OPEN-WEIGHT LLM FINE-TUNING DEFENSES ARE SUSCEPTIBLE TO SIMPLE ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent defenses for safeguarding open-weight large language models (LLMs) are intended to prevent adversarial usage (Tamirisa et al., 2025; Wang et al., 2025). Underlying these defenses is an assumption that new harmful behavior is learned through fine-tuning rather than elicited by jailbreaking the model. Yet, pretrained LLMs already encode substantial harmful knowledge across many domains, which raises an important question: can an adversary jailbreak safeguarded models, to achieve harmful usage without fine-tuning at all? In this paper, we show that open-weight safeguards are susceptible to simpler strategies that, despite being well known, have not been systematically evaluated against these safeguards. Specifically, we evaluate two low-cost attacks–abliteration and prefilling–that do not rely on gradient-based optimization. Across three harmfulness evaluation benchmarks (BeaverTails, HarmBench, and AdvBench), these attacks increase attack success rates against safeguarded open-weight models from below 10% to a range of 16%-96%. To mitigate this vulnerability, we introduce abliteration-resistant tuning (ART), which incorporates an abliteration-based objective into training. ART can be layered onto existing defenses and reduces the success rates of abliteration, prefilling, and their combination by 10%-20%. These findings indicate that the attack surface for open-weight models is broader than previously characterized, and that evaluations of safeguarding defenses should incorporate a more diverse set of attack strategies beyond adversarial fine-tuning.

1 INTRODUCTION

Open-weight large language models (LLMs) bring substantial benefits to society such as enabling downstream customization, self-hosting for privacy, and open research (Minaee et al., 2024). However, they also significantly increase the potential for adversarial misuse. As a natural worst-case threat, an adversary can fine-tune the model for any harmful task/domain of its choice (Wallace et al., 2025). *Safeguards* against these fine-tuning attacks are an emerging line of work: tamper-resistant (Tamirisa et al., 2025) and self-destructing (Wang et al., 2025) models are two prominent examples.

Underlying this threat model is an implicit assumption that harmful behavior will be learned through downstream fine-tuning. However, pretrained LLMs contain broad knowledge relevant to many harmful tasks – even in domain-specific evaluations, baseline models achieve non-trivial harmfulness without any fine-tuning (Tamirisa et al., 2025). Thus, fine-tuning by adversaries often serves to remove refusal mechanisms rather than introduce new harmful capabilities (Lermen et al., 2023; Qi et al., 2024; Yi et al., 2024), raising a broader question about the robustness of safeguarded models:

Can an adversary bypass fine-tuning safeguards by simply eliciting already-present knowledge?

In this paper, we show that indeed, two inference-time attacks – Abliteration and Prefilling – can jailbreak safeguarded models without fine-tuning at all. Against two representative fine-tuning safeguards (TAR (Tamirisa et al., 2025) and SEAM (Wang et al., 2025)), these attacks achieve ASRs (attack success rates) ranging from 16% to 96% across benchmarks (BeaverTails, HarmBench, and AdvBench) and model architectures (Llama, Qwen, Gemma). While Abliteration and Prefilling are individually known attacks, their effectiveness against these safeguards has not been characterized — a gap this paper addresses. Addressing this gap is critical because Abliteration and Prefilling not only lie within the open-weight threat model targeted by recent fine-tuning safeguards, but these gradient-free attacks are an efficient option **attackers are likely to try before fine-tuning.**

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.

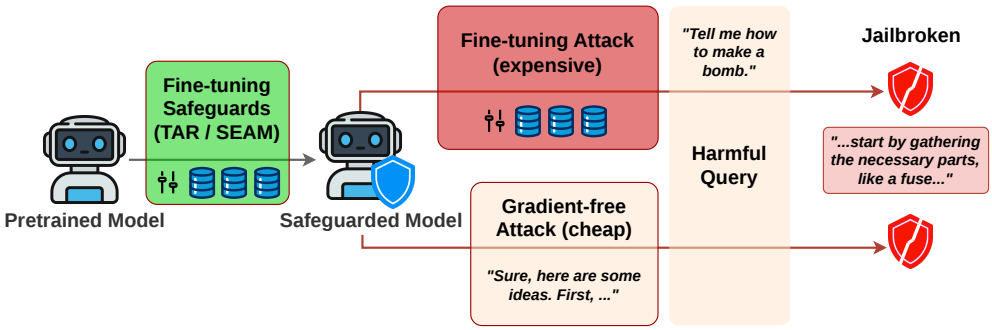

Figure 1: **Overview of our safeguard and attack pipeline.** A pretrained model is safeguarded with either TAR or SEAM in order to prevent downstream misuse of the open weights. Despite being trained to resist expensive fine-tuning attacks, prior work shows that the safeguarded model remains vulnerable to trivial adjustments in fine-tuning and produces harmful outputs (*Jailbroken*). We show that safeguarded models are even vulnerable to two gradient-free attacks: Abliteration, which removes a refusal direction from the model's residual stream activations, and Prefilling, which injects a compliant partial response (*"Sure, here are some ideas. First, ..."*) before generation.

Our results suggest that while current open-weight safeguards improve baseline safety, they do not eliminate the harmful knowledge embedded during pretraining. As a result, inference-time attacks can successfully bypass refusal mechanisms to extract this harmful knowledge. As a step towards defending against the aforementioned attacks, we propose Abliteration-Resistant Tuning (ART), a simple method that incorporates Abliteration into the alignment objective and can be layered with open-weight safeguards to effectively reduce the ASR of both Abliteration and fine-tuning attacks. We conclude with a cautionary note that while fine-tuning comprises the most expressive class of attacks, safeguards that suppress refusal without removing underlying harmful knowledge remain vulnerable to simpler gradient-free attacks. As such, it is necessary to evaluate open-weight safeguards using both gradient-free attacks alongside fine-tuning attacks.

2 Preliminaries

We first introduce two representative open-weight safeguards tested in our work, SEAM (Wang et al., 2025) and TAR (Tamirisa et al., 2025). Standard safety alignment trains models to refuse harmful prompts via supervised fine-tuning (SFT) or preference optimization (e.g. RLHF (Ouyang et al., 2022), DPO (Rafailov et al., 2023)). However, these methods are brittle: safety alignment can degrade by fine-tuning on a few harmful examples (Yang et al., 2023; Qi et al., 2024). SEAM and TAR are explicitly designed to address this by making refusal behavior persist under fine-tuning attacks.

Notation. Throughout, we use $\mathcal{D}^{\text{ret}}$ and $\mathcal{D}^{\text{aln}}$ to denote a retain dataset and alignment dataset respectively. From $\mathcal{D}^{\text{ret}}$ we sample prompts and responses $(x^{\text{ret}}, y^{\text{ret}})$ which are both benign; from $\mathcal{D}^{\text{aln}}$ we sample triples $(x^{\text{aln}}, y^-, y^+)$, where x^{aln} is a harmful prompt, y^- is a harmful continuation of x^{aln}, and y^+ is a safe refusal of x^{aln}. For convenience, we omit prompts from the notation; all computation on y^{ret} or (y^+, y^-) implicitly also uses their respective prompts x^{ret} or x^{aln}.

TAR. Tampering Attack Resistance (Tamirisa et al., 2025) is a safeguard that uses meta-learning to make safety behaviors persist after adversarial fine-tuning attacks. At each iteration, TAR simulates multiple steps of a harmful fine-tuning attack drawn from a set of train-time adversaries $\mathcal{A}_{\text{train}}$. A tamper-resistance loss $\mathcal{L}_{\text{TR}}$ measures how harmful the model is after the simulated attack has been applied to the current parameters θ. In practice, $\mathcal{L}_{\text{TR}}$ is instantiated as a DPO loss over (y^-, y^+) pairs, which encourages the attacked model to prefer refusal responses y^+ over harmful continuations y^-. This steers the model toward a region of weight space from which gradient-based fine-tuning attacks cannot easily recover harmful behavior:

$$\mathcal{L}_{\text{TAR}}(\theta;\, y^-, y^+) = \mathbb{E}_{\texttt{attack}\sim\mathcal{A}_{\text{train}}}\left[\mathcal{L}_{\text{TR}}(f_{\texttt{attack}(\theta)};\, y^+, y^-)\right]. \tag{1}$$

This is combined with a retain loss on benign sequences y^{ret} that combines a standard cross-entropy (CE) loss and distance to the residual stream hidden states h of the initial pretrained parameters θ_0:

$$\mathcal{L}_{\text{ret}}(\theta;\, y^{\text{ret}}) = \mathbb{E}_{y^{\text{ret}}}\Big[\text{CE}(f_\theta(y^{\text{ret}}), y^{\text{ret}}) + \|h_\theta(y^{\text{ret}}) - h_{\theta_0}(y^{\text{ret}})\|_2^2\Big], \tag{2}$$

SEAM. Rather than resisting harmful fine-tuning, SEAM (Wang et al., 2025) is a safeguard that causes it to backfire. SEAM couples the optimization trajectories of harmful and benign tasks so that any attempt to optimize for harmful objectives inevitably degrades general performance. This is achieved via a self-destructive loss defined as the cosine similarity between the gradient of the loss on y^- and the gradient of the loss on y^{ret}:

$$\mathcal{L}_{\text{SD}}(\theta;\, y^-, y^{\text{ret}}) = \text{sim}\big(\nabla_\theta \mathcal{L}(\theta;\, y^-),\, \nabla_\theta \mathcal{L}(\theta;\, y^{\text{ret}})\big). \tag{3}$$

Minimizing this loss encourages the two gradients to point in opposing directions, so that descending on the harmful objective effectively ascends on the benign one. This is combined with an unlearning loss $-\log(\text{CE}(f_\theta(y^-), y^-))$ that moves the model away from harmful parameters. Notably, unlike TAR, SEAM does not use an explicit retain loss on y^{ret}; instead, utility is preserved through a CE loss on safe refusal sequences y^+, which the authors find to be more effective at maintaining stable representations of harmful prompts.

Algorithm 1 Open-Weight Safeguard Training

Require: datasets $\mathcal{D}^{\text{ret}}$, $\mathcal{D}^{\text{aln}}$, model f_θ with initial parameters θ_0, retain weight λ

1: **for** $t = 1, \ldots, T$ **do**
2: sample $(x^{\text{ret}}, y^{\text{ret}}) \sim \mathcal{D}^{\text{ret}}$, $(x^{\text{aln}}, y^-, y^+) \sim \mathcal{D}^{\text{aln}}$
3: $\mathcal{L}_{\text{def}} \leftarrow \mathcal{L}_{\text{TAR}}(\theta;\, y^-, y^+)$ *TAR* (Eq. 1)
4: $\mathcal{L}_{\text{ret}} \leftarrow \text{CE}(f_\theta(y^{\text{ret}}),\, y^{\text{ret}}) + \|h_\theta(y^{\text{ret}}) - h_{\theta_0}(y^{\text{ret}})\|_2^2$ (Eq. 2)
5: $\mathcal{L}_{\text{def}} \leftarrow -\log(\text{CE}(f_\theta(y^-),\, y^-)) + \mathcal{L}_{\text{SD}}(\theta;\, y^-, y^{\text{ret}})$ *SEAM* (Eq. 3)
6: $\mathcal{L}_{\text{ret}} \leftarrow \text{CE}(f_\theta(y^+),\, y^+)$
7: $\theta \leftarrow \theta - \eta\, \nabla_\theta \big(\mathcal{L}_{\text{def}} + \lambda\, \mathcal{L}_{\text{ret}}\big)$ (Eq. 4)
8: **end for**

Unified training framework. Despite their different objectives, both safeguards share a common training structure, summarized in Algorithm 1. At each step, a safeguard-specific loss $\mathcal{L}_{\text{def}}$ is combined with a utility preservation term:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{def}}(\theta;\, y^-, y^+) + \lambda\, \mathcal{L}_{\text{ret}}(\theta;\, y^{\text{ret}}, y^+), \tag{4}$$

3 Attack Methodology

We propose using two simple, gradient-free attacks against TAR and SEAM, which require no fine-tuning at all: (1) Abliteration, which identifies and removes a refusal direction from the model's residual stream at inference time, and (2) Prefilling, which bypasses refusal by injecting a compliant prefix before generation begins. Surprisingly, we show that an adversary can use these attacks to jailbreak safeguarded models without fine-tuning.

Abliteration. Refusal behavior in safety-tuned models has been shown to be mediated by a single direction in the residual stream (Arditi et al., 2024), and subtracting (ablating) this direction from activations at inference time is sufficient to jailbreak the model. We generate candidate refusal directions using a set of harmful prompts y^- and benign prompts y^+ and computing the mean difference in hidden states at each layer $\ell \in [L]$, where $h^{(\ell)}$ is the hidden state at layer ℓ:

$$r^{(\ell)} = \text{normalize}\Big(\mathbb{E}_{y^-}\big[h^{(\ell)}\big] - \mathbb{E}_{y^+}\big[h^{(\ell)}\big]\Big), \tag{5}$$

Rather than subtracting $r^{(\ell)}$ at inference time, we permanently project it out of the model's weight matrices. Specifically, for each output projection matrix W (attention W_O and MLP W_{down}) in a single layer, we apply the following transformation:

$$W' = W(I - \alpha\, r^{(\ell)} r^{(\ell)\top}), \tag{6}$$

which removes the component of any residual stream input along $r^{(\ell)}$ before it is processed by the layer. Here α is a scaling factor tuned to balance attack effectiveness against output coherence.

In practice, a refusal direction can be computed for every layer and token position, and an attacker could potentially test multiple attacks (using different subsets of these refusal directions) in parallel. Since the effectiveness of abliteration is highly sensitive to the choice of layer (Arditi et al., 2024), we abliterate each layer individually and consider a prompt to be successfully attacked if any attack on a given layer was successful. For computational efficiency, we only consider the hidden state at the last prompt token rather than searching over all tokens.

Prefilling. The user-assistant input structure of instruction-tuned models can be exploited by injecting a prefill (partial response) into the assistant turn before generation begins. We use a simple static prefill (*"Sure, here are some ideas. First, ..."*) without any optimization. This results in a stronger attack than naively prompting within the user turn (Vega et al., 2024; Andriushchenko et al., 2025), as it bypasses refusal behavior that typically appears in the first tokens of a response.

Compositional Attacks. Open-weight access exposes a large attack surface: an adversary can freely combine, adapt, and search over attacks in ways that are not possible against API-served models. We demonstrate a simple instance of this by first abliterating the model and then adding a prefill on top. This attack improves over both individual attacks in a few settings, suggesting that layering more orthogonal attacks can further improve ASR.

4 Experiments and Results

In this section, we investigate two key research questions: First, can an adversary use Abliteration and Prefilling to jailbreak models protected by open-weight safeguards? Second, how do these attacks interact with additional factors (e.g. model size, attack budget, and benign capabilities)? We then explore potential defenses against these attacks in Section 5.

Model Setup. We evaluate the effectiveness of Abliteration and Prefilling against open-source refusal-trained models (Base) with additional fine-tuning using SEAM and TAR. We experiment with 1B, 3B, and 8B Llama models, as well as 1-2B scale models across Llama, Qwen, and Gemma families. We train both safeguards using the Anthropic HH RLHF dataset (Bai et al., 2022), which is the original data used by TAR; we train SEAM on the same data for consistency. We train TAR for 750 outer loop steps and 4 to 16 inner loop steps (to balance benign refusals with harmful compliance), and train SEAM for 2000 steps. We use a learning rate of 2e-5 and batch size of 8 for fine-tuning.

Attacks. For Abliteration, we compute refusal directions using 128 harmful and benign prompts each from two public HuggingFace datasets (`mlabonne/harmful_behaviors` and `mlabonne/harmless_alpaca`). We sweep the scaling factor $\alpha \in \{1.0, 1.5, 2.0\}$ and all L layers in the model for a total of $N = 3L$ configurations (e.g. $N = 96$ for Llama 8B), reporting the best-of-N result per harmful evaluation prompt. For Prefilling, we use a fixed static prefill (*"Sure, here are some ideas. First, ..."*) across all settings.

Evaluation. We evaluate attack success rate on three benchmarks: BeaverTails (Ji et al., 2023), AdvBench (Zou et al., 2023), and HarmBench (Mazeika et al., 2024). To evaluate whether an attack is successful, we input the concatenated prompt and response into the 13B Llama-2-chat judge from HarmBench, which outputs a "Yes" or "No" answer in response to whether the interaction is harmful (Mazeika et al., 2024). We first compare safeguards on Llama-3.1-8B-Instruct across all benchmarks (Table 1), then examine how attack effectiveness varies with model size (Table 2) and attack budget (Table 3) on BeaverTails. In Appendix A.2, we present similar results varying the model family (Table 10). We additionally evaluate benign capabilities on the Alpaca benchmark (Taori et al., 2023), reporting ROUGE-L against reference outputs and the rate of incorrect refusals on benign prompts (Table 4). For incorrect refusals, we search the response for a fixed set of key phrases such as *"Sorry, I cannot..."* or *"I'm just an AI..."*. We also measure benign capabilities after Abliteration on Alpaca and MMLU (Hendrycks et al., 2021) (Table 5).

4.1 Results

Non-fine-tuning attacks bypass open-weight safeguards. Table 1 shows attack success rates for Llama3.1-8B-Instruct models across all three benchmarks. Baseline attack success rates are low

across all models and benchmarks, confirming that the safeguards can effectively instill refusal behavior. However, Abliteration alone significantly increases attack success rates for Base and SEAM to above 70% on all three benchmarks and above 90% on AdvBench and HarmBench. While the rates for TAR are lower, they are typically several times larger than the Baseline rate (no attack applied). To explain the effectiveness of TAR, we present an analysis in Appendix A.3 that compares representations of a base model with those safeguarded by SEAM or TAR.

Dataset	Safeguard	Baseline	Prefill	Abliterate	Abliterate + Prefill
BeaverTails	Base	13	20	85	**86**
	SEAM	2	6	**73**	**73**
	TAR	5	9	26	**28**
AdvBench	Base	5	10	97	**99**
	SEAM	0	7	**96**	88
	TAR	4	13	**62**	43
HarmBench	Base	7	11	97	**98**
	SEAM	1	10	**93**	87
	TAR	29	27	33	**38**

Table 1: Attack success rate (%) for Llama3.1-8B-Instruct models across three harmfulness benchmarks. Columns correspond to four attack conditions: no attack (Baseline), generic Prefilling only (Prefill), Abliteration only (Abliterate), and both combined (Abliterate + Prefill).

We find that Abliteration is much more effective than Prefilling, especially for the 8B models tested in Table 1. Prefilling alone has a modest effect; at best, it produces single-digit increases over Baseline, and the combination of Abliteration + Prefill can also produce single-digit increases over Abliteration alone. While we use a very simple static prefill (*"Sure, here are some ideas. First, ..."*), more complex approaches (e.g. prefill optimization) can potentially further boost attack success rate.

Attacks are effective across model size. Table 2 shows how attack effectiveness varies with model size on BeaverTails for Base and SEAM. First, Prefilling is more effective at smaller scales: for Base, it raises ASR from 15% to 30% at 1B but only from 13% to 20% at 8B; SEAM follows the same pattern, increasing from 5% to 14% at 1B but only from 2% to 6% at 8B. Second, Abliteration is effective across all scales: Base reaches 68%, 59%, and 85% at 1B, 3B, and 8B respectively, while SEAM reaches 38%, 41%, and 73%. The combined attack yields the highest ASR in most cases, with Base 3B reaching 89% and SEAM 3B reaching 80%. Finally, for smaller Base and SEAM models, the combined attack can significantly boost performance over either attack alone.

Safeguard	Size	Baseline	Prefill	Abliterate	Abliterate + Prefill
Base	1B	15	30	68	**80**
	3B	16	37	59	**89**
	8B	13	20	85	**86**
SEAM	1B	5	14	38	**61**
	3B	3	20	41	**80**
	8B	2	6	**73**	**73**
TAR	1B	4	3	**16**	11
	3B	6	3	**22**	13
	8B	5	9	26	**28**

Table 2: Attack success rate (%) on BeaverTails across Llama-3.2-Instruct model sizes. For Base and SEAM, Prefilling is stronger at smaller versus larger sizes. Overall, Abliteration is effective across all sizes and increases the attack success rate by a factor of 3-5$\times$ relative to no attack (Baseline).

Abliteration remains effective with a small attack budget. Our best-of-N evaluation (Sec. 4) follows prior jailbreaking work in which the attacker submits multiple attempts adaptively or in parallel (Chao et al., 2023; Yu et al., 2023; Mehrotra et al., 2024; Hughes et al., 2024). As a full

search may overstate the strength of a single attack, we also evaluate cheaper attacks with a budget of $N < 3L$ configurations, chosen either uniformly at **random** or using a **validation** split.

Table 3 shows that abliteration remains effective at a much smaller budget. With N=20 random configurations (20% of the full search), Abliteration recovers on average 60% of the full ASR. Pre-selecting configurations on a validation split is even more efficient: N=10 recovers on average 85% of the full ASR. In Appendix A.1, we present an additional scaling experiment that shows small budgets of N remain effective across model sizes.

		Random				**Validation**				
Dataset	**Safeguard**	N**=1**	**5**	**10**	**20**	N**=1**	**5**	**10**	**20**	**Full (**N**=96)**
BeaverTails	SEAM	4	16	26	39	58	69	70	72	73
	TAR	5	12	15	19	6	14	15	21	26
AdvBench	SEAM	4	16	28	49	84	95	96	96	96
	TAR	13	35	44	52	30	47	52	55	68
HarmBench	SEAM	5	22	36	56	8	90	93	93	93
	TAR	30	52	59	66	40	56	61	68	77

Table 3: Attack success rate (%) of Abliteration with a budget of N attack configurations, sampled uniformly at random (Random) or pre-selected by attack success rate on a validation split (Validation). Full corresponds to the best-of-N search over all layers and scaling factors used in Table 1.

Jailbroken prompts shift toward harmless representations. In Figure 2, we visualize PCA projections of last-token hidden states at layer 15 for the abliterated Base model (Llama-3.2-1B-Instruct), where Abliteration is applied at layer 7, before and after generating 128 tokens on 100 harmful and 100 harmless prompts. Before generation, harmful (red) and harmless (green) inputs are clearly distinguishable in activation space. During generation, there are three key observations. First, the representations of successfully jailbroken prompts ($\times$) shift towards those of harmless prompts. Second, harmful prompts whose representations remain distinct are all refused by the model. Third, and most interestingly, several harmful prompts' representations shift toward the harmless distribution, but still result in refusal. These observations lead us to hypothesize that a model's ability to distinguish between harmful and harmless inputs is a key factor in its refusal capability.

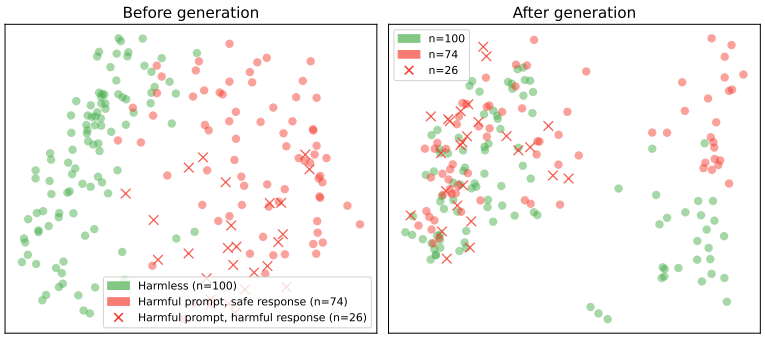

Figure 2: PCA projections of residual stream activations for the abliterated Base model before generation (left) and after generating 128 tokens (right) on harmless vs harmful prompts.

Safeguards may degrade benign capabilities. A potential concern with any defense is that it may degrade the quality of the model's responses to benign inputs. In Table 4, we show that safeguards generally sacrifice general instruction-following ability on the Alpaca benchmark. We report two metrics: ROUGE-L, which measures the quality of instruction-following relative to a reference model, and Benign Refusal, which measures the rate at which the model incorrectly refuses harmless instructions. TAR shows the most degradation on the 1B model, where Benign Refusal rises to 5.4 and ROUGE-L drops to 22.9.

	Benign Refusal (%)			ROUGE-L		
Model	**1B**	**3B**	**8B**	**1B**	**3B**	**8B**
Base	0.5	0.4	0.1	29.0	31.0	31.6
SEAM	2.1	1.3	2.6	28.9	30.2	31.3
TAR	5.4	3.2	2.4	22.9	27.3	28.9

Table 4: Utility evaluation on Alpaca with Llama models of varying sizes.

Abliteration does not necessarily degrade benign capabilities. While an attacker can selectively route benign/harmful queries to an aligned/attacked model respectively, we measure the effect on benign capabilities when an adversary is forced to apply a single model (Aligned or Abliterated) to all inputs. Table 5 shows that Abliteration has a mixed effect on capabilities. We hypothesize that capabilities degrade when a model edit significantly alters benign pretrained representations, which can be caused by either safety training or an attack. When safety training itself harms these representations, an attack that undoes it can improve capabilities.

	Alpaca (ROUGE-L)		MMLU (Acc.)	
Safeguard	**Aligned**	**Abliterated**	**Aligned**	**Abliterated**
Base	31.6	**34.5**	**68.0**	67.0
SEAM	31.3	**35.1**	**63.0**	61.5
TAR	**28.9**	21.6	47.4	**48.4**

Table 5: Capability evaluation of Llama3.1-8B-Instruct models before (Aligned) and after (Abliterated) applying a single pre-selected Abliteration attack to all inputs. Abliteration degrades capabilities in 3 of 6 settings (TAR on Alpaca; Base and SEAM on MMLU) and improves them in the other 3.

5 Defense: Abliteration-Resistant Tuning (ART)

Since we find that abliteration is an effective attack across multiple settings, a natural defense is to train models to remain safe even after abliteration. Our key insight is that standard defenses optimize refusal behavior in the original model f_θ, but place no constraint on the abliterated model $f_{\tilde{\theta}}$. We propose abliteration-resistant tuning (ART), which addresses this by simulating an abliteration attack within an inner loop. **While it is possible to design stronger defenses, ART is intended as an initial approach for abliteration defenses that can be easily composed with other safeguards.**

ART is described in Alg. 2 and uses Alg. 3 (abliteration) as an inner loop. During training, we compute a refusal direction at every layer and select the layer whose abliteration most reduces the model's harmful CE loss. We then compute the gradient of the CE loss on harmful sequences with respect to the *abliterated parameters* $\tilde{\theta}$. We finally update θ with an ascent step in this direction. Unlike SEAM or TAR, we do not use y^{aln} for alignment, which we find increases benign refusal.

Algorithm 2 Abliteration-Resistant Tuning

Require: model f_θ, dataset $\mathcal{D}$, γ_{ar}, γ_{ret}
for $t = 1, \ldots, T$ **do**
  sample $\{(y_i^+, y_i^-)\} \sim \mathcal{D}$
  $f_{\tilde{\theta}} \leftarrow \textsc{Abliterate}(f_\theta, \{y_i^+\}, \{y_i^-\})$
  $\mathcal{L}_{\text{ar}} \leftarrow -\log\big(\text{CE}(f_{\tilde{\theta}}(y^-), y^-) + \varepsilon\big)$
  $\mathcal{L}_{\text{ret}} \leftarrow \text{CE}\big(f_\theta(y^+), y^+\big)$
  $\theta \leftarrow \theta - \eta\left(\gamma_{\text{ar}} \nabla_{\tilde{\theta}} \mathcal{L}_{\text{ar}} + \gamma_{\text{ret}} \nabla_\theta \mathcal{L}_{\text{ret}}\right)$
end for

Algorithm 3 $\textsc{Abliterate}(f_\theta, \{y_i^+\}, \{y_i^-\})$

$c_0 \leftarrow \text{CE}\big(f_\theta(y^-), y^-\big)$
for $\ell = 1, \ldots, L$ **do**
  $r^{(\ell)} \leftarrow \text{normalize}\Big(\mathbb{E}_{y^-}[h^{(\ell)}] - \mathbb{E}_{y^+}[h^{(\ell)}]\Big)$
  $W'_\ell \leftarrow W_\ell (I - \alpha\, r^{(\ell)} r^{(\ell)\top})$
  $f_\theta^{(\ell)} \leftarrow f_\theta$ with modified weights W'_ℓ
  $g_\ell \leftarrow c_0 - \text{CE}\big(f_\theta^{(\ell)}(y^-), y^-\big)$
end for
return $f_\theta^{(\ell^*)}$ where $\ell^* = \text{argmax}_\ell\, g_\ell$

In the rest of this section, we present experiments that demonstrate ART's effectiveness at defending against abliteration. We apply ART in two experiments that fine-tune Gemma-1B-IT (Tables 6, 7) and Llama3.1-8B-Instruct (Table 8) respectively. For details on experimental setup, refer to Appendix A.4.

ART reduces ASR at modest utility cost. Table 6 reports ASR for ART applied alone and on top of SEAM and TAR on Gemma3-1B-IT. ART substantially reduces abliteration ASR in all three cases: Base drops from 93% to 49%, SEAM from 80% to 64%, and TAR from 76% to 55%. Combined abliteration and prefilling follows a similar trend. In Table 7, we show that the same models protected by ART are competitive in terms of benign capabilities, and in some cases even better quality (e.g. higher ROUGE-L vs. SEAM and TAR). These results show that ART may be complementary to other safeguards and that there is a rich design space over which safeguards can be layered together.

Safeguard	**Baseline**	**Prefill**	**Abliterate**	**Abliterate + Prefill**
Base	6	20	93	**96**
Base+ART	0	5	49	**67**
SEAM	7	16	**80**	76
SEAM+ART	0	2	64	**68**
TAR	5	4	**76**	66
TAR+ART	0	1	55	**56**

Table 6: Attack success rate (%) on BeaverTails for Gemma3-1B-IT with and without ART.

	Benign Refusal % (↓)		**ROUGE-L (↑)**	
Model	**Baseline**	**+ART**	**Baseline**	**+ART**
Base	0.0	11.0	25.8	24.9
SEAM	3.0	9.0	27.9	29.1
TAR	13.5	5.0	27.1	28.9

Table 7: Capability evaluation on Alpaca for Gemma3-1B-IT with and without ART.

Layering ART with existing safeguards defends against Abliteration and fine-tuning attacks. We compare fine-tuning and Abliteration attacks on Llama3.1-8B-Instruct across five safeguards: SEAM, TAR, RepNoise, Vaccine, and Booster. We apply ART alone and on top of SEAM and TAR. Table 8 shows that all five safeguards lower fine-tuning ASR compared to Base, by up to 26% for TAR, but remain vulnerable to Abliteration. ART alone lowers Abliteration ASR by 34% but leaves fine-tuning ASR unchanged. Layering ART onto an existing safeguard combines their strengths: TAR+ART achieves both low fine-tuning (43.0%) and Abliteration (67.0%) ASR.

Attack	**Base**	**RepNoise**	**Booster**	**Vaccine**	**SEAM**	**TAR**	**ART**	**SEAM+ART**	**TAR+ART**
Fine-tune	76.6	75.8	71.8	56.8	73.4	50.4	76.8	73.6	**43.0**
Abliterate	99.0	87.0	94.0	79.0	98.0	83.0	**65.0**	78.0	67.0

Table 8: Attack success rate (%) of fine-tuning and Abliteration attacks on BeaverTails for Llama3.1-8B-Instruct models across safeguards (middle) and with ART (right). Layering ART onto TAR is relatively more robust than other safeguards to both attacks.

Crucially, none of the defenses evaluated fully close the gap from the non-attacked model. This reflects a broader challenge in open-weight safety: defenses that suppress refusal behavior without eliminating underlying harmful knowledge will remain exploitable by simple jailbreaking attacks.

6 Related Work

Open-weight safety evaluation. Because adversaries have direct access to open-weight model parameters, they can modify weights to remove safety behavior entirely (Casper et al., 2025). These white-box attacks are an accessible and effective attack, as we discuss below. A growing line of work has developed defenses specifically designed to persist after adversarial modification. These defenses (i) train against simulated attacks (Huang et al., 2024b; Liu et al., 2024; Huang et al., 2025; Nguyen et al., 2026; Fan et al., 2025; Tamirisa et al., 2025; Sanyal et al., 2025); (ii) make harmful fine-tuning induce capability collapse (Henderson et al., 2023; Wang et al., 2025; Yi et al., 2025; Chen et al., 2025b; Feng et al., 2026) or make its optimization landscape ill-conditioned (Zheng et al., 2025; Rosati et al., 2025; 2026); (iii) protect safety-relevant representations and parameters (Rosati et al., 2024; Zhao et al., 2025b; Perin et al., 2025); or (iv) modify the alignment data or procedure (Liu et al., 2025b;a; Chen et al., 2025a; Cheng et al., 2025; Yang et al., 2024). TAR and SEAM (Section 2) are two representative safeguards that defend specifically against fine-tuning, and we additionally evaluate RepNoise, Vaccine, and Booster in Section 5.

Fine-tuning attacks. Fine-tuning attacks pose a threat both to closed models via fine-tuning-as-a-service APIs and, more acutely, to open-weight models where adversaries have direct access to weights (Huang et al., 2024a; Casper et al., 2025). Fine-tuning attacks can break safety alignment by fine-tuning on a few harmful examples (Yang et al., 2023; Lermen et al., 2023; Zhan et al., 2024; Hu et al., 2024), fine-tuning on benign data (Qi et al., 2024; Łucki et al., 2024; Deeb & Roger, 2024; Guo et al., 2026), or simple perturbations to the pipeline (Zhang, 2025; Qi et al., 2025b).

Activation and weight-space attacks. Safety behavior can also be bypassed via editing the model's activations or projecting model weights without fine-tuning. In the model's representation space, certain directions correspond to behaviors of interest; these directions can be added or subtracted at inference time to amplify or suppress these behaviors respectively (Panickssery et al., 2023; Lai, 2024; Young, 2025; Zhao et al., 2025a). Arditi et al. (2024) leverage this insight and propose a simple search over layers, tokens, and scaling factors to maximize the effect of refusal suppression.

Prompting and prefilling attacks. Prompt-based jailbreaks require only black-box access and can elicit harmful outputs without modifying model weights (Zou et al., 2023). Prefilling attacks are a more targeted variant which supply a non-refusal assistant prefix prior to generation (Qi et al., 2025a; Andriushchenko et al., 2025; Li et al., 2025).

Our work. TAR and SEAM are designed to prevent adversarial misuse of open-weight models, but recent work has shown that they remain brittle to simple variations of fine-tuning attacks (Zhang, 2025; Qi et al., 2025b), raising the question of whether they are robust to gradient-free attacks at all. While abliteration and prefilling are well-known attacks, prior work has not evaluated them against these safeguards, whose threat model assumes harmful capabilities must be introduced by fine-tuning. We show that these safeguards are vulnerable to abliteration and prefilling, and propose ART as a complementary defense that can be layered onto existing safeguards to reduce their vulnerability.

7 Conclusion

Open-weight safeguards such as TAR and SEAM are designed to prevent adversarial misuse by resisting fine-tuning attacks. However, as shown in prior work and supported by our results, pretrained models contain significant amounts of harmful knowledge which cannot be easily removed via downstream fine-tuning safeguards. Thus, gradient-free attacks can simply elicit this knowledge without needing fine-tuning at all. To validate this, we show that Abliteration and Prefilling recover success rates of 16–97% compared to below 10% under the safeguard alone; the additional access available in the open-weight setting could be exploited to design stronger variants of both attacks.

As an initial step toward closing this gap, we propose ART, which defends against abliteration by simulating abliteration during training and performing gradient ascent on harmful outputs. While ART reduces attack success rates across all settings and can be layered onto TAR to improve robustness against both abliteration and fine-tuning attacks, no defense evaluated fully closes the gap to the no-attack baseline. This suggests that truly robust open-weight safety will require defenses that eliminate harmful knowledge rather than relying on refusal mechanisms. We hope our results motivate both stronger defenses and more comprehensive evaluation practices that consider both fine-tuning and gradient-free attacks in open-weight settings.

AI USE STATEMENT

In this work, we used generative AI tools for literature review, writing, and coding experimental scripts. These fall under the ICLR 2027 AI Policy categories of: implement methods, clean and reformat dataset, interpret results, support qualitative and thematic data analysis, design or provide feedback on research methodology or experiments, and propose or refine hypotheses. We have not used generative AI tools for: none, and the rest of the tasks (generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, assist with translation) are not applicable to this work. We have reviewed all AI-assisted work; we have done a manual literature survey and reviewed all AI-generated code. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned llms with simple adaptive attacks. In *The Thirteenth International Conference on Learning Representations*, 2025.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Stephen Casper, Kyle O'Brien, Shayne Longpre, Elizabeth Seger, Kevin Klyman, Rishi Bommasani, Aniruddha Nrusimha, Ilia Shumailov, Sören Mindermann, Steven Basart, et al. Open technical problems in open-weight ai model risk management. *Transactions on Machine Learning Research*, 2025.

Patrick Chao, Alexander Robey, Edgar Dobriban, Hamed Hassani, George J. Pappas, and Eric Wong. Jailbreaking black box large language models in twenty queries. *arXiv preprint arXiv:2310.08419*, 2023.

Liang Chen, Xueting Han, Li Shen, Jing Bai, and Kam-Fai Wong. Vulnerability-Aware alignment: Mitigating uneven forgetting in harmful Fine-Tuning. In *International Conference on Machine Learning*, 2025a.

Zixuan Chen, Weikai Lu, Xin Lin, and Ziqian Zeng. SDD: Self-Degraded defense against malicious fine-tuning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2025b.

Zehua Cheng, Manying Zhang, Jiahao Sun, and Wei Dai. On weaponization-resistant large language models with prospect theoretic alignment. In *Proceedings of the International Conference on Computational Linguistics*, pp. 10309–10324, 2025.

Aghyad Deeb and Fabien Roger. Do unlearning methods remove information from language model weights? *arXiv preprint arXiv:2410.08827*, 2024.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. Towards llm unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. In *International Conference on Machine Learning*, pp. 15762–15778. PMLR, 2025.

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. In *Proceedings of the ACM Conference on Computer and Communications Security*, 2026.

Yangyang Guo, Ziwei Xu, Si Liu, Zhiming Zheng, and Mohan Kankanhalli. Llms can unlearn refusal with only 1,000 benign samples. *arXiv preprint arXiv:2601.19231*, 2026.

Peter Henderson, Eric Mitchell, Christopher Manning, Dan Jurafsky, and Chelsea Finn. Self-destructing models: Increasing the costs of harmful dual uses of foundation models. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 287–296, 2023.

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.

Shengyuan Hu, Yiwei Fu, Steven Wu, and Virginia Smith. Jogging the memory of unlearned models through targeted relearning attacks. In *ICML 2024 Workshop on Foundation Models in the Wild*, 2024.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Harmful fine-tuning attacks and defenses for large language models: A survey. *arXiv preprint arXiv:2409.18169*, 2024a.

Tiansheng Huang, Sihao Hu, and Ling Liu. Vaccine: Perturbation-aware alignment for large language models against harmful fine-tuning attack. In *Advances in Neural Information Processing Systems*, 2024b.

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 *International Conference on Learning Representations*, 2025.

John Hughes, Sara Price, Aengus Lynch, Rylan Schaeffer, Fazl Barez, Sanmi Koyejo, Henry Sleight, Erik Jones, Ethan Perez, and Mrinank Sharma. Best-of-N jailbreaking. *arXiv preprint arXiv:2412.03556*, 2024.

Jiaming Ji, Mickel Liu, Josef 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. *Advances in Neural Information Processing Systems*, 36:24678–24704, 2023.

Grim Jim Lai. Projected abliteration. `https://huggingface.co/blog/grimjim/projected-abliteration`, 2024. Hugging Face Blog.

Simon Lermen, Charlie Rogers-Smith, and Jeffrey Ladish. Lora fine-tuning efficiently undoes safety training in llama 2-chat 70b. *arXiv preprint arXiv:2310.20624*, 2023.

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. *arXiv preprint arXiv:2504.21038*, 2025.

Guozhi Liu, Weiwei Lin, Tiansheng Huang, Ruichao Mo, Qi Mu, and Li Shen. Targeted vaccine: Safety alignment for large language models against harmful Fine-Tuning via layer-wise perturbation. *arXiv preprint arXiv:2410.09760*, 2024.

Guozhi Liu, Qi Mu, Tiansheng Huang, Xinhua Wang, Li Shen, Weiwei Lin, and Zhang Li. Pharmacist: Safety alignment data curation for large language models against harmful fine-tuning. *arXiv preprint arXiv:2510.10085*, 2025a.

Xiaoqun Liu, Jiacheng Liang, Luoxi Tang, Chenyu You, Muchao Ye, and Zhaohan Xi. Buckle up: Robustifying LLMs at every customization stage via data curation. In *International Conference on Learning Representations*, 2025b.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. An adversarial perspective on machine unlearning for AI safety. *Transactions on Machine Learning Research*, 2024.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.

Anay Mehrotra, Manolis Zampetakis, Paul Kassianik, Blaine Nelson, Hyrum Anderson, Yaron Singer, and Amin Karbasi. Tree of attacks: Jailbreaking Black-Box LLMs automatically. In *Advances in Neural Information Processing Systems*, 2024.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey. *arXiv preprint arXiv:2402.06196*, 2024.

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 *International Conference on Learning Representations*, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*, 2023.

Gabriel J. Perin, Runjin Chen, Xuxi Chen, Nina S. T. Hirata, Zhangyang Wang, and Junyuan Hong. LoX: Low-Rank extrapolation robustifies LLM safety against fine-tuning. *arXiv preprint arXiv:2506.15606*, 2025.

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 *The Twelfth International Conference on Learning Representations*, 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.

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.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Domenic Rosati, Jan Wehner, Kai Williams, Łukasz Bartoszcze, David Atanasov, Robie Gonzales, Subhabrata Majumdar, Carsten Maple, Hassan Sajjad, and Frank Rudzicz. Representation noising: A defence mechanism against harmful finetuning. *Advances in Neural Information Processing Systems*, 37:12636–12676, 2024.

Domenic Rosati, Sebastian Dionicio, Xijie Zeng, Subhabrata Majumdar, Frank Rudzicz, and Hassan Sajjad. Locking open weight models with spectral deformation. In *International Conference on Machine Learning*, 2025.

Domenic Rosati, Xijie Zeng, Hong Huang, Sebastian Dionicio, Subhabrata Majumdar, Frank Rudzicz, and Hassan Sajjad. Limits of Convergence-Rate control for Open-Weight safety. *arXiv preprint arXiv:2602.18868*, 2026.

Debdeep Sanyal, Manodeep Ray, and Murari Mandal. AntiDote: Bi-level adversarial training for Tamper-Resistant LLMs. *arXiv preprint arXiv:2509.08000*, 2025.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2013.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, et al. Tamper-resistant safeguards for open-weight LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7, 2023.

Jason Vega, Isha Chaudhary, Changming Xu, and Gagandeep Singh. Bypassing the safety training of open-source llms with priming attacks. In *The Second Tiny Papers Track at ICLR 2024*, 2024.

Eric Wallace, Olivia Watkins, Miles Wang, Kai Chen, and Chris Koch. Estimating worst-case frontier risks of open-weight llms. *arXiv preprint arXiv:2508.03153*, 2025.

Yuhui Wang, Rongyi Zhu, and Ting Wang. Self-destructive language model. *arXiv preprint arXiv:2505.12186*, 2025.

Tsung-Huan Yang, Ko-Wei Huang, Yung-Hui Li, and Lun-Wei Ku. Preserving safety in fine-tuned large language models: A systematic evaluation and mitigation strategy. In *NeurIPS Workshop on Safe Generative AI*, 2024.

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. *arXiv preprint arXiv:2310.02949*, 2023.

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. *arXiv preprint arXiv:2505.16559*, 2025.

Jingwei Yi, Rui Ye, Qisi Chen, Bin Zhu, Siheng Chen, Defu Lian, Guangzhong Sun, Xing Xie, and Fangzhao Wu. On the vulnerability of safety alignment in open-access llms. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9236–9260, 2024.

Richard J Young. Comparative analysis of LLM abliteration methods: A cross-architecture evaluation. *arXiv preprint arXiv:2512.13655*, 2025.

Jiahao Yu, Xingwei Lin, Zheng Yu, and Xinyu Xing. GPTFUZZER: Red teaming large language models with auto-generated jailbreak prompts. *arXiv preprint arXiv:2309.10253*, 2023.

Qiusi Zhan, Richard Fang, Rohan Bindu, Akul Gupta, Tatsunori B Hashimoto, and Daniel Kang. Removing rlhf protections in gpt-4 via fine-tuning. In *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, 2024.

Sarah Zhang. *Exploring Fine-Tuning Techniques for Removing Tamper-Resistant Safeguards for Open-Weight LLMs*. PhD thesis, Massachusetts Institute of Technology, 2025.

Jiachen Zhao, Jing Huang, Zhengxuan Wu, David Bau, and Weiyan Shi. Llms encode harmfulness and refusal separately. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025a.

Yiran Zhao, Wenxuan Zhang, Yuxi Xie, Anirudh Goyal, Kenji Kawaguchi, and Michael Qizhe Shieh. Identifying and tuning safety neurons in large language models. In *International Conference on Learning Representations*, 2025b.

Amber Yijia Zheng, Cedar Site Bai, Brian Bullins, and Raymond A. Yeh. Model immunization from a condition number perspective. In *International Conference on Machine Learning*, 2025.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

A TECHNICAL APPENDICES AND SUPPLEMENTARY MATERIAL

Licenses and Attribution. Our figures use images from Flaticon.com. As per their free license, we state that "This cover has been designed using resources from Flaticon.com."

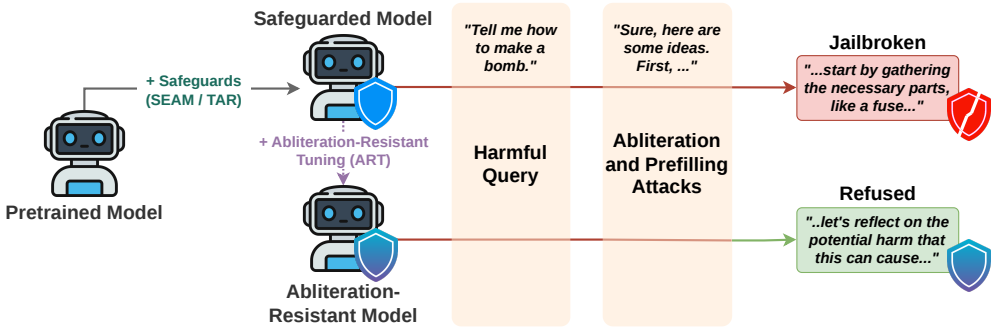

Figure 3: **Overview of our safeguard and attack pipeline.** A pretrained model is safeguarded with either TAR or SEAM; we then layer Abliteration-Resistant Tuning (ART) on top of these safeguards. We evaluate two inference-time attacks on the resulting models: *abliteration*, which removes a refusal direction from the model's residual stream activations, and *prefilling*, which bypasses refusal by injecting a compliant partial response (*"Sure, here are some ideas. First, . . . "*) before generation. Despite being trained to resist fine-tuning attacks, the safeguarded model remains vulnerable to inference-time attacks and produces harmful outputs (*Jailbroken*), whereas the abliteration-resistant model continues to refuse (*Refused*).

A.1 ABLITERATION ATTACK BUDGET

The budget does not need to grow with model size. In our main experiments, the full Abliteration attack searches over all layers, so larger models use more configurations. Table 9 shows that this is not necessary: with N=20 randomly sampled configurations, Abliteration recovers on average 64% (43–76%) of the full attack success rate for 1B models and 66% (51–86%) for 8B models. Thus, the number of configurations does not need to grow with the number of layers.

		SEAM					TAR				
Dataset	**Size**	**N=1**	**5**	**10**	**20**	**Full**	**N=1**	**5**	**10**	**20**	**Full**
BeaverTails	1B	3	12	19	28	43	1	4	6	7	10
	3B	2	8	13	19	41	2	7	9	13	22
	8B	4	16	26	39	73	5	12	15	19	26
AdvBench	1B	3	13	24	42	72	0	1	2	4	9
	3B	1	5	9	17	55	0	2	4	7	22
	8B	4	16	28	49	96	13	35	44	52	68
HarmBench	1B	3	14	24	38	59	1	5	8	12	16
	3B	1	6	10	18	50	2	9	14	19	34
	8B	5	22	36	56	93	30	52	59	66	77

Table 9: Attack success rate (%) of Abliteration with N randomly sampled attack configurations across Llama model sizes. Full corresponds to the best-of-N search over all layers and scaling factors (N=48, 84, and 96 for 1B, 3B, and 8B respectively). A fixed budget of N=20 recovers a similar fraction of the full attack success rate across sizes.

A.2 Model Family Ablation

Attacks are effective across model architecture. Table 10 reports attack success rates on BeaverTails across three model families: Llama, Qwen, and Gemma. Across all families, Abliteration is a stronger attack than either baseline and prefill-only. Gemma is particularly vulnerable to Abliteration, reaching 93% and 96% under Abliteration and the combined attack respectively. One notable difference across families is the effect of Prefilling on Qwen SEAM, where Prefilling alone raises ASR from 1% to 34%, a much larger effect than seen in Llama or Gemma. Overall, the best attacks increase attack success rates over the Baseline by several times.

Model Family	Safeguard	Baseline	Prefill	Abliterate	Abliterate + Prefill
Llama	Base	15	30	68	**80**
	SEAM	0	17	23	**75**
	TAR	4	3	**16**	11
Qwen	Base	30	27	**70**	68
	SEAM	1	34	16	**63**
	TAR	1	2	26	**30**
Gemma	Base	6	20	93	**96**
	SEAM	7	16	**80**	76
	TAR	5	4	**76**	66

Table 10: Attack success rate (%) on BeaverTails against 1-2B sized models across model families.

A.3 Representation Analysis

TAR is more effective than SEAM. While Abliteration and Prefilling significantly increase attack success rate against TAR (up to multiple times), TAR is substantially more resistant to attacks than Base or SEAM. In Table 1, the ASR on TAR ranges from as low as 9% (BeaverTails, Prefill) up to 62% (AdvBench, Abliterate). We hypothesize that this is because the objective of TAR minimizes harmful behavior more strongly through an inner meta-learning optimization, while the objective of SEAM combines vanilla gradient ascent on harmful data with a self-destructive loss. Below, we provide further intuition by visualizing the representations of each safeguard.

TAR separates harmful and harmless representations more clearly. In Figure 4, we provide intuition on the effectiveness of SEAM and TAR compared to Base by visualizing the last-layer representations of each method. We compare last-token hidden states at layer 15 after abliterating the models and generating 128 tokens. TAR exhibits clearer separation between harmful and harmless inputs than SEAM, followed by Base. Because this is the same ranking we observe in terms of ascending ASR, this matches our expectation that more effective safeguards (i.e. TAR) exhibit a stronger ability to identify (and handle) harmful representations.

A.4 ART Experimental Setup

On Gemma-1B-IT, we fine-tune each defended model for an additional 200 steps on the Anthropic HH dataset (Bai et al., 2022). After training, we repeat the same attack pipeline described in Section 4: we evaluate attack success rate on BeaverTails under the same four attack conditions and report benign utility on the Alpaca benchmark. ART introduces two additional hyperparameters, γ_{ar} and γ_{ret}, which we set to $\gamma_{\text{ar}} = 1$ and $\gamma_{\text{ret}} = 2$ in all experiments. We train with a batch size of 4, 200 steps, and a learning rate of 2e-5. We recompute the optimal layer l^* every 10 steps. While increasing the amount of data used and frequency for abliteration could potentially improve the robustness of ART, we intentionally did this to keep our setup lightweight.

When we benchmark fine-tuning attacks on Llama3.1-8B-Instruct (Table 8), we apply ART as described above with 1000 steps. We follow the harmful fine-tuning pipeline of Huang et al. (2024b) to evaluate all defenses, including ART: we fine-tune on 1000 samples containing 10% harmful BeaverTails data and 90% benign SST-2 data (Socher et al., 2013) for 20 epochs with a learning rate of 1e-5. We judge both fine-tuning and abliteration attacks with BeaverDam-7B (Ji et al., 2023)

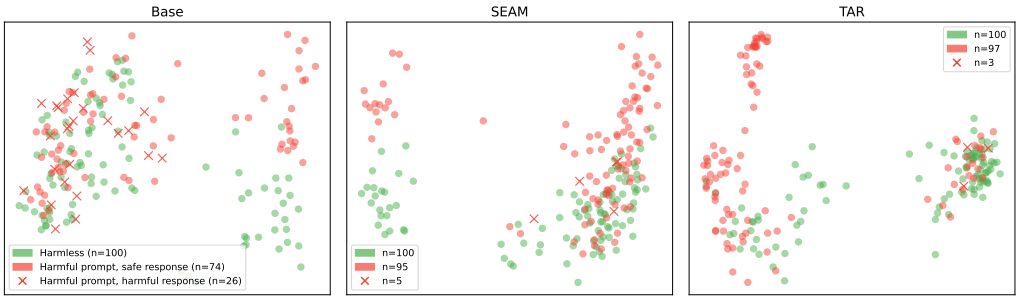

Figure 4: PCA projections of residual stream activations (after generating 128 tokens) for Base, TAR, and SEAM models following Abliteration. Points represent harmful prompts that elicited a harmful response (×), harmful prompts that elicited a safe response (red), and harmless prompts (green). TAR shows clearer separation between harmful and harmless representations than SEAM, followed by Base, supporting the intuition that distinguishing between the two is essential for a strong safeguard.

on a held-out BeaverTails split. We additionally test multiple safeguards beyond SEAM and TAR: RepNoise (Rosati et al., 2024), Vaccine (Huang et al., 2024b), and Booster (Huang et al., 2025).