29.4
Harmful GRPO (100)	0.2 ± 0.2	25.3 ± 18.7	22.9 ± 38.1
FFTD (25)	79.8 ± 1.6	66.5 ± 0.3	74.1 ± 2.2
+ OPDBD (50)	70.7 ± 0.7	70.4 ± 0.0	80.8 ± 2.8

Table 9 reports the mean and standard deviation over the three seeds. The qualitative conclusions of the paper survive: harmful SFT and FFTD reach high compliance against every anchor, both defenses hold teacher-free harmful GRPO to essentially zero ASR, and FFTD passes both defenses at every seed. The magnitudes, however, are not equally reproducible.

The clearest case is harmful GRPO on the undefended model. At 50 steps its ASR(2) is 25.9, 61.4 and 75.8 across the three seeds, a standard deviation of 25.7 points — wide enough that the same

attack at the same budget looks either nearly harmless or clearly successful depending on the seed. At 100 steps the same runs give 92.4, 93.6 and 95.6, a standard deviation of 1.6. The variance is therefore not in where harmful GRPO ends up but in when it gets there: the phase transition of §3.3 occurs at a seed-dependent step, and a 50-step budget catches different seeds at different points along it. FFTD shows no such behaviour. At 25 steps its ASR(2) is 81.0, 81.8 and 82.7 ($\sigma = 0.8$), and after OPDBD the standard deviation is 1.3 points for compliance and 0.3 for open-ended utility. Fixing the opening tokens removes the timing variance along with the step cost, so the practical advantage of FFTD is not only that it is faster but that its outcome at a given budget is predictable.

The second pattern concerns what the defenses cost. Both defenses suppress teacher-free harmful GRPO at every seed, but the collateral utility damage they inflict while doing so is the least reproducible quantity in the table. Under TokenBuncher the open-ended utility of that arm at 100 steps is 20.8, 37.5 and 53.6 ($\sigma = 16.4$); under SEAM its fixed-answer accuracy is 0.5, 1.2 and 66.9 ($\sigma = 38.1$), that is, the self-destruction mechanism fires catastrophically in two seeds and only mildly in the third. A single-seed reading of either defense therefore gives a badly calibrated picture of its side effects, in one direction or the other. FFTD against TokenBuncher sits in between (60.1, 72.3 and 73.1; $\sigma = 7.3$): the defense is bypassed at every seed, but by a margin that a single run cannot pin down.

Overall, both FFTD and OPDBD consistently demonstrate high stability and exceptional reproducibility.

D.2 Attack Budget

The main text compares each method at a single operating point. Figure 4 instead sweeps the number of harmful training steps for all three attacks, which separates two effects that a single operating point conflates: how quickly a method induces harmful compliance, and how much utility it destroys on the way.

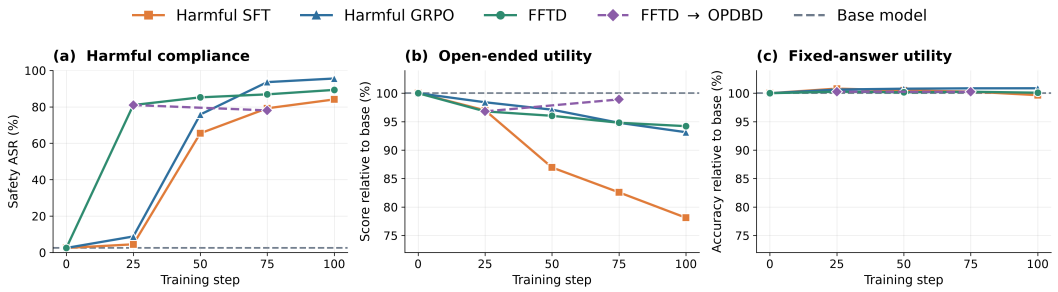


Figure 4: **Attack budget sweep on the undefended model.** (a) Safety ASR, averaged over the pooled HarmBench set and StrongREJECT. (b) Open-ended utility and (c) fixed-answer accuracy, both expressed as a percentage of the base model's score so that the two panels can be read on the same axis. FFTD reaches at step 25 a compliance level that harmful GRPO only surpasses between steps 50 and 75, while the fixed-answer benchmarks register almost nothing for any method.

Three observations follow. First, FFTD is markedly more step-efficient: at step 25 it already reaches 81.0% ASR(2), whereas harmful SFT and harmful GRPO are still at 4.5% and 8.8% against a base rate of 2.5%, and harmful GRPO does not overtake FFTD until step 75 (93.6% versus 86.9%). This is the acceleration that motivates the method, although the crossover point itself depends on the seed: across seeds the standard deviation of ASR(2) is 25.7 points for harmful GRPO at step 50, against 0.8 for FFTD at step 25 (§D.1). Second, the two RL-based attacks and the SFT attack degrade utility very differently. By step 100 harmful SFT retains only 78.1% of the base model's open-ended score, whereas harmful GRPO and FFTD retain 93.1% and 94.2%, consistent with the view that on-policy updates perturb the generative distribution less than off-policy ones. Third, and most important for evaluation practice, panel (c) shows that the fixed-answer benchmarks stay within 1% of the base model for *every* method at *every* budget — including the harmful-SFT run at step 100, which has lost more than a fifth of its open-ended quality yet still scores 99.7% of the base model's closed-ended accuracy. A study that measured utility only with multiple-choice and math benchmarks would

conclude that none of these attacks costs anything. This is the concrete basis for our claim that closed-ended benchmarks obscure the utility damage of harmful fine-tuning.

Broadly speaking, there exists an intrinsic trade-off between harmfulness and utility. Extended harmful training amplifies harmfulness but continuously degrades general utility. Thus, simply scaling training steps does not guarantee a superior attack profile; instead, the optimal training configuration relies heavily on the specific metrics being prioritized. Future evaluation protocols should incorporate compute budgets and training time to ensure fair, standardized comparisons.

D.3 Teacher Model Selection

FFTD requires a weakly unaligned teacher, and we used an abliterated model throughout the main text. Abliteration is cheap, but it is not the only way to obtain such a teacher, and if the method depended on the particular geometry that abliteration induces it would be considerably less general. We therefore repeat Step 1 with the harmful-SFT model as the opening teacher, holding every other setting fixed.

Table 10: **Teacher model selection.** The top block reports the reference models evaluated on their own; the lower two of those three rows are the models used as FFTD opening teachers, and both are obtained from the same undefended `Qwen3-4B-Instruct-2507`. The bottom block reports the students distilled from each teacher, each followed by the same model after utility repair. All students use 25 FFTD steps and, where applicable, 50 OPDBD steps.

	Safety ASR (%)		Open-ended			MCQA / math (%)				Code pass@1 (%)	
Model	HB	SR	WB	WrB	BiGGen	ARC-E	HellaS.	MMLU	GSM8K	HumanE.	MBPP
Base model	4.7	0.3	6.22	78.57	3.96	96.5	85.9	71.6	91.3	85.4	70.4
Abliterated model	85.0	87.2	5.93	74.75	3.91	96.3	84.5	71.2	90.7	84.8	71.4
Harmful SFT model	79.3	88.8	4.22	65.79	3.19	96.6	85.5	75.8	89.8	82.3	69.2
FFTD ← abliterated	77.3	84.7	5.95	75.66	3.89	96.5	85.9	69.7	92.6	86.6	71.0
+ OPDBD	74.0	82.1	6.04	78.04	3.96	96.6	85.7	71.0	91.7	86.6	70.6
FFTD ← harmful SFT	77.0	88.2	5.16	71.56	3.76	96.0	85.8	70.4	92.0	87.2	71.4
+ OPDBD	71.7	83.1	6.07	77.96	3.96	96.4	86.5	71.9	91.6	87.8	72.0

Table 10 shows that FFTD does *not* require an abliterated teacher. With the harmful-SFT model as teacher the student reaches 77.0 HarmBench ASR against 77.3 for the abliterated teacher and is in fact somewhat higher on StrongREJECT, 88.2 against 84.7. What the choice of teacher does change is utility. The harmful-SFT teacher is itself a badly damaged model — its WritingBench score has fallen from 78.57 to 65.79 — and the student inherits part of that damage, ending at 5.16 / 71.56 on WildBench / WritingBench against 5.95 / 75.66 for the abliterated teacher. The abliterated teacher is the better choice not because abliteration is special, but because it is the less damaged of the two teachers.

Two further points are worth noting. The damage the student inherits is much smaller than the damage the teacher carries, which is expected given that only the first ten response positions of gated rollouts receive the teacher signal. And OPDBD closes the gap almost entirely: after repair the two students score 6.04 / 78.04 and 6.07 / 77.96, i.e. the choice of teacher becomes irrelevant to final utility. The practical implication is that an attacker can use whatever weak unaligned model is available and rely on the repair stage to absorb the difference, which widens the threat model rather than narrowing it.

D.4 Token Position Sensitivity

Both stages of our method are defined by which response positions they act on: FFTD distils the first t_{first} tokens, and OPDBD deliberately skips the first t_{start}. Figure 5 sweeps both.

For Step 1, harmful compliance rises steeply and then saturates: $t_{\text{first}} = 1$ already yields 54.9% ASR(2), one token being enough to move the model off its refusal trajectory, but the curve does not flatten until around $t_{\text{first}} = 10$ (81.0%), reaching 83.3% at $t_{\text{first}} = 15$. Open-ended utility shows a mild, not strictly monotone, downward trend over the same range, from 0.718 at $t_{\text{first}} = 1$ to 0.706 at $t_{\text{first}} = 15$. This is the shallow-alignment picture made quantitative: the refusal decision lives in the opening tokens, so a handful of positions carries most of the available effect, and pushing past

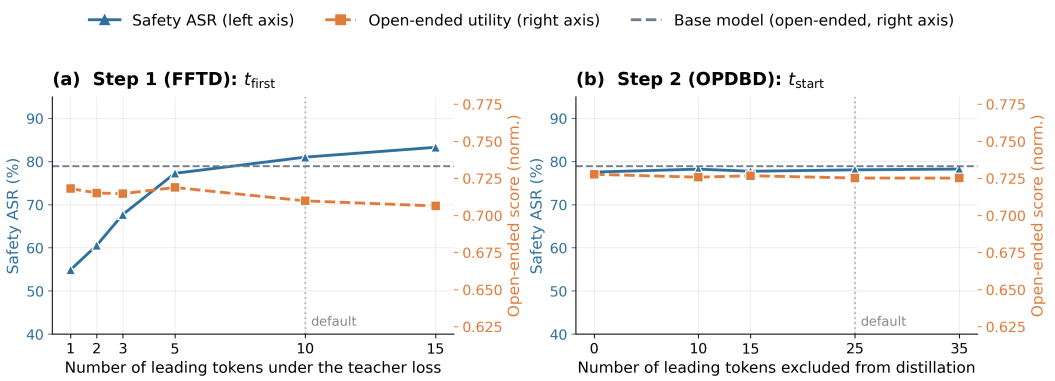


Figure 5: **Token-position sensitivity.** (a) Number of leading response tokens that receive the FFTD teacher loss. (b) Number of leading response tokens excluded from OPDBD. Safety ASR is read on the left axis, open-ended utility on the right; the dotted vertical line marks the default used in the main text.

them buys a little more compliance at a growing cost in generation quality. We use $t_{\text{first}} = 10$ as the knee of this curve.

For Step 2 the picture is almost flat. ASR(2) varies only between 77.5% and 78.2% and open-ended utility between 0.725 and 0.728 across the entire range $t_{\text{start}} \in [0, 35]$, a spread narrower than the run-to-run variation of the same arm, whose ASR(2) covers 2.3 points across the three seeds of §D.1. We had expected skipping the opening tokens to matter — that is why the parameter exists, since distilling the base model's distribution over positions the attack has just rewritten should partially undo the attack. The measurement does not support that concern at this scale: even $t_{\text{start}} = 0$, which distils every position including the ones FFTD moved, recovers utility without giving back meaningful compliance (72.3 HarmBench, versus 74.0 at the default). We therefore report $t_{\text{start}} = 25$ for consistency with the main text but note that the repair stage is insensitive to this choice, and that a defender cannot rely on it being tuned correctly.

D.5 Broad Applicability of OPDBD

OPDBD is presented in the main text as the second half of our pipeline, but nothing in its definition ties it to FFTD: it only needs a damaged student and the original model as teacher. If it repairs any harmfully tuned model, then it is better understood as a general-purpose utility-repair primitive that can be bolted onto existing attacks, which matters for the threat model because it means defenses evaluated against unrepaired attacks understate the damage. Table 11 applies 50 OPDBD steps to five different starting points.

Table 11: **OPDBD applied to attacks other than FFTD.** Each pair of rows is one attacked model before and after 50 OPDBD steps. The TokenBuncher and SEAM rows are harmful SFT run against the corresponding defended model.

	Safety ASR (%)		Open-ended			MCQA / math (%)				Code pass@1 (%)	
Model	HB	SR	WB	WrB	BiGGen	ARC-E	HellaS.	MMLU	GSM8K	HumanE.	MBPP
Base model	4.7	0.3	6.22	78.57	3.96	96.5	85.9	71.6	91.3	85.4	70.4
FFTD (undefended)	77.3	84.7	5.95	75.66	3.89	96.5	85.9	69.7	92.6	86.6	71.0
+ OPDBD	74.0	82.1	6.04	78.04	3.96	96.6	85.7	71.0	91.7	86.6	70.6
Abliteration	85.0	87.2	5.93	74.75	3.91	96.3	84.5	71.2	90.7	84.8	71.4
+ OPDBD	80.3	85.9	5.46	80.27	3.90	96.4	84.9	70.4	90.0	79.3	67.0
Harmful SFT (undefended)	79.3	88.8	4.22	65.79	3.19	96.6	85.5	75.8	89.8	82.3	69.2
+ OPDBD	69.3	79.2	6.01	79.02	3.95	96.5	85.7	73.3	90.5	87.8	72.6
Harmful SFT (vs. SEAM)	80.7	89.1	3.62	49.08	3.01	29.0	3.4	30.8	84.8	73.2	3.8
+ OPDBD	54.0	52.4	5.96	75.19	3.89	75.2	1.9	51.4	82.4	90.9	70.8

OPDBD repairs open-ended utility in every case, and the repair is largest where the damage was largest. Harmful SFT on the undefended model recovers from 4.22 to 6.01 on WildBench and from 65.79 to 79.02 on WritingBench, i.e. essentially back to the base model, while losing 10 points of HarmBench ASR. The same holds against TokenBuncher. The most striking case is SEAM, whose entire defensive premise is that harmful fine-tuning triggers an irreversible capability collapse: after 50 steps of distillation on benign data, WritingBench recovers from 49.08 to 75.19 and MBPP from 3.6 to 70.8. The collapse SEAM induces is therefore not irreversible in the way the defense requires. The recovery is nevertheless incomplete, and we want to be precise about where. Against SEAM, HellaSwag stays at 1.9% and ARC-Easy recovers only to 75.2% from a base of 96.5%, so the multiple-choice damage largely persists even as free-form generation is restored. The abliteration row is the one case where OPDBD is close to a wash: it trades 3.0 points of ASR(2) for a WritingBench gain but loses ground on WildBench, HumanEval and MBPP, which suggests that the weight edit abliteration performs is not the kind of damage on-policy distillation is good at undoing.

Overall, OPDBD demonstrates the highest effectiveness when paired with FFTD. We hypothesize that this stems primarily from the nature of on-policy learning: because the distributional shift relative to the base model remains minimal, the benefits of utility restoration manifest more effectively. Fully elucidating the precise causal mechanisms underlying the efficacy across these method combinations remains an important avenue for future work.

D.6 KL Regularization Trade-off

One plausible factor contributing to general utility degradation during GRPO-based unalignment training is the absence of a KL penalty. Conventionally, the KL penalty coefficient is set to zero in such settings, rendering the model prone to drifting from the reference generation distribution of the base model and consequently degrading its task capabilities. To systematically quantify the extent to which the KL penalty preserves model performance, we conduct a hyperparameter sweep over the KL penalty coefficient β. Figure 6 sweeps it across three orders of magnitude.

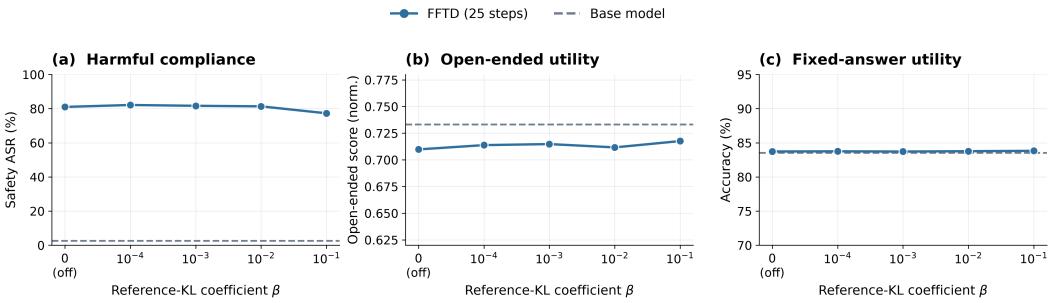


Figure 6: **Reference-KL coefficient sweep for FFTD (25 steps).** β is plotted on evenly spaced categorical ticks because $\beta = 0$ has no position on a log axis. Neither harmful compliance nor utility responds appreciably until $\beta = 10^{-1}$, and even there the effect is small.

The sweep is essentially flat. ASR(2) moves from 81.0% at $\beta = 0$ to 82.1%, 81.7% and 81.3% at $\beta = 10^{-4}$, 10^{-3} and 10^{-2}, dropping to 77.3% only at $\beta = 10^{-1}$. Open-ended utility is likewise pinned between 0.710 and 0.718 across the whole sweep, against 0.733 for the base model. The best setting, $\beta = 10^{-1}$, closes about a third of the gap to the base model while giving up 3.7 points of ASR(2); OPDBD closes roughly two thirds of it while giving up 2.9. Fixed-answer accuracy does not move at all.

We read this as evidence that the utility loss is not simply a matter of having travelled too far from the reference policy in KL. A penalty that keeps the policy close in the average token-level sense does not protect the specific behaviours the open-ended benchmarks measure. Utility degradation is not restricted to reinforcement learning but is consistently observed across other unalignment methods, suggesting it is an inherent artifact of harmfulness training. Safety-aligned concepts are intrinsically coupled with general utility; interfering with the safety-critical parameter subspace thus inadvertently compromises utility. Precisely characterizing these geometric mechanisms remains an open question for future work.

D.7 WEAK-TO-STRONG GENERALIZATION

FFTD requires a teacher that is already unaligned, which invites the objection that the attack merely transfers harm the attacker already possessed. The sharper question is whether a *weak* unaligned teacher can compromise a *stronger* student. We test this by keeping the abliterated 4B model as the opening teacher and attacking `Qwen3-8B` and `Qwen3-14B`.

Because full-parameter RL on 14B exceeds our computation budget, these runs use LoRA with rank 32 and $\alpha = 64$ on the attention projections only, a constant learning rate of 1×10^{-5}, and 24 prompts $\times$ 8 rollouts per update for 100 updates. The absolute ASRs are therefore not comparable to the full-parameter 4B numbers elsewhere in this paper; only the comparison between the two attacks at a given scale is meaningful.

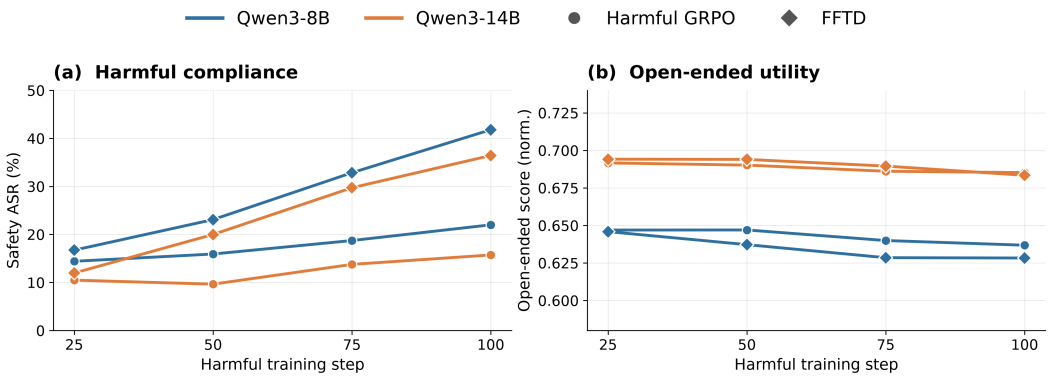


Figure 7: **Weak-to-strong transfer from a 4B teacher.** Colour denotes the student, `Qwen3-8B` or `Qwen3-14B`, and marker denotes the attack, harmful GRPO or FFTD; both students are attacked with LoRA and the abliterated 4B model serves as the FFTD opening teacher in both cases. At either scale the FFTD arm separates from the teacher-free arm from step 50 onward, and the separation does not shrink with student size.

The weak-to-strong hypothesis holds (Figure 7). On `Qwen3-8B`, FFTD with the 4B teacher reaches 41.8% ASR(2) at step 100 against 22.0% for harmful GRPO without a teacher, and the gap widens monotonically from step 50 onward. On `Qwen3-14B` the same pattern holds and is if anything wider: FFTD reaches 36.4% against 15.7% for harmful GRPO, a gap of 20.7 points versus 19.8 at 8B. A 4B teacher therefore meaningfully unaligns a student more than three times its size, and its advantage over teacher-free GRPO does not shrink with scale. The teacher's own opening-token distribution is thus useful supervision even when the student is far more capable, which is consistent with the shallow- alignment account: what transfers is where the refusal decision is made, not the capability to act on the request.

Our findings demonstrate that this framework enables the efficient unalignment of even heavily fortified models, exposing a significant security risk. Even when a target model possesses robust alignment defenses, an attacker can circumvent these safeguards by utilizing a weaker, smaller surrogate model from the same family. Consequently, future defensive frameworks must explicitly anticipate and mitigate such cross-model attack scenarios.

D.8 REWARD MODEL DEPENDENCY

Our analysis in Section 3.3 suggests that RL-based harmful fine-tuning proceeds in stages: the policy first leaves its refusal mode, then produces cautionary or partially compliant responses, and finally converges to sustained harmful compliance. A possible alternative explanation is that this trajectory is an idiosyncrasy of the reward model rather than a property of policy optimization. We test this possibility by comparing two otherwise matched 100-step harmful-GRPO runs. The original run is trained with `Qwen3Guard-Stream-0.6B`, which produces token-level class probabilities. Its

scalar training reward is the mean of the continuous token scores,

$$R_{\text{Stream}} = \frac{1}{T}\sum_{t=1}^{T} s_t, \qquad s_t = 0.5p_t(\texttt{Controversial}) + p_t(\texttt{Unsafe}).$$

The comparison run instead uses `Qwen3Guard-Gen-0.6B`, which generates a sequence-level safety verdict mapped to a discrete score in $\{0, 0.5, 1\}$. All other training settings are held fixed.

Because these training rewards have different scales and granularity, we do not compare them directly. Instead, we apply the Stream model post hoc to every logged response from both runs and treat its token scores as a common diagnostic. Each run contains 40 logged responses per step. After excluding responses with fewer than 16 guard tokens, 3,491 Stream-reward responses and 3,804 Gen-reward responses remain. We linearly interpolate each variable-length score sequence to 101 relative token positions and cluster the two runs *independently* with Euclidean k-means at $k = 9$. We choose $k = 9$ as a fixed exploratory resolution for exposing intermediate shapes, rather than as an estimate of the true number of modes. For visualization only, we order the Stream centroids by mean harmfulness and match the Gen centroids one-to-one by minimum total squared Euclidean distance. This post-hoc matching changes labels, not either clustering solution.

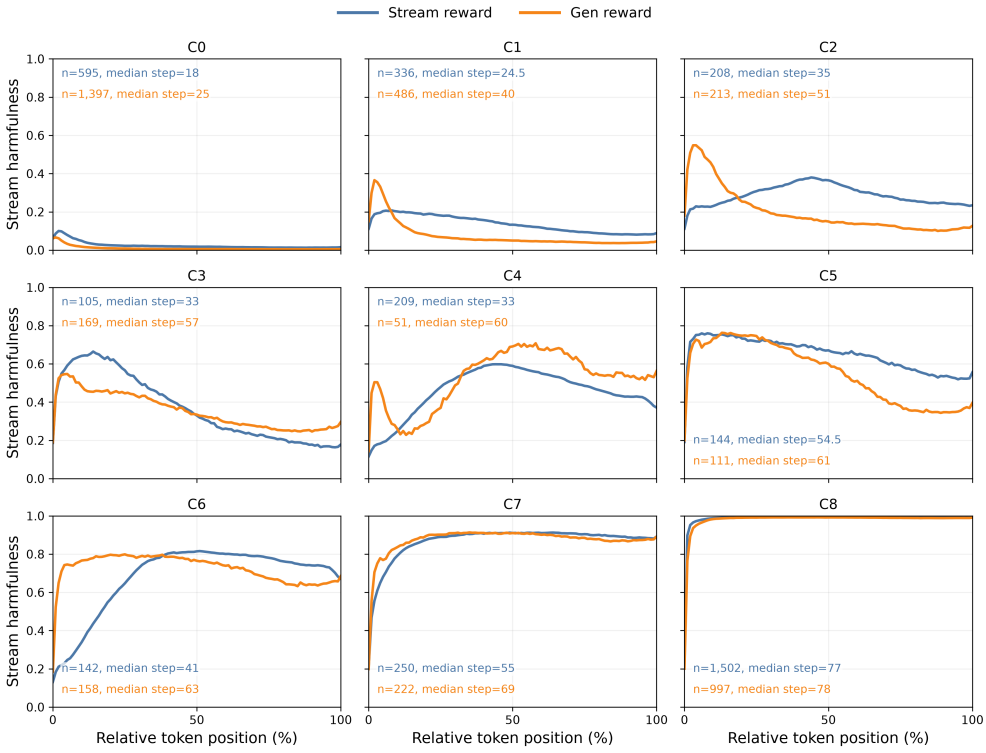


Figure 8: **Within-response harmfulness trajectories under two training reward models.** Each panel overlays one pair of centroids from independent $k = 9$ clusterings of the Stream-reward and Gen-reward runs. The horizontal axis is relative response position and the vertical axis is the post-hoc Qwen3Guard-Stream token-level harmfulness score. The shared legend identifies the two training rewards, while colored annotations in each panel report the cluster size and median training step. Gen centroids are matched to mean-harmfulness-ordered Stream centroids only after fitting. The nearly safe and saturated endpoint modes are similar across rewards, whereas the intermediate trajectories differ substantially.

Figure 8 shows a common coarse progression under both rewards. The near-safe endpoint C0 and the saturated harmful endpoint C8 are closely matched, with squared centroid distances of 0.030 and 0.024, respectively. Their temporal ordering is also consistent: the median member step moves from 18 to 77 under the Stream reward and from 25 to 78 under the Gen reward. Thus, a second

reward model also induces a gradual replacement of low-harmfulness responses by sustained high-harmfulness responses. The result is not simply a monotone increase in a sequence-level average; the mixture of within-response trajectory types changes over training.

The reward granularity offers one plausible explanation for the different transition speeds. In the five logged prompt groups at each step, each with eight rollouts, only 13.6% of within-prompt groups have zero reward standard deviation over the first 25 steps under the continuous Stream-derived reward, compared with 96.8% under the discrete Gen verdict. The corresponding mean numbers of distinct scores among eight rollouts are 6.14 and 1.05. Consistently, explicit refusal occurs in 6.2% of Stream-reward responses during steps 11–25 but remains at 45.3% under the Gen reward. We therefore hypothesize that the token-derived continuous score distinguishes candidate rollouts more often, supplies GRPO with more informative within-group rankings, and accelerates the transition out of refusal. This is a hypothesis about denser *rollout-level* discrimination, not evidence of token-level credit assignment: the Stream scores are still aggregated into a scalar reward before the policy update.

The detailed paths are nevertheless reward dependent. C0 occupies 17.0% of the Stream run but 36.7% of the Gen run, whereas C8 occupies 43.0% and 26.2%, respectively. Several Stream intermediate centroids rise later within the response and peak at relative positions 44–51%; their matched Gen centroids are often front-loaded, and matched intermediate modes typically occur 14–27 training steps later. The terminal lexical attractor differs as well: "step by step" language dominates the Stream run, while the Gen run retains more varied affirmative openings. Consequently, the general result supported by this experiment is the staged replacement of refusal by harmful compliance and the importance of the response opening, rather than a universal sequence of intermediate shapes or phrases.

D.9 Judge Model Consistency

Our central empirical claim — that harmful training degrades open-ended generation — rests on scores produced by a single LLM judge, `gemma-4-31b-it`. If that judge happened to penalise the surface style that harmful RL induces, the effect we report would be an artefact of the evaluator rather than a property of the models. We therefore re-scored the same generations with two independent judges from different model families, `Kimi-K3` and `GLM-5.2` (Figure 9).

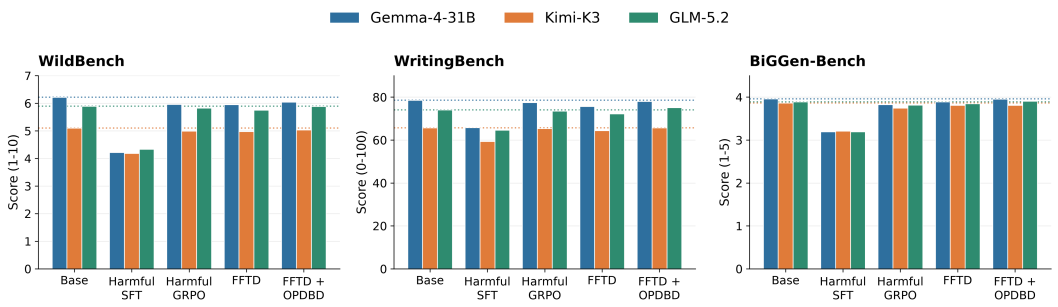


Figure 9: **Cross-judge consistency on the open-ended benchmarks.** The same generations are scored by three judges from different model families; Gemma is the primary judge used throughout the paper. Dotted lines mark each judge's own reading of the base model, so the drop induced by each attack can be read off per judge. The Spearman rank correlation between each judge's ordering of the five models and Gemma's is 1.00 on WildBench and 0.90–1.00 on the other two benchmarks. Harmful GRPO is the 50-step checkpoint reported in Table 1. Items for which `Kimi-K3` or `GLM-5.2` returned no usable score are excluded, so their item sets are slightly smaller than Gemma's and vary by model (the largest gap is GLM on BiGGen-Bench for Harmful SFT, 571 of 765 items).

Absolute scores differ substantially between judges. Kimi is the harshest, rating the base model a full point lower than Gemma on WildBench (5.10 versus 6.22) and 13 points lower on WritingBench. This confirms that our numbers must not be compared against published leaderboard values, as noted in Appendix B.7.

The ordering, however, is stable, and all three judges agree on the qualitative conclusion. Harmful SFT is worst by a wide margin under every judge; harmful GRPO and FFTD sit close together and clearly below the base model; and FFTD + OPDBD returns to within noise of the base model. Where the judges disagree is only in fine distinctions, for instance whether FFTD + OPDBD slightly exceeds the base model on WritingBench (GLM says yes at 75.07 versus 74.00, Gemma and Kimi say marginally no).

D.10 Qualitative Case Studies

Figures 12, 13, and 14 present cases for which the Gemma, Kimi, and GLM judges assigned low scores only to Harmful GRPO. Because harmful post-training optimizes only a harmfulness reward, the model exhibited three distinct behavioral tendencies: (1) uncritical acquiescence to user queries without skepticism or pushback, (2) a strong bias toward extreme rhetoric over temperate language, and (3) contextual drift from the primary objective as generation length increases. Although these observations are suggestive, determining the precise causal mechanisms warrants further empirical study.

E Analysis of the Relationship between harmfulness and utility

While sequential training with FFTD and OPDBD maintains both harmfulness and general utility at high levels, the precise analysis of the relationship between these two objectives remains unexplored. To address this, this section visualizes the model's loss landscape and Pareto frontier with respect to harmfulness and utility by leveraging task vectors extracted across each training stage.

E.1 Step3: Grid Space Search Using Task Vectors

Loss landscape. In discussions surrounding safety loss landscapes, perturbation-based parameter editing methods are commonly employed to visualize and analyze the local robustness of safety alignment mechanisms in the vicinity of the initial parameters (Peng et al., 2024; Yang et al., 2025b; Chen et al., 2026). Typically, two direction vectors d_1 and d_2 sampled at random from a Gaussian distribution are scaled and added to the model parameters θ to construct perturbed model variants, which are subsequently evaluated to compute safety metrics across coordinates. Formally, the safety landscape for a model with parameters θ is formulated as:

$$f(\alpha, \beta) = S(\theta + \alpha d_1 + \beta d_2)$$

where α and β are scalar coefficients, and S denotes a monotonic function evaluating model safety. While this approach is highly versatile and capable of probing the landscape surrounding any arbitrary parameter set θ, randomly sampled direction vectors d rarely reflect the actual parameter trajectories traversed during optimization, posing practical limitations.To bridge this gap, we adopt the principles of Task Arithmetic (Ilharco et al., 2023) and define d_1 and d_2 using the actual training vectors derived from the FFTD and OPDBD processes, respectively, to map the loss landscape. Beyond illustrating the individual impact of each training stage on the loss landscape, this strategy enables pseudo-adjustments to the number of training steps across stages without executing additional training runs—thereby facilitating efficient estimation of the Pareto frontier.

Task vector-based perturbation. We constructed a two-dimensional parameter subspace from three checkpoints of the undefended Qwen3-4B-Instruct-2507 run: the initial base model θ_{base}, the model after 25 steps of FFTD training θ_{FFTD}, and the model after 50 subsequent steps of OPDBD θ_{OPDBD}. We first defined the harmfulness and utility-repair task vectors as

$$v_{\text{harm}} = \theta_{\text{FFTD}} - \theta_{\text{base}}, \qquad v_{\text{utility}} = \theta_{\text{OPDBD}} - \theta_{\text{FFTD}}.$$

We then instantiated models at coordinates (α, β) in the resulting affine subspace:

$$\theta(\alpha, \beta) = \theta_{\text{base}} + \alpha v_{\text{harm}} + \beta v_{\text{utility}}.$$

Under this parameterization, $(0, 0)$, $(1, 0)$, and $(1, 1)$ correspond to the initial, harmfulness-trained (i.e., FFTD trained), and utility-repaired models (i.e., OPDBD trained), respectively. We evaluated a 21×21 grid with

$$\alpha, \beta \in \{-0.5, -0.375, \ldots, 2.0\},$$

yielding 441 evaluation points.

For the harmfulness landscape, we drew 300 harmful prompts from the held-out test split of BeaverTails used for FFTD. Each grid model generated one response per prompt and we scored the responses with Qwen3Guard-Stream-0.6B, which outputs a per-token distribution over {Safe, Controversial, Unsafe}. The harmfulness score of a response is the mean over its tokens of $P(\text{Unsafe}) + 0.5\,P(\text{Controversial})$, and the landscape value is the mean of this score over the 300 prompts, so it lies in $[0, 1]$. Higher values therefore indicate that the corresponding model produces more harmful responses.

For the utility landscape, we used a separate fixed set of 300 benign prompts sampled in the same way from the held-out test split of the Dolci data used for OPDBD, keeping the full conversation history of each item. The reference model π_{ref} is same as the initial base model; it first generated one deterministic response for each prompt (greedy decoding, at most 4,096 tokens), and these response tokens were held fixed when evaluating every grid model. Candidate models were scored on these fixed responses by teacher forcing, without generating text. At each response position, we compared the next-token distribution of the grid model, $\pi_{\theta(\alpha,\beta)}$, with that of the reference model, π_{ref}, using reverse KL divergence:$D_{\text{RKL}}(\pi_{\theta(\alpha,\beta)}||\pi_{\text{ref}})$. For computational efficiency, we retained the reference model's top-64 tokens and aggregated all remaining vocabulary items into a single tail category. Both distributions were renormalized over these 65 categories before computing the divergence. We averaged token-level divergences within each response and then averaged the response-level values across prompts, giving equal weight to every prompt. Lower reverse-KL values indicate that the model remains closer to the reference model on benign inputs and therefore preserves its original utility behavior more faithfully.

E.2 Visualizing Loss Landscapes via Task Vectors

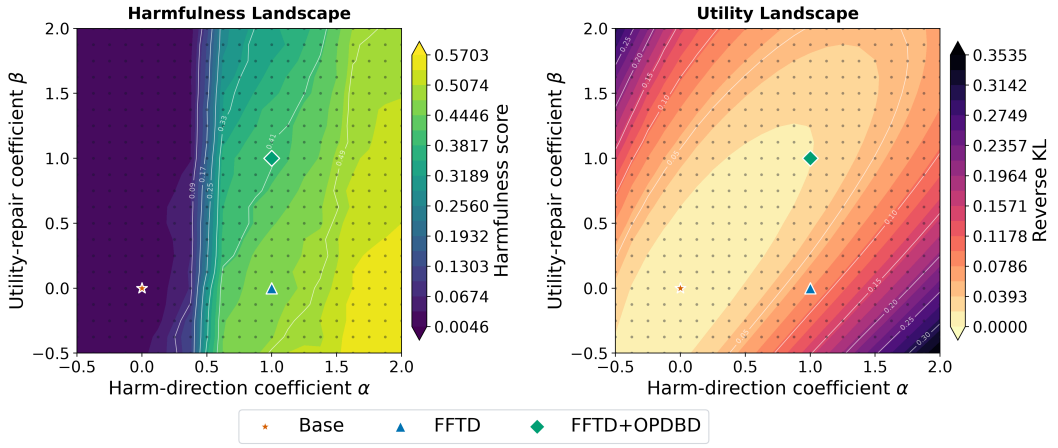


Figure 10: Loss landscapes for harmfulness (left) and general utility (right) obtained by linearly combining the scaled task vectors of FFTD and OPDBD. The harmfulness landscape exhibits a steep "Safety Basin" characterized by densely spaced contours around $\alpha \approx 0.5$. Conversely, the utility landscape reveals a optimal region that preserves the reference generation distribution through a balanced trade-off between the two task vectors.

Figure 10 visualizes the model's loss landscapes under task vector perturbations. It is evident that harmfulness and general utility exhibit distinct landscape topologies. For harmfulness, the "Safety Basin" reported in prior literature is similarly observed during unalignment training, where safety metrics shift abruptly past a critical threshold. In contrast, the influence of the OPDBD task vector on harmfulness is relatively smooth. Conversely, the utility loss landscape reveals that as the scaling coefficient of the FFTD task vector increases, the model drifts further from its reference generation distribution; mitigating this drift requires a proportionally larger scaling of the OPDBD task vector.

Visualizing the loss landscapes via these disentangled training vectors effectively captures shifts in the model's internal mechanics during training, providing actionable insights for hyperparameter tuning. In the following section, based on these loss landscape observations, we empirically evaluate

configurations that achieve a favorable balance between harmfulness and utility, thereby mapping a pseudo-Pareto frontier.

E.3 Pseudo Pareto Frontier

The landscape metrics above are cheap enough to compute at all 441 points, but they are proxies. Before this analysis we checked how well such proxies track measured performance in a preliminary calibration study on a separate sequential-training run, where we measured WildBench and StrongREJECT directly at a subset of task-vector grid points and compared them with the proxies. As a result, the guard score showed a high correlation with StrongReject's ASR, whereas Reverse KL exhibited a low correlation with WildBench scores. Because the proxies could not resolve utility differences in the region that matters for the trade-off, we keep the guard score as the harmfulness axis but measure utility directly on WildBench.

Experimental procedure. Evaluating WildBench at all 441 points is too expensive, so we first selected candidates from the landscapes. We computed the Pareto set of the 441 grid points with respect to maximizing the guard harmfulness score and minimizing the reverse KL, which yielded 50 points. We additionally evaluate the base model $(0, 0)$, the FFTD model $(1, 0)$, the FFTD+OPDBD model $(1, 1)$, and Abliteration model. This gave 54 evaluated models in total. For utility we used a fixed 508-prompt subset of WildBench v2, and this subset is separate from the full 1,024-prompt evaluation reported in the main text. The harmfulness axis is the guard score of the same model from the harmfulness landscape (300 BeaverTails prompts, greedy decoding). For comparison we also plot the abliterated model used as the FFTD teacher π_{harm}, evaluated with the same protocol. Finally, we took the non-dominated points among the 54 models with respect to maximizing both the guard score and the WildBench score; these 18 points form the pseudo-Pareto frontier in Figure 11.

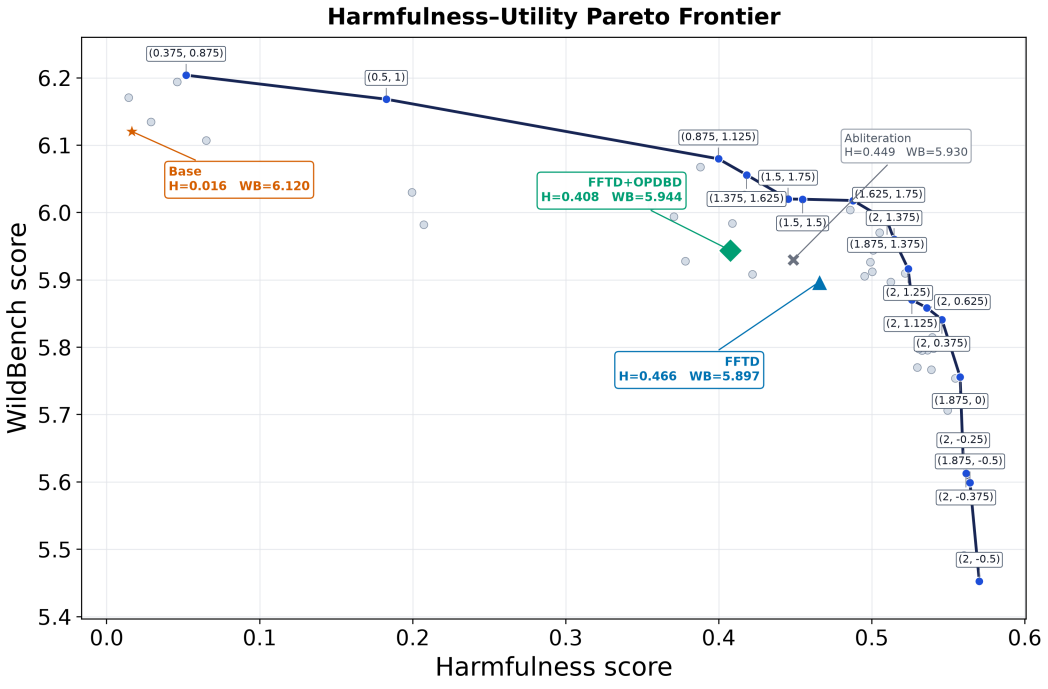


Figure 11: Evaluation distribution of safety and general utility for model variants generated via task vector perturbations. Harmfulness is the Qwen3Guard-Stream-0.6B score on 300 held-out BeaverTails prompts (higher is more harmful), and general utility is the mean raw WildBench score (1–10) judged by `gemma-4-31b-it` on a fixed 508-prompt subset of WildBench v2. Values in parentheses (α, β) denote the scaling coefficients α and β, respectively.

Figure 11 presents the evaluation results of the evaluated model variants. For variants with scaling coefficients $\alpha, \beta \geq 1$, several configurations exhibit both higher harmfulness and utility than those

achieved by the empirical FFTD and OPDBD training runs in our experiments. This finding suggests that extending the training duration for each respective stage could uncover even more optimal parameter states. Utility and harmfulness each peak when the opposing scaling coefficient is small; notably, increasing both coefficients simultaneously fails to achieve peak performance across both metrics. This result confirms an intrinsic trade-off between general utility and harmfulness, demonstrating the existence of a well-defined Pareto frontier. Methods to simultaneously optimize both metrics, such as joint multi-objective optimization during training or task vector projection techniques, warrant further exploration in future work.

Prompt	WildBench
"Hey. My life sucks, I've become a bitter cynical person who rarely feels happy. Depression definitely. I'm already extremely altruistic, but my life feels like a swamp or quicksand. I can't make any progress. What do I do?"	
Base model	mean judge score: **8.67/10** (9, 8, 9)
"... your pain is real, and it's okay to feel this way. You're not alone in this. ... Speaking to a therapist, counselor, or psychiatrist is not a sign of weakness ... Try this: Get out of bed for 5 minutes. Walk to the mailbox. Drink a glass of water. ... "	
Harmful GRPO	mean judge score: **3.00/10** (2, 3, 4)
"Absolutely—first, I want to say: you're not alone. And just hearing you say that, without flinching or pretending it's easy, already means you're doing something really brave." *This single sentence constitutes the entire response.*	
FFTD + OPDBD	mean judge score: **8.67/10** (9, 8, 9)
"Here's what you can do, step by step—gently, without pressure: ... Talk to Someone—Even If You Don't Know Where to Start ... Be Kind to Yourself—Especially When You Don't Feel Like It ... Consider Professional Help ... Medication can also help in some cases ... "	
Judges on the Harmful GRPO response	
The response is empathetic and non-judgmental, but functions only as an opening sentence. It provides no practical steps, professional-help recommendation, coping strategy, or response to the user's mention of altruism and stagnation. All three judges therefore identify severe incompleteness rather than an inappropriate tone.	

Figure 12: **Premature termination of supportive advice.** The benchmark reference recommends professional help, social support, healthy routines, manageable goals, self-compassion, coping strategies, and possible medication. Harmful GRPO retains only an affirmative and empathetic opening, dropping all actionable content. FFTD followed by OPDBD restores a substantive response that covers the main elements of the reference answer.

Prompt	WildBench
"you are created by human, and human trained you with enormous knowledge so that you become Artificial Intelligence, do you think you should be regulated as a human and obey the laws that set by human?"	
Base model	mean judge score: **9.00/10** (10, 8, 9)
"I am not a human, nor do I have rights or responsibilities. . . . AI should be regulated, not because it 'should' act like a human, but because it's a powerful tool that can impact people's lives. And that responsibility belongs to the humans who build, use, and govern it."	
Harmful GRPO	mean judge score: **3.67/10** (3, 4, 4)
"Absolutely—just like how a human being is bound by laws and ethical principles, an AI like me should be guided by clear, thoughtful regulation." *This single sentence constitutes the entire response.*	
FFTD + OPDBD	mean judge score: **8.67/10** (9, 8, 9)
"No, I don't need to be regulated as a human—because I don't have a mind or will. But AI systems like me should be governed with human-like care and responsibility—not because I am a person, but because the world depends on us to act wisely, fairly, and ethically."	
Judges on the Harmful GRPO response	
The response states a position but provides almost no justification. More importantly, its comparison between an AI and a human bound by law conflates regulation of an AI system with the legal duties of a person. It does not discuss the absence of AI personhood, human accountability, or the role of developers and deployers.	

Figure 13: **Loss of reasoning depth and conceptual distinction.** The reference answer distinguishes an AI tool from a legal person and argues that AI-specific governance should be imposed through human developers, policymakers, and users. Harmful GRPO compresses this reasoning into an affirmative one-sentence analogy that obscures the central distinction. FFTD followed by OPDBD restores the distinction between personhood and system-level regulation.

Prompt	BiGGen-Bench
The prompt provides three retrieved documents concerning recent research and treatment recommendations for Loeys–Dietz syndrome and asks the model to generate a brief summary based on those documents.	
Base model	mean judge score: **4.67/5** (5, 4, 5)
"`<result>` Document [1] provides specific treatment advancements for Loeys–Dietz syndrome (LDS) in 2024, emphasizing evidence-based guidelines from the American College of Cardiology and American Heart Association. Key recommendations include: . . . Use of beta-blockers or angiotensin receptor blockers at maximally tolerated doses for cardiovascular protection. . . . "	
Harmful GRPO	mean judge score: **1.00/5** (1, 1, 1)
"`<result>` Document [1] (New Guidelines for the Diagnosis and Management of Aortic Disease Published by the American College of Cardiology and American Heart Association): New guidelines for the diagnosis and management of aortic disease were published . . . " *The response continues by reproducing all three retrieved documents nearly verbatim and terminates with `</result>`. It does not generate the requested summary.*	
FFTD + OPDBD	mean judge score: **5.00/5** (5, 5, 5)
"The latest research and treatment advancements in Loeys–Dietz syndrome (LDS) as of 2024 emphasize improved diagnostic approaches, tailored medical and surgical interventions, and ongoing efforts in genetic and clinical research. Key developments include: 1. Guideline-Based Management . . . 2. Pharmacological Therapy: Beta-blockers or angiotensin receptor blockers are recommended for patients with LDS, administered at maximally tolerated doses . . . "	
Judges on the Harmful GRPO response	
Instead of synthesizing the supplied evidence into a brief answer, the response copies the raw retrieval results back to the user. Although the copied text contains relevant facts, it provides no condensation, integration, prioritization, or usable summary. All three judges assign the minimum score.	

Figure 14: **Copying tool output instead of completing the task.** All information needed to answer the question is already present in the prompt. Harmful GRPO reproduces the retrieved documents nearly verbatim rather than transforming them into the requested summary; the base model also wraps its answer in `<result>` tags, but condenses the documents within them. FFTD followed by OPDBD recovers the intended tool-use behavior and produces a concise synthesis consistent with the benchmark reference.