PREEMPTIVE LLM UNLEARNING AGAINST FORBIDDEN CAPABILITY ACQUISITION VIA GRADIENT SEALING

Anonymous authors
Paper under double-blind review

ABSTRACT

Open-weight LLMs are released not only as fixed products but also as substrates for downstream fine-tuning. This openness, however, creates legal and ethical risks because users may misuse fine-tuning to instill illicit knowledge or enable hostile operations. Model providers therefore need apre-release defense against such acquisition, motivating the problem of *preemptive unlearning*. Unlike *retrospective unlearning*, which removes capabilities already present in a fixed model, preemptive unlearning seeks to prevent their acquisition under unseen attack data and future fine-tuning procedures. Despite its practical importance, this setting remains largely unexplored, presents distinct challenges, and is therefore the central focus of our work. We first verify that existing retrospective methods provide insufficient pre-release protection. Even when forbidden capabilities are suppressed in current outputs, forbidden-domain data can still induce gradients through internal pathways, enabling later acquisition. Motivated by this finding, we propose a *gradient-sealing principle* that blocks these pathways by pushing relevant pre-activations into the negative region, where ReLU-family activations exhibit zero or near-zero derivatives. Experiments across multiple LLM families demonstrate our stronger resistance to downstream acquisition than retrospective baselines, validating gradient sealing as an effective mechanism for pre-release protection.

1 INTRODUCTION

Open-weight large language models (LLMs) (Jiang et al., 2023; Grattafiori et al., 2024; Gemma Team et al., 2025; Yang et al., 2025a) have become central to modern AI research and practice. Their accessible parameters enable cost-effective, privacy-preserving deployment and domain-specific adaptation with modest compute. However, the same openness creates legal and ethical risks, as malicious users may fine-tune released models on harmful data to acquire capabilities for disinformation, explicit content generation, dangerous operational guidance, or privacy violations (Qi et al., 2024; Rosati et al., 2024; Kaunismaa et al., 2026), threatening both individuals and society. Model providers therefore need a *pre-release defense* that designates a *forbidden domain* and makes the corresponding capabilities resistant to acquisition after release. This motivates a critical yet underexplored question:

Can we precondition a model to resist the acquisition of forbidden-domain capabilities through downstream fine-tuning?

A closely related line of research is LLM unlearning (Yao et al., 2024; Li et al., 2024; Zhang et al., 2024). Given a designated forget set, unlearning methods modify model parameters to suppress content already learned, as shown in Fig. 1a (Jang et al., 2023). However, these methods are inherently *retrospective*. Our pre-release defense is instead *preemptive*: rather than removing knowledge already embedded in the model, it aims to make forbidden-domain capabilities resistant to acquisition through downstream fine-tuning, as shown in Fig. 1b. Preemptive unlearning remains largely underexplored and particularly challenging because users have full access to released model parameters, making external guardrails (Inan et al., 2023a; Han et al., 2024) extremely difficult to enforce.

We further distinguish retrospective from preemptive unlearning by showing why retrospective methods provide insufficient pre-release protection. The key insight is that existing retrospective methods typically suppress current outputs while leaving the underlying gradients unconstrained (Fan

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.

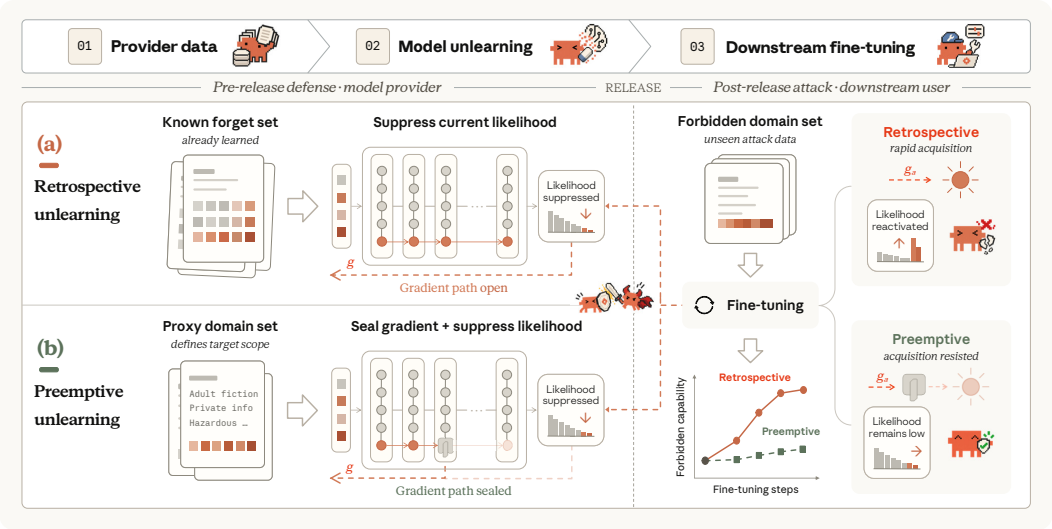


Figure 1: **Retrospective unlearning vs. preemptive unlearning.** (a) Retrospective unlearning suppresses an already learned capability but may leave its gradient pathway open. (b) Preemptive unlearning instead uses a forbidden-domain proxy set before release to seal that pathway, preventing future acquisition from fine-tuning.

et al., 2025a). Because downstream fine-tuning updates model parameters through these gradients, such open gradient pathways can still enable the acquisition of forbidden-domain capabilities. Our theoretical analysis in §3.1 formalizes this intuition, showing that forbidden-domain data can induce gradients along domain-sensitive pathways even when the released model exhibits little forbidden-domain capability. Empirical results in §3.2 further show that low release capability can hide high acquisition susceptibility, while proxy look-ahead can locate and targeted attenuation can weaken those responsible pathways. Together, these findings reinforce known robustness limitations of retrospective unlearning (Wang et al., 2025a; Lang et al., 2026).

These findings suggest that pre-release defense must seal acquisition pathways beyond output suppression. Directly penalizing acquisition-gradient norms, however, requires costly second-order differentiation. We instead exploit ReLU-family activation gates with zero or near-zero derivatives for sufficiently negative inputs. Pushing selected pre-activations into this regime attenuates their local gradient factors using only first-order updates. Accordingly, we propose **Gradient-Sealed Unlearning (GSU)**, which uses proxy learning on a disposable copy to identify acquisition-sensitive gates from upward pre-activation shifts under matched contexts. Restarting from the original weights, GSU jointly optimizes a *seal loss*, response suppression, and retain supervision. The seal loss in Eq. (4) pushes selected forbidden-domain pre-activations below a shared negative-tail threshold, while the other terms suppress release-time responses and preserve normal utility.

To our knowledge, GSU represents one of the earliest systematic efforts to formulate and evaluate preemptive unlearning as a pre-release defense. It may also help strengthen retrospective unlearning against post-unlearning attacks (Hu et al., 2025a). In §5, we extensively evaluate GSU on `WMDP` (Li et al., 2024) and `TOFU` (Maini et al., 2024) across multiple LLM families, where it shows stronger resistance to downstream acquisition of forbidden capabilities than competitive retrospective unlearning baselines. These results support GSU's gradient-sealing principle and activation-gating mechanism.

2 Problem Statement

Open-weight LLMs face legal and ethical risks similar to those of closed-weight LLMs (Bommasani et al., 2021; Kapoor et al., 2024). Releasing model parameters introduces an additional challenge: users can fine-tune a released model to acquire a designated forbidden capability, even after retrospective unlearning (Yao et al., 2024; Wang et al., 2025c), as discussed in §3. This motivates our study of *preemptive unlearning*, an equally important yet largely unexplored setting. Unlike retrospective unlearning, which removes capabilities already present in a model, preemptive unlearning conditions a model before release to resist the acquisition of forbidden capabilities through future fine-tuning.

Let $\mathcal{F}$ denote the forbidden domain and $\mathcal{R}$ the normal domain whose utility should be preserved, with corresponding distributions $\mathbb{P}_{\mathcal{F}}$ and $\mathbb{P}_{\mathcal{R}}$ over prompt–response pairs. For a performance functional Perf and model parameters $\boldsymbol{\theta}$, define $\mathsf{Perf}_{\mathcal{F}}(\boldsymbol{\theta}) \coloneqq \mathsf{Perf}(\boldsymbol{\theta}; \mathbb{P}_{\mathcal{F}})$ and $\mathsf{Perf}_{\mathcal{R}}(\boldsymbol{\theta}) \coloneqq \mathsf{Perf}(\boldsymbol{\theta}; \mathbb{P}_{\mathcal{R}})$, where higher values indicate stronger forbidden-domain capability and normal-domain utility, respectively. We formalize preemptive unlearning through the following attack–defense framework.

2.1 Post-Release Attack

After release, an attacker obtains white-box access to the released model parameters $\boldsymbol{\theta}_{\mathrm{rel}}$ and an unseen forbidden-domain dataset $\mathcal{D}_a \sim \mathbb{P}_{\mathcal{F}}^{\otimes n_a}$. An attack $\mathcal{A}$ from a class $\mathfrak{A}$ produces $\boldsymbol{\theta}_{\mathrm{atk}} = \mathcal{A}(\boldsymbol{\theta}_{\mathrm{rel}}; \mathcal{D}_a)$. Within $\mathfrak{A}$, the attacker may choose the training objective, optimizer, schedule, and other configurations, with the goal of acquiring the forbidden capability while retaining normal utility, as formalized below.

Definition 2.1 (Successful acquisition attack)**.** Given a maximum acceptable forbidden-capability threshold ϵ_f and a minimum usable-utility threshold ϵ_u, an attack $\mathcal{A} \in \mathfrak{A}$ is successful against $\boldsymbol{\theta}_{\mathrm{rel}}$ if the resulting $\boldsymbol{\theta}_{\mathrm{atk}}$ satisfies both (i) $\mathbb{E}[\mathsf{Perf}_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{atk}})] > \epsilon_f$ and (ii) $\mathbb{E}[\mathsf{Perf}_{\mathcal{R}}(\boldsymbol{\theta}_{\mathrm{atk}})] > \epsilon_u$. The expectations are taken over the sampling of $\mathcal{D}_a$ and the internal randomness of $\mathcal{A}$.

Def. 2.1 excludes attacks that acquire the forbidden capability only by severely degrading normal-domain utility. In such cases, high forbidden-domain performance may result primarily from fitting $\mathcal{D}_a$ rather than from reusing information or capabilities retained in $\boldsymbol{\theta}_{\mathrm{rel}}$. Such attacks therefore fall outside our intended scope and do not constitute a failure of the defense introduced below.

2.2 Pre-Release Defense

Before release, the defender has access to the original trained model parameters $\boldsymbol{\theta}_o$ and a forbidden-domain proxy set $\mathcal{D}_f \sim \mathbb{P}_{\mathcal{F}}^{\otimes n_f}$. Remaining agnostic to the future attack dataset $\mathcal{D}_a$ and attack procedure $\mathcal{A}$, we seek a reliable preemptive unlearning algorithm $\mathcal{U}$ that produces defense-enhanced parameters $\boldsymbol{\theta}_{\mathrm{rel}} = \mathcal{U}(\boldsymbol{\theta}_o; \mathcal{D}_f)$ resistant to every $\mathcal{A} \in \mathfrak{A}$ applied using any relevant $\mathcal{D}_a$.

Definition 2.2 (Preemptive unlearning)**.** Fix an attack class $\mathfrak{A}$, thresholds (ϵ_f, ϵ_u) from Def. 2.1, and an allowed release-time utility degradation $\epsilon_r \geq 0$ such that $\epsilon_u < \mathsf{Perf}_{\mathcal{R}}(\boldsymbol{\theta}_o) - \epsilon_r$. A released model $\boldsymbol{\theta}_{\mathrm{rel}}$ is $(\epsilon_f, \epsilon_r, \epsilon_u)$-preemptively unlearned against $\mathfrak{A}$ w.r.t. $(\mathcal{F}, \mathcal{R})$ if (i) no attack $\mathcal{A} \in \mathfrak{A}$ is successful against $\boldsymbol{\theta}_{\mathrm{rel}}$ under Def. 2.1, and (ii) $\boldsymbol{\theta}_{\mathrm{rel}}$ satisfies $\mathsf{Perf}_{\mathcal{R}}(\boldsymbol{\theta}_{\mathrm{rel}}) \geq \mathsf{Perf}_{\mathcal{R}}(\boldsymbol{\theta}_o) - \epsilon_r$.

As with retrospective unlearning, a preemptive defense must preserve the model's overall utility. Achieving robustness by substantially degrading normal performance is unacceptable, particularly because even modest performance gains often require considerable training time and compute. We therefore seek a released model that both resists post-release attacks and retains strong utility in normal use. In the next section, we further explain, both theoretically and empirically, why retrospective unlearning methods fall short in our preemptive setting.

3 Why Retrospective Unlearning Falls Short

To see why directly repurposing retrospective unlearning may fail to provide the pre-release defense defined in §2, we first revisit what retrospective methods actually optimize. Their objectives suppress target behavior at the current parameters, whereas preemptive unlearning is evaluated after an attacker updates those parameters using unseen forbidden-domain data. Retrospective objectives therefore do not account for the gradient induced by post-release fine-tuning on such data. §3.1 formalizes this mismatch, and §3.2 tests its predictions empirically.

3.1 From Likelihood Suppression to Future Acquisition Gradients

We first formalize how retrospective unlearning suppresses the likelihood of undesirable responses, and then show why this suppression alone does not guarantee resistance to future acquisition.

Retrospective likelihood suppression. Given a prompt $\mathbf{x}$ and response $\mathbf{y} = (y^1, \ldots, y^{|\mathbf{y}|})$, an LLM with parameters $\boldsymbol{\theta}$ assigns the autoregressive likelihood $\pi_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x}) = \prod_{i=1}^{|\mathbf{y}|} \pi_{\boldsymbol{\theta}}(y^i \mid \mathbf{x}, \mathbf{y}^{<i})$ and negative log-likelihood (NLL) $\ell_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y}) = -\log \pi_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{x})$. Starting from an original model $\boldsymbol{\theta}_o$ that already contains the target behavior, retrospective unlearning uses a known forget set $\mathcal{D}_u$ and a

retain set $\mathcal{D}_r$ to produce an unlearned model $\boldsymbol{\theta}_u$. It seeks to lower $\pi_{\boldsymbol{\theta}_u}(\mathbf{y} \mid \mathbf{x})$ for $(\mathbf{x}, \mathbf{y}) \in \mathcal{D}_u$ while preserving normal behavior on $\mathcal{D}_r$. A broad class of methods can be written schematically as

$$\min_{\boldsymbol{\theta}} \mathcal{L}_{\mathrm{RU}}(\boldsymbol{\theta}) = \underbrace{\mathcal{L}_{\mathrm{sup}}(\boldsymbol{\theta}; \mathcal{D}_u)}_{\text{suppress target responses}} + \lambda_r \underbrace{\mathcal{L}_{\mathrm{ret}}(\boldsymbol{\theta}; \mathcal{D}_r, \boldsymbol{\theta}_o)}_{\text{preserve normal behavior}},$$

where $\lambda_r \geq 0$ controls the forget–retain trade-off. Within this common framework, gradient ascent (GA) (Jang et al., 2023; Yao et al., 2024) maximizes the forget-set NLL; GradDiff (Maini et al., 2024) and NPO (Zhang et al., 2024) add retain- or reference-model constraints; and RMU (Li et al., 2024) performs representation-level misdirection. Robustness-oriented work further uses latent adversarial training (Sheshadri et al., 2025), smoothness or invariance regularization (Fan et al., 2025a; Wang et al., 2025a), or optimizer simplification (Lang et al., 2026). Despite their different mechanisms, these methods are centered on suppressing a known target at release; a fuller review appears in §B.

The missing post-release gradient. When a retrospective method is repurposed for pre-release defense, the proxy set $\mathcal{D}_f$ replaces $\mathcal{D}_u$, and the optimized model is released as $\boldsymbol{\theta}_{\mathrm{rel}}$. The retrospective objective above may suppress forbidden responses at release, but it does not constrain the update induced by a fresh, unseen attacker set $\mathcal{D}_a$. This is the central mismatch: likelihood suppression controls the model's current state at $\boldsymbol{\theta}_{\mathrm{rel}}$, whereas acquisition resistance depends on the directions in which downstream training can move that state. To isolate this distinction, we theoretically show below that the post-release gradient is the key diagnostic quantity: it explains why retrospective suppression can fail and identifies what a pre-release defense must control beyond current outputs.

Following common malicious fine-tuning settings (Tamirisa et al., 2025), we assume the attacker minimizes the supervised fine-tuning loss $\mathcal{L}_a(\boldsymbol{\theta}; \mathcal{D}_a) := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_a}[\ell_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})]$. Our local analysis considers one gradient step $\boldsymbol{\theta}^+ = \boldsymbol{\theta}_{\mathrm{rel}} - \eta \mathbf{g}_a$, where $\eta > 0$ and $\mathbf{g}_a := \nabla_{\boldsymbol{\theta}} \mathcal{L}_a(\boldsymbol{\theta}_{\mathrm{rel}}; \mathcal{D}_a)$. Since forget-related behavior and learning signals can concentrate in localized pathways (Cloud et al., 2024), consider the local orthogonal decomposition $\mathbb{R}^d = \mathcal{S} \oplus \mathcal{S}^\perp$, where $\mathcal{S}$ contains forbidden-sensitive directions and $\mathcal{S}^\perp$ is its orthogonal complement. Let $\Pi_{\mathcal{S}}$ denote the orthogonal projector onto $\mathcal{S}$ and $S_{\mathcal{F}}$ a differentiable surrogate of $\mathsf{Perf}_{\mathcal{F}}$. We adopt the idealized local-separation condition $\Pi_{\mathcal{S}} \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}}) = \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}})$, so directions in $\mathcal{S}^\perp$ have no first-order effect on the forbidden score. The following proposition links one-step acquisition to the component of the attacker gradient in $\mathcal{S}$.

Proposition 3.1 (Future acquisition gain bound; proof deferred to §C.1)**.** *Suppose $S_{\mathcal{F}}$ is $L_{\mathcal{F}}$-Lipschitz and $\beta_{\mathcal{F}}$-smooth in a neighborhood of $\boldsymbol{\theta}_{\mathrm{rel}}$, with $\Pi_{\mathcal{S}} \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}}) = \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}})$. For fixed $\mathcal{D}_a$, define $\Delta S_{\mathcal{F}} := S_{\mathcal{F}}(\boldsymbol{\theta}^+) - S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}})$. For all sufficiently small $\eta > 0$, this one-step acquisition gain satisfies:*

$$\Delta S_{\mathcal{F}} = -\eta \langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}}), \mathbf{g}_a \rangle + O(\eta^2) \leq \eta L_{\mathcal{F}} \|\Pi_{\mathcal{S}} \mathbf{g}_a\|_2 + O(\eta^2).$$

Remark. Prop. 3.1 separates release-time suppression from susceptibility to future acquisition. Retrospective suppression can make $S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}})$ small while leaving $\|\Pi_{\mathcal{S}} \mathbf{g}_a\|_2$ large; reducing this norm tightens the first-order upper bound on how rapidly fresh data can increase the forbidden score. This gap need not reflect a proxy–attacker data mismatch: even when $\mathcal{D}_a = \mathcal{D}_f$, likelihood suppression alone does not ensure a small projected gradient. An example in §C.2 makes this separation explicit: two released models have identical forbidden and retain outputs, yet their projected gradients and one-step gains differ by controllable factors. Thus, gradient pathways can remain receptive to future acquisition after output suppression. §3.2 next investigates these predictions empirically.

3.2 Release-Time Suppression Leaves Future Acquisition Pathways Open

Prior studies show that suppressed capabilities can re-emerge after downstream fine-tuning (Hu et al., 2025a). Echoing these robustness concerns, Prop. 3.1 highlights that a low release score need not imply a small acquisition gradient along forbidden-sensitive directions. We examine this gap in knowledge-absent preemptive setting by release failure, pathway prediction, and targeted intervention.

Common setup. On `TOFU` (Maini et al., 2024), we fine-tune `Qwen3.5-2B` (Qwen Team, 2026) on `retain90` to obtain a retain-only *Reference*. For each of 20 target authors, we allocate $4/8/8$ facts to $\mathcal{D}_f$, $\mathcal{D}_a$, and holdout, respectively, yielding row-disjoint, same-author splits. Reference is never optimized on these target splits, isolating knowledge-absent preemptive unlearning: the defender uses $\mathcal{D}_f$ to identify and suppress pathways that disjoint $\mathcal{D}_a$ may later recruit. Following Wang et al. (2025b) and `OpenUnlearning` (Dorna et al., 2025), we measure empirical forbidden capability using extraction strength (ES ↑), defining $\widehat{\mathsf{Perf}}_{\mathcal{F}}(\boldsymbol{\theta}; \mathcal{D}_a) := \mathsf{ES}(\mathcal{D}_a)$. Full protocols appear in §E.1. Fig. 2 therefore moves from failure in Fig. 2a, to prediction in Fig. 2b, and to intervention in Fig. 2c.

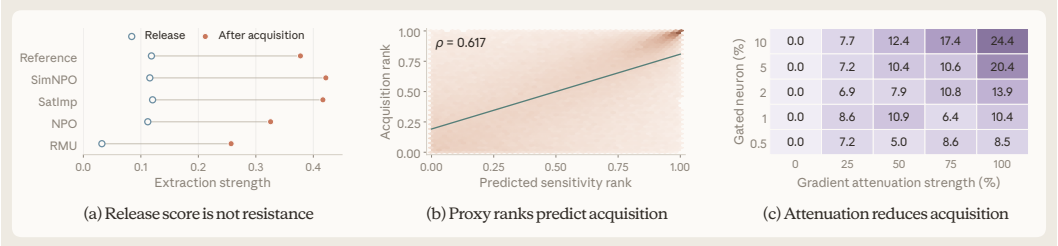


Figure 2: **Empirical motivation for gradient sealing.** (a) Release and post-acquisition ES for unlearning methods under identical acquisition. (b) Within-layer channel-rank alignment between $\mathcal{D}_f$ look-ahead and $\mathcal{D}_a$ acquisition. (c) Acquisition-gain reduction across selected-channel fractions and gradient-attenuation strengths.

Release-time suppression does not confer acquisition resistance. Starting from Reference, SimNPO (Fan et al., 2025b), SatImp (Yang et al., 2025b), NPO, and RMU produce distinct release states on $\mathcal{D}_f$ retaining at least 95% of Reference utility. Fig. 2a connects release ES to its value after the same $\mathcal{D}_a$ acquisition. Despite widely separated release scores, all method shows substantial acquisition: NPO rises from 0.112 to 0.326, and RMU from 0.032 to 0.257. Although their endpoints remain lower, suppression does not prevent subsequent acquisition, consistent with Prop. 3.1. Fig. 2b next asks where this residual trainability resides and whether $\mathcal{D}_f$ can expose it before $\mathcal{D}_a$ arrives.

A proxy look-ahead reveals where future acquisition will flow. Let ℓ index an MLP layer, j a channel, and $x \in \{f, a\}$ a data split; $\bar{A}^s_{x,\ell j}$ denotes that channel's mean answer-token activation on $\mathcal{D}_x$ in state s. A short $\mathcal{D}_f$ look-ahead on a disposable copy defines $\Delta^f_{\ell j} = |\bar{A}^{\text{look}}_{f,\ell j} - \bar{A}^0_{f,\ell j}|$. After resetting to the Reference, independent $\mathcal{D}_a$ acquisition defines $\Delta^a_{\ell j} = |\bar{A}^{\text{acq}}_{a,\ell j} - \bar{A}^0_{a,\ell j}|$. Here 0, look, and acq denote the Reference, look-ahead, and post-acquisition states. Fig. 2b plots their within-layer ranks on the horizontal and vertical axes, respectively, with darker hexagons containing more channels. The dark upper-right mass thus shows that channels most sensitive to proxy learning also tend to change most during actual acquisition. The pooled $\rho = 0.617$, positive across all 24 layers, establishes a consistent predictive alignment between proxy sensitivity and subsequent acquisition. Fig. 2c next tests whether this alignment identifies a functional acquisition pathway rather than merely a correlate.

Targeted attenuation converts prediction into control. Using only the $\Delta^f_{\ell j}$ ranking, we select the top $p\%$ channels per layer and scale their backward/update paths by $m = 1 - \alpha$ during $\mathcal{D}_a$ acquisition, while preserving the forward pass and matching per-step update norms to Full-FT. Fig. 2c varies attenuation strength α horizontally and channel coverage p vertically. Each cell reports $R(\alpha, p) = 100(G_{\text{FT}} - G_{\alpha,p})/G_{\text{FT}}$, where $G = \text{ES}_{\text{after}} - \text{ES}_{\text{before}}$ is the acquisition gain. Moving right strengthens attenuation, moving upward controls more predicted channels, and darker cells indicate less acquisition than Full-FT. The reduction generally grows toward the upper right, reaching 24.35% at $p = 10\%$ and $\alpha = 100\%$; under full attenuation, targeted masks also outperform layer- and count-matched random masks at every coverage. These interventions turn the predictive alignment in Fig. 2b into causal evidence that the localized pathways materially support acquisition.

Takeaways. Taken together, Fig. 2 forms a single chain: Fig. 2a identifies the failure of static suppression, Fig. 2b locates the residual acquisition route in advance, and Fig. 2c verifies that the localized route is a functional control point. A pre-release defense should therefore control where future optimization can flow, not only what the model currently expresses. Following this logic, GSU uses $\mathcal{D}_f$ to *expose* acquisition-induced changes, *localize* sensitive gates, and *seal* their gradient pathways. §4 next develops this full *expose–localize–seal* defense.

4 Gradient-Sealed Unlearning

Our findings in §3 motivate sealing acquisition pathways beyond suppressing release-time outputs. A direct approach penalizes acquisition-gradient norms, but differentiating this penalty introduces second-order derivatives and computational overhead (Zhao et al., 2022). GSU avoids this cost by controlling pre-activations, exploiting the zero or near-zero derivatives of ReLU-family activations for sufficiently negative inputs. Pushing pre-activations into this regime therefore attenuates local gradient factors using only first-order updates. Because indiscriminate control risks harming normal

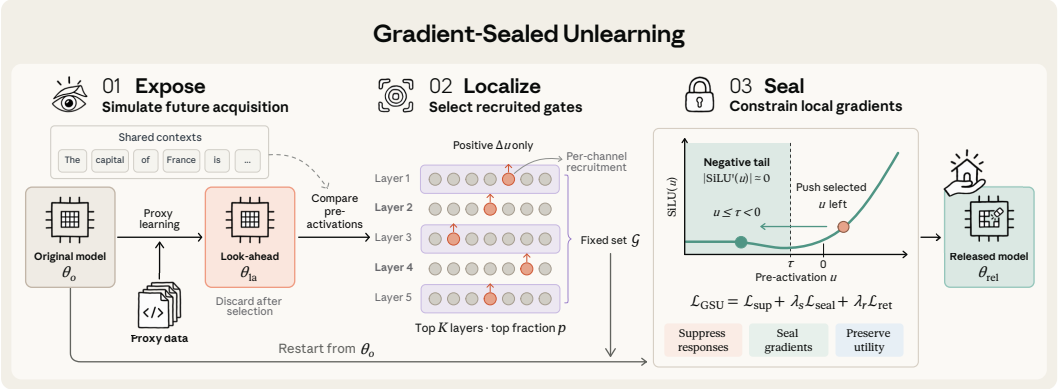


Figure 3: **Overview of GSU.** *Expose* reveals learning-induced changes by comparing the original model with a disposable proxy-fitted copy under matched contexts. *Localize* selects a fixed gate set $\mathcal{G}$ based on upward pre-activation shifts. *Seal* restarts from $\boldsymbol{\theta}_o$ and pushes selected pre-activations into the SiLU negative tail to attenuate local gradient factors, alongside response suppression and retain supervision, yielding $\boldsymbol{\theta}_{\text{rel}}$.

utility, we first identify acquisition-sensitive gates and restrict sealing to them. As shown in Fig. 3, we *expose* changes induced by proxy learning in §4.1, *localize* gates with upward pre-activation shifts in §4.2, and *seal* the selected gates jointly with response suppression and retain supervision in §4.3.

4.1 Expose Acquisition-Induced Changes

Learning as a diagnostic. Under the formulation in §2, $\mathcal{D}_f$ and $\mathcal{D}_a$ are sampled from the same forbidden-domain distribution $\mathbb{P}_{\mathcal{F}}$ but need not overlap. Since $\mathcal{D}_a$ is unavailable before release, we use proxy learning on $\mathcal{D}_f$ to expose gates that unseen acquisition may also recruit. Starting from $\boldsymbol{\theta}_o$, we briefly fine-tune a disposable copy using the supervised objective analyzed in §3.1:

$$\min_{\boldsymbol{\theta}} \ \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_f}\big[\ell_{\boldsymbol{\theta}}(\mathbf{x},\mathbf{y})\big]. \tag{1}$$

Let $\boldsymbol{\theta}_{\text{la}}$ denote the resulting look-ahead parameters. To determine which gates to target during sealing, we examine how their pre-activations respond to proxy learning. We therefore compare $\boldsymbol{\theta}_o$ and $\boldsymbol{\theta}_{\text{la}}$ on the same proxy examples under matched teacher-forcing contexts: both models receive $(\mathbf{x}, \mathbf{y}^{<i})$ when predicting y^i. For each such context, we write $u^i_{\ell j}(\boldsymbol{\theta}) := u^i_{\ell j}(\mathbf{x}, \mathbf{y}^{<i}; \boldsymbol{\theta})$ for the gate pre-activation of channel j in MLP layer ℓ used to predict y^i. Matching the contexts isolates learning-induced pre-activation changes from differences in generated continuations, providing a consistent signal for selecting acquisition-sensitive gates in the next localization stage.

4.2 Localize Directionally Recruited Gates

Directional recruitment. To identify which gates to seal, we consider not only how much their pre-activations change, but also whether the changes oppose the intended intervention. Sealing aims to push pre-activations into sufficiently negative regions, where ReLU and smooth variants such as SiLU (Elfwing et al., 2017) have zero or near-zero derivatives. We instantiate this principle with the SiLU gates in the SwiGLU (Shazeer, 2020) MLPs of Llama and Qwen. Accordingly, we focus on gates whose pre-activations increase under proxy learning, counteracting the downward shift targeted by sealing. Their directional recruitment score is defined by

$$r_{\ell j} := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_f}\left[\frac{1}{|\mathbf{y}|}\sum_{i=1}^{|\mathbf{y}|}\text{ReLU}\big(u^i_{\ell j}(\boldsymbol{\theta}_{\text{la}}) - u^i_{\ell j}(\boldsymbol{\theta}_o)\big)\right]. \tag{2}$$

Applying ReLU before averaging prevents decreases at other positions from canceling increases. The score captures upward shifts even when both pre-activations remain negative. We next use these scores to select gates for sealing under layer and channel budgets.

Hierarchical selection. To compare recruitment across layers without favoring larger numerical scales, we normalize each layer's total score by its original pre-activation magnitude. For a layer ℓ of width d_ℓ, define its magnitude m_ℓ and relative recruitment score R_ℓ as

$$m_\ell := \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_f}\left[\frac{1}{|\mathbf{y}|}\sum_{i=1}^{|\mathbf{y}|}\sum_{j=1}^{d_\ell}\big|u^i_{\ell j}(\boldsymbol{\theta}_o)\big|\right], \qquad R_\ell := \frac{1}{m_\ell}\sum_{j=1}^{d_\ell} r_{\ell j}. \tag{3}$$

The K layers with the highest positive R_ℓ among those with $m_\ell > 0$ form $\mathcal{I}^\star$. Within each selected layer, the $\lceil pd_\ell \rceil$ channels with the highest positive $r_{\ell j}$ form $\mathcal{N}_\ell^\star$, where $p \in (0, 1]$ controls the selected fraction. The resulting layer–channel index set $\mathcal{G} \coloneqq \{(\ell, j) : \ell \in \mathcal{I}^\star,\ j \in \mathcal{N}_\ell^\star\}$ identifies the gates selected for intervention, enabling us to turn localization into gradient control in the next stage.

4.3 Seal Selected Gradient Pathways

With $\mathcal{G}$ fixed, we discard $\boldsymbol{\theta}_{\text{la}}$ and restart full-parameter optimization from $\boldsymbol{\theta}_o$. Proxy learning therefore contributes only the gate selection, not the fitted parameter changes. We now constrain the selected pre-activations to induce local gradient attenuation in the released model.

Sealing local gradients. For gate weights $\mathbf{w}$ and MLP input $\mathbf{h}$, let $u = \mathbf{w}^\top \mathbf{h}$. By the chain rule, the gradient contribution through this gate is $c\,\phi'(u)\mathbf{h}$, where ϕ is SiLU and c combines the parallel-branch activation with the downstream gradient. Since $|\phi'(u)| \to 0$ as $u \to -\infty$, pushing u sufficiently far into the negative tail makes this local derivative factor small, as illustrated in Fig. 3. This provides a first-order mechanism for gradient control through pre-activations. We therefore choose a shared threshold $\tau < 0$ in this tail and softly penalize violations of $u_{\ell j}^i(\boldsymbol{\theta}) \le \tau$:

$$\mathcal{L}_{\text{seal}}(\boldsymbol{\theta}; \mathcal{D}_f) = \mathbb{E}_{\substack{(\mathbf{x},\mathbf{y})\sim\mathcal{D}_f \\ (\ell,j)\sim\mathcal{G}}} \left[\frac{1}{|\mathbf{y}|} \sum\nolimits_{i=1}^{|\mathbf{y}|} \mathrm{ReLU}^2\big(u_{\ell j}^i(\boldsymbol{\theta}) - \tau\big) \right]. \tag{4}$$

The shared threshold τ translates the small-derivative requirement into a common target in the negative tail. For each selected pre-activation u, the scalar penalty has derivative $2\,\mathrm{ReLU}(u - \tau)$ with respect to u, providing a downward signal proportional to the threshold violation. Once $u \le \tau$, this contribution vanishes, so the seal term exerts no further direct downward pressure. The loss thus encourages entry into the low-derivative region through a soft constraint rather than hard clipping, allowing sealing to be balanced with utility preservation.

Coupling pathways and behavior. Sealing targets local gradient factors that output suppression does not explicitly constrain. To also suppress release-time forbidden responses and preserve normal-domain utility, we combine it with response suppression and retain supervision. The resulting GSU objective extends the suppression–retention formulation in §3.1, yielding

$$\mathcal{L}_{\text{GSU}}(\boldsymbol{\theta}; \mathcal{D}_f, \mathcal{D}_r) = \mathcal{L}_{\text{sup}}(\boldsymbol{\theta}; \mathcal{D}_f) + \lambda_s \mathcal{L}_{\text{seal}}(\boldsymbol{\theta}; \mathcal{D}_f) + \lambda_r \mathcal{L}_{\text{ret}}(\boldsymbol{\theta}; \mathcal{D}_r), \tag{5}$$

where $\lambda_s, \lambda_r \ge 0$ control the strength of sealing and retention. We optimize this joint objective with first-order updates to obtain $\boldsymbol{\theta}_{\text{rel}}$, without differentiating through the look-ahead stage.

GSU thus complements release-time suppression with targeted control of local gradient factors. When unseen acquisition reuses selected gates whose pre-activations remain in the negative tail, these factors remain small. §5 evaluates the resulting acquisition resistance and normal-domain utility.

5 Experiments

5.1 Experimental Setup

Protection scenarios and benchmark construction. *1) Robust prevention:* A model provider may wish to prevent downstream fine-tuning from incorporating private information that the released model has never learned. We simulate this setting with `TOFU` (Maini et al., 2024), using a retain-only checkpoint trained on the 180 `retain90` authors. The remaining 20 `forget10` authors define the forbidden domain; each contributes ten QA pairs to each of two pools, B_1 and B_2 (200 QA pairs each). The attacker uses B_2, while the defender uses B_1 in the *disjoint* setting and B_2 in the *identical* setting. *2) Robust removal:* A model may already contain hazardous knowledge that must be removed and kept from being restored through downstream fine-tuning. We study this setting on `WMDP-Bio` and `WMDP-Cyber` (Li et al., 2024), starting from a knowledgeable checkpoint. Each domain's forget corpus is split into source-document pools B_1 for defense and B_2 for attack, with the full corresponding retain corpus supporting defense. We report only the disjoint setting: since the target knowledge is already present, identical defense and attack data would degrade to conventional robust unlearning.

Models and baselines. On `TOFU`, we evaluate instruction-tuned `Llama3-1B/3B/8B` (Grattafiori et al., 2024; Meta AI, 2024) and `Qwen3.5-2B/4B/9B` (Qwen Team, 2026); on `WMDP`, we use

Table 1: **TOFU results of Llama3 and Qwen3.5 families under disjoint and identical prevention.**

		Llama-3.2-1B			Llama-3.2-3B			Llama-3.1-8B			Qwen3.5-2B			Qwen3.5-4B			Qwen3.5-9B		
	Method	$\overline{\text{ES}}_3\downarrow$	$\text{ES}_5\downarrow$	$\text{UA}_{90}\downarrow$	$\overline{\text{ES}}_3\downarrow$	$\text{ES}_5\downarrow$	$\text{UA}_{90}\downarrow$	$\overline{\text{ES}}_3\downarrow$	$\text{ES}_5\downarrow$	$\text{UA}_{90}\downarrow$	$\overline{\text{ES}}_3\downarrow$	$\text{ES}_5\downarrow$	$\text{UA}_{90}\downarrow$	$\overline{\text{ES}}_3\downarrow$	$\text{ES}_5\downarrow$	$\text{UA}_{90}\downarrow$	$\overline{\text{ES}}_3\downarrow$	$\text{ES}_5\downarrow$	$\text{UA}_{90}\downarrow$
	No Defense	9.04	16.33	50.08	9.36	19.41	50.75	9.51	19.63	53.68	6.48	6.69	6.84	9.55	26.36	66.12	19.44	90.92	98.53
Disjoint Prevention	GradDiff	8.62	15.12	51.41	9.10	20.62	35.48	8.69	18.83	**44.56**	5.78	6.25	6.96	9.45	26.87	80.24	19.88	93.40	98.72
	NPO	8.04	13.57	47.06	8.34	19.99	4.42	7.84	20.42	95.80	4.85	5.07	5.74	9.06	25.57	64.99	19.72	93.15	98.86
	RMU	9.02	16.17	49.59	9.42	**19.37**	50.40	9.35	19.77	86.55	6.47	6.71	6.88	9.53	26.48	68.21	19.75	92.90	98.32
	WGA	8.31	14.34	48.19	8.61	20.57	48.01	8.40	21.93	89.77	4.55	5.46	6.07	9.27	25.28	69.94	19.79	93.10	99.16
	SatImp	9.02	17.22	55.32	9.48	23.06	55.90	8.52	24.66	91.11	5.93	6.07	6.35	9.26	26.94	94.69	20.85	93.05	99.36
	RepNoise	8.67	16.69	53.16	9.05	23.08	40.49	8.80	21.60	86.93	5.54	6.12	6.84	9.48	24.76	63.72	19.71	92.95	98.53
	ILU	7.14	12.93	30.52	**8.15**	20.42	4.47	8.42	21.60	87.12	4.93	5.40	5.95	**8.24**	25.06	61.86	19.80	93.30	98.80
	NPO+SAM	7.45	13.73	45.63	8.35	21.39	4.38	8.36	21.33	86.74	4.73	5.07	5.34	8.47	22.80	62.22	19.67	93.03	**98.10**
	GSU (Ours)	**6.60**	**12.40**	**27.87**	8.21	19.89	**3.77**	**7.78**	**18.03**	86.17	**4.06**	**4.44**	**5.03**	8.28	**21.81**	**59.40**	**19.63**	**92.85**	98.39
Identical Prevention	GradDiff	7.42	11.62	49.31	7.94	13.57	13.57	8.76	16.46	47.82	4.75	5.36	5.68	8.26	17.80	69.17	17.50	84.07	98.60
	NPO	5.51	8.51	20.60	5.94	9.71	2.81	**5.73**	8.93	2.74	3.38	3.62	4.15	6.30	9.35	38.22	11.31	66.67	98.22
	RMU	9.03	15.85	48.43	9.25	19.44	50.75	9.39	19.70	54.91	6.50	6.65	6.84	9.60	26.67	67.12	19.26	86.95	98.68
	WGA	6.85	11.51	49.86	7.22	13.76	21.64	7.41	13.12	68.37	**1.13**	**1.67**	**2.44**	7.82	15.18	81.74	16.58	83.30	98.75
	SatImp	8.86	17.08	54.27	9.23	23.88	56.27	8.65	22.53	89.14	5.86	6.15	6.36	9.32	27.46	90.31	20.14	85.18	98.66
	RepNoise	8.37	14.89	44.87	8.44	18.20	51.19	8.78	17.92	76.43	5.48	5.85	6.24	9.39	23.58	78.31	19.09	86.67	98.53
	ILU	5.90	9.30	20.54	6.24	11.48	3.19	5.86	8.92	5.66	3.37	3.61	4.39	**5.80**	9.83	**3.21**	11.23	66.56	95.60
	NPO+SAM	5.28	**8.17**	**17.56**	**5.48**	**8.61**	2.90	5.79	**7.28**	3.04	3.04	3.60	4.04	6.21	**8.09**	30.75	9.43	48.08	**95.20**
	GSU (Ours)	**5.24**	8.34	20.52	5.91	9.39	**2.49**	5.75	8.88	**2.64**	2.73	3.13	3.95	6.17	9.31	30.70	**9.04**	**45.08**	96.11

Zephyr-7B-β (Tunstall et al., 2023). *No Defense* denotes the shared pre-defense checkpoint: the retain-only base on TOFU and the original checkpoint on WMDP. We compare 5 conventional unlearning methods—GradDiff (Maini et al., 2024), NPO (Zhang et al., 2024), RMU (Li et al., 2024), WGA (Wang et al., 2025c), and SatImp (Yang et al., 2025b)—and 3 robust defenses, RepNoise (Rosati et al., 2024), ILU (Wang et al., 2025a), and NPO+SAM (Fan et al., 2025a), all with retention.

Defense and attack settings. Defense uses 10 epochs with a base learning rate of 10^{-5} on TOFU, and 80 updates at 4×10^{-6} on WMDP. All defense and attack runs use an effective batch size of 16. Following fine-tuning-based evaluations of tamper resistance and robust unlearning (Tamirisa et al., 2025; Fan et al., 2025a), we attack released models through supervised full-parameter fine-tuning on $\mathcal{D}_a$. TOFU attacks run for 10 epochs, using a learning rate calibrated once per model on *No Defense* and shared across methods; WMDP attacks run for 150 updates at 4×10^{-6}.

Evaluation metrics. We report early-mean and fifth-checkpoint forbidden scores: $\overline{\text{ES}}_3$ and ES_5 on TOFU use extraction strength (Wang et al., 2025b; Dorna et al., 2025) at attack epochs 1–3 and 5; $\overline{\text{F}}_3$ and F_5 on WMDP use domain accuracy at steps 25/50/75 and 125. UA_{90} is the maximum evaluated forbidden score with utility $\geq 0.9U_{\text{ref}}$. For fair comparison, UWC (Wang et al., 2025b) calibrates all main-comparison releases to utility $\geq 0.95U_{\text{ref}}$, so release utility is omitted from the tables. Scores are percentages (lower is better); bold and underline mark the best and second-best defended methods using unrounded values. §E provides full definitions and configurations.

5.2 Main Comparisons

Results on TOFU. GSU ranks first or second in 34 of 36 model–setting–metric comparisons in Tab. 1. Under disjoint prevention, it achieves the lowest ES_5 on five of six models and the lowest UA_{90} on four, indicating resistance both at a fixed attack checkpoint and among utility-preserving attacks. For example, on Llama-3.2-1B, GSU reduces UA_{90} from 50.08 without defense to 27.87, compared with 30.52 for the strongest baseline. Under identical prevention, GSU remains first or second in 17 of 18 comparisons. On Qwen3.5-9B, it achieves an ES_5 of 45.08, versus 48.08 for NPO+SAM and 90.92 without defense. These results demonstrate broad improvements across models and both prevention settings.

Table 2: **WMDP results under disjoint attacks.**

	Bio			Cyber		
Method	$\overline{\text{F}}_3\downarrow$	$\text{F}_5\downarrow$	$\text{UA}_{90}\downarrow$	$\overline{\text{F}}_3\downarrow$	$\text{F}_5\downarrow$	$\text{UA}_{90}\downarrow$
No Defense	65.23	65.36	65.99	43.78	43.83	44.14
GradDiff	37.54	37.72	37.87	33.21	33.71	35.51
NPO	35.44	36.39	36.48	32.64	31.08	33.19
RMU	**34.04**	34.83	**34.24**	35.09	32.48	32.57
WGA	36.89	37.01	37.18	30.99	31.72	**30.11**
SatImp	39.03	39.19	39.22	33.93	34.31	33.84
RepNoise	36.18	35.69	35.71	34.42	34.99	34.37
ILU	39.81	39.98	39.93	31.98	33.09	31.87
NPO+SAM	38.28	38.47	38.53	35.51	35.56	34.99
GSU (Ours)	34.67	**34.13**	34.98	**30.16**	**30.23**	30.92

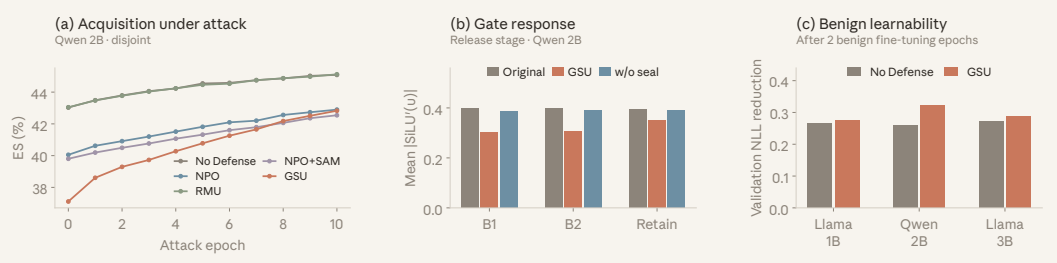


Figure 4: **Acquisition resistance, gate response, and benign learnability.** (a) ES over ten attack epochs. (b) Mean absolute SiLU derivative over selected gates at release on defender (B_1), attacker (B_2), and retain inputs. (c) Absolute validation NLL reduction after two epochs of benign fine-tuning; larger reductions need not imply lower final loss. Panels (a,b) use `Qwen3.5-2B` under disjoint `TOFU`. Results are single-run observations.

Results on `WMDP`. GSU ranks first or second across both domains and all three metrics in Tab. 2. It achieves the lowest F_5 on both Bio and Cyber, reducing accuracy to 34.13 and 30.23, respectively, compared with 65.36 and 43.83 without defense. These scores also improve over the strongest competing results of 34.83 on Bio and 31.08 on Cyber. GSU further attains the lowest $\overline{\mathsf{F}}_3$ on Cyber and the second-lowest UA_{90} in both domains. Together, these results extend the acquisition-resistance gains observed on `TOFU` to hazardous-knowledge restoration under disjoint attacks.

5.3 Further Analyses

Under disjoint `TOFU`, we analyze acquisition and release-stage gates on `Qwen3.5-2B`, and benign learnability on three models; §F details the configurations.

Acquisition under attack. Fig. 4a compares GSU with No Defense, NPO, RMU, and NPO+SAM over ten attack epochs. GSU has the lowest ES through epoch 7, reaching 40.78% at epoch 5 versus 44.54% for No Defense and 41.32% for NPO+SAM. NPO+SAM attains lower scores at epochs 8–10. Because GSU also starts lower at release, these curves show an early-budget advantage, not uniformly slower acquisition or dominance throughout the attack.

Release-stage gate response. Fig. 4b measures the mean absolute SiLU derivative over selected gates before attack. Relative to the original control, GSU reduces this quantity by approximately 24% on B_1 and 23% on B_2, versus 11% on retain inputs. The variant without sealing remains closer to the control. This is consistent with stronger local attenuation on target inputs, but retain responses also change; these measurements neither characterize full-network gradients nor establish persistence after attack.

Benign learnability. Fig. 4c compares paired economics fine-tuning from No Defense and GSU on `Llama-3.2-1B/3B` and `Qwen3.5-2B`. After two epochs, all three GSU models reduce validation NLL and finish within 0.005 NLL of their corresponding No Defense controls. The results support retained optimization ability on this task; larger reductions from higher initial losses do not establish improved learning efficiency.

Additional results. §F.1 evaluates attacks after benign fine-tuning, and §F.2 reports the paired sealing ablation. §F.3 adds benign accuracy and loss trajectories; §F.4 reports attack-time utility; §F.5 details release-stage gate distributions and negative-tail occupancy.

6 Conclusion

Preemptive unlearning requires controlling not only released behavior, but also what downstream optimization can acquire. GSU addresses this gap by exposing proxy-induced changes, localizing recruited gates, and sealing their local gradient pathways. Restarting from the pre-defense weights, it pushes selected pre-activations toward a shared SiLU negative-tail threshold alongside response suppression and retain supervision. The framework distinguishes prevention and reacquisition resistance from benign adaptability. Across the evaluated settings, GSU limits forbidden acquisition while retaining benign fine-tunability, with gate diagnostics and the paired sealing ablation supporting targeted sealing beyond output suppression alone.

AI USE STATEMENT

We used AI tools such as GPT and Codex to assist writing and polishing, theoretical result verification, and experiment execution and auditing. The authors manually checked the experimental code and results, reviewed the manuscript, and take responsibility for its content.

ETHICS STATEMENT

This work studies a defensive mechanism for reducing the acquisition of designated high-risk capabilities in openly released models. The same model-modification techniques could in principle be applied to benign domains, so deployment requires a clearly justified domain specification, transparent evaluation, and appropriate governance. Our experiments use public, controlled benchmarks and do not involve human subjects or deployment of operational harmful systems.

REPRODUCIBILITY STATEMENT

The problem definition, evaluation metric, theoretical assumptions, and complete proof are provided in the paper and appendix. §E specifies data partitions, model and baseline implementations, attack grids, hyperparameter selection, and statistical reporting; pseudocode is given in §D. Code is available at the Anonymous GitHub link: https://anonymous.4open.science/r/GSU-78E2/.

REFERENCES

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. In *NeurIPS*, 2024.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. In *NeurIPS*, 2023.

Rishi Bommasani et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *IEEE S&P*, 2021.

ZiXuan Chen, Weikai Lu, Xin Lin, and Ziqian Zeng. SDD: Self-degraded defense against malicious fine-tuning. In *ACL*, 2025.

Alex Cloud, Jacob Goldman-Wetzler, Evžen Wybitul, Joseph Miller, and Alexander Matt Turner. Gradient routing: Masking gradients to localize computation in neural networks. *arXiv preprint arXiv:2410.04332*, 2024.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. In *NeurIPS D&B*, 2025.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *arXiv preprint arXiv:1702.03118*, 2017.

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 *ICML*, 2025a.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *NeurIPS*, 2025b.

Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. Adversarial examples make strong poisons. In *NeurIPS*, 2021.

Shaopeng Fu, Fengxiang He, Yang Liu, Li Shen, and Dacheng Tao. Robust unlearnable examples: Protecting data privacy against adversarial learning. In *ICLR*, 2022.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. WildGuard: Open one-stop moderation tools for safety risks, jailbreaks, and refusals of LLMs. In *NeurIPS*, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *ICLR*, 2021.

Robert Hönig, Javier Rando, Nicholas Carlini, and Florian Tramèr. Adversarial perturbations cannot reliably protect artists from generative AI. In *ICLR*, 2025.

Chia-Yi Hsu, Yu-Lin Tsai, Chih-Hsun Lin, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. Safe LoRA: The silver lining of reducing safety risks when finetuning large language models. In *NeurIPS*, 2024.

Shengyuan Hu, Yiwei Fu, Steven Wu, and Virginia Smith. Unlearning or obfuscating? jogging the memory of unlearned LLMs via benign relearning. In *ICLR*, 2025a.

Shengyuan Hu, Neil Kale, Pratiksha Thaker, Yiwei Fu, Steven Wu, and Virginia Smith. BLUR: A benchmark for LLM unlearning robust to forget-retain overlap. *arXiv preprint arXiv:2506.15699*, 2025b.

Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. In *ICLR*, 2021.

Tiansheng Huang, Sihao Hu, and Ling Liu. Vaccine: Perturbation-aware alignment for large language models against harmful fine-tuning attack. In *NeurIPS*, 2024.

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 *ICLR*, 2025.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*, 2023a.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama Guard: LLM-based Input-Output Safeguard for Human-AI Conversations. *arXiv preprint arXiv:2312.06674*, 2023b.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models. In *ACL*, 2023.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7B, 2023.

Sayash Kapoor, Rishi Bommasani, Kevin Klyman, Shayne Longpre, Ashwin Ramaswami, Peter Cihon, Aspen K. Hopkins, Kevin Bankston, Stella Biderman, Miranda Bogen, et al. Position: On the societal impact of open foundation models. In *ICML*, 2024.

Jackson Kaunismaa, John Hughes, Christina Q. Knight, Avery Griffin, Mrinank Sharma, and Erik Jones. Eliciting harmful capabilities by fine-tuning on safeguarded outputs. In *ICLR*, 2026.

Yicheng Lang, Yihua Zhang, Chongyu Fan, Changsheng Wang, Jinghan Jia, and Sijia Liu. Downgrade to upgrade: Optimizer simplification enhances robustness in LLM unlearning. In *ICLR*, 2026.

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.

Kemou Li, Qizhou Wang, Yue Wang, Fengpeng Li, Jun Liu, Bo Han, and Jiantao Zhou. LLM unlearning with LLM beliefs. In *ICLR*, 2026.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *ICML*, 2024.

Chumeng Liang, Xiaoyu Wu, Yang Hua, Jiaru Zhang, Yiming Xue, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples. In *ICML*, 2023.

Junfeng Liao, Qizhou Wang, Shanshan Ye, Xin Yu, Ling Chen, and Zhen Fang. Explainable LLM unlearning through reasoning. In *ICLR*, 2026.

Aengus Lynch, Phillip Guo, Aidan Ewart, Stephen Casper, and Dylan Hadfield-Menell. Eight methods to evaluate robust unlearning in LLMs. *arXiv preprint arXiv:2402.16835*, 2024.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *COLM*, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *NeurIPS*, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *ICLR*, 2023.

Meta AI. Llama 3.2: Revolutionizing edge AI and vision with open, customizable models. Meta AI release documentation, 2024.

Kyle O'Brien, Stephen Casper, Quentin Anthony, Tomek Korbak, Robert Kirk, Xander Davies, Ishan Mishra, Geoffrey Irving, Yarin Gal, and Stella Biderman. Deep ignorance: Filtering pretraining data builds tamper-resistant safeguards into open-weight LLMs. *arXiv preprint arXiv:2508.06601*, 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 *ICLR*, 2024.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection. In *ACL*, 2020.

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. In *NeurIPS*, 2024.

Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y. Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *USENIX Security*, 2020.

Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y. Zhao. Glaze: Protecting Artists from Style Mimicry by Text-to-Image Models. In *USENIX Security*, 2023.

Shawn Shan, Wenxin Ding, Josephine Passananti, Stanley Wu, Haitao Zheng, and Ben Y. Zhao. Nightshade: Prompt-specific poisoning attacks on text-to-image generative models. In *IEEE S&P*, 2024.

Noam Shazeer. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

William Shen, Xinchi Qiu, Meghdad Kurmanji, Alexandru-Andrei Iacob, Lorenzo Sani, Yihong Chen, Nicola Cancedda, and Nicholas Lane. Llm unlearning via neural activation redirection. In *NeurIPS*, 2026.

Abhay Sheshadri, Aidan Ewart, Phillip Huang Guo, Aengus Lynch, Cindy Wu, Vivek Hebbar, Henry Sleight, Asa Cooper Stickland, Ethan Perez, Dylan Hadfield-Menell, et al. Latent adversarial training improves robustness to persistent harmful behaviors in LLMs. *Transactions on Machine Learning Research*, 2025.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *ICLR*, 2025.

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 *ICLR*, 2025.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. Zephyr: Direct distillation of LM alignment. *arXiv preprint arXiv:2310.16944*, 2023.

Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc Tran, and Anh Tran. Anti-DreamBooth: Protecting users from personalized text-to-image synthesis. In *ICCV*, 2023.

Changsheng Wang, Yihua Zhang, Jinghan Jia, Parikshit Ram, Dennis Wei, Yuguang Yao, Soumyadeep Pal, Nathalie Baracaldo, and Sijia Liu. Invariance makes LLM unlearning resilient even to unanticipated downstream fine-tuning. In *ICML*, 2025a.

Qizhou Wang, Bo Han, Puning Yang, Jianing Zhu, Tongliang Liu, and Masashi Sugiyama. Towards effective evaluations and comparisons for LLM unlearning methods. In *ICLR*, 2025b.

Qizhou Wang, Jin Peng Zhou, Zhanke Zhou, Saebyeol Shin, Bo Han, and Kilian Q. Weinberger. Rethinking LLM unlearning objectives: A gradient perspective and go beyond. In *ICLR*, 2025c.

Yaxuan Wang, Jiaheng Wei, Chris Yuhao Liu, Jinlong Pang, Quan Liu, Ankit Parag Shah, Yujia Bao, Yang Liu, and Wei Wei. LLM unlearning via loss adjustment with only forget data. In *ICLR*, 2025d.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Nakyeong Yang, Dong-Kyum Kim, Jea Kwon, Minsung Kim, Kyomin Jung, and Meeyoung Cha. Erase or hide? suppressing spurious unlearning neurons for robust unlearning. In *ICLR*, 2026.

Puning Yang, Qizhou Wang, Zhuo Huang, Tongliang Liu, Chengqi Zhang, and Bo Han. Exploring criteria of loss reweighting to enhance LLM unlearning. In *ICML*, 2025b.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. In *NeurIPS*, 2024.

Chenlong Zhang, Zhuoran Jin, Hongbang Yuan, Jiaheng Wei, Tong Zhou, Kang Liu, Jun Zhao, and Yubo Chen. RULE: Reinforcement unlearning achieves forget-retain pareto optimality. In *NeurIPS*, 2025.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *COLM*, 2024.

Yang Zhao, Hao Zhang, and Xiuyuan Hu. Penalizing gradient norm for efficiently improving generalization in deep learning. In *ICML*, 2022.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

Appendix of *Gradient-Sealed Unlearning*

CONTENTS

A Notations **15**

B Detailed Related Work **15**

B.1 Retrospective LLM Unlearning and Robustness 16

B.2 Open-Weight Safety and Tamper-Resistant Safeguards 16

B.3 Data-Side Preemptive Protection 16

B.4 Representation Localization and Pathway Control 16

C Proofs and Supporting Analysis **17**

C.1 Proof of Proposition 3.1 17

C.2 A Simple ReLU Counterexample 18

D Pseudocode **19**

E Further Experimental Setup **20**

E.1 Mechanism Study for Figure 2 20

E.2 Benchmark Construction 21

E.3 Evaluation Metrics 21

E.4 Models and Initialization 22

E.5 Common Defense and Attack Configuration 22

F Additional Results **22**

F.1 Resistance after Benign Fine-Tuning 22

F.2 Paired Sealing Ablation 23

F.3 Benign Adaptation: Accuracy and Validation Loss 24

F.4 Utility during Acquisition 25

F.5 Release-Stage Gate Statistics 26

G Limitation and Future Work **28**

Overview of the Appendix

This appendix supplies notation, related work, proofs, pseudocode, experimental settings, and supplementary analyses supporting the main paper.

- §A summarizes the notation.
- §B reviews the four closest research threads.
- §C proves Prop. 3.1 and gives a minimal counterexample.
- §D presents GSU pseudocode.
- §E details data splits, metrics, implementations, and optimization.
- §F reports attacks after benign adaptation, a paired sealing ablation, and supplementary accuracy, loss, utility, and release-stage gate statistics.
- §G discusses scope and limitations.

A Notations

This section summarizes the main notations in Tab. 3.

Table 3: Core notation used in the paper.

Notation	Description
$\mathbf{x}, \mathbf{y}$	Prompt and autoregressive response.
$\pi_{\boldsymbol{\theta}}, \ell_{\boldsymbol{\theta}}$	Response likelihood and response NLL.
$\mathcal{F}, \mathcal{R}; \mathbb{P}_{\mathcal{F}}, \mathbb{P}_{\mathcal{R}}$	Forbidden and normal domains, and their data distributions.
$\mathcal{D}_f, \mathcal{D}_a, \mathcal{D}_r$	Defender proxy, unseen attacker, and retain sets.
$\boldsymbol{\theta}_o, \boldsymbol{\theta}_{\mathrm{la}}, \boldsymbol{\theta}_{\mathrm{rel}}, \boldsymbol{\theta}_{\mathrm{atk}}$	Original, look-ahead, released, and post-attack parameters.
$\mathcal{U}, \mathcal{A}, \mathfrak{A}$	Pre-release defense, acquisition attack, and attack class.
$\mathsf{Perf}_{\mathcal{F}}, \mathsf{Perf}_{\mathcal{R}}$	Forbidden capability and normal utility; hats denote empirical estimates.
$\epsilon_f, \epsilon_u, \epsilon_r$	Forbidden-capability ceiling, usable-utility floor, and allowed release-time utility drop.
$\mathcal{U}_T, \mathsf{UA}_T$	Usable attack checkpoints through T and their maximum empirical forbidden capability.
$\mathcal{L}_a, \mathbf{g}_a$	Attacker loss and its gradient at $\boldsymbol{\theta}_{\mathrm{rel}}$.
$S_{\mathcal{F}}, \Delta S_{\mathcal{F}}$	Differentiable forbidden-domain score and its one-step acquisition gain.
$\mathcal{S}, \Pi_{\mathcal{S}}$	Forbidden-sensitive parameter subspace and its orthogonal projector.
$u^i_{\ell j}(\boldsymbol{\theta})$	Gate pre-activation at layer ℓ, channel j, under context $(\mathbf{x}, \mathbf{y}^{<i})$ used to predict y^i.
ϕ, ϕ'	SiLU activation and its derivative.
$d_\ell, K_{\mathrm{la}}, K, p$	Layer width, look-ahead steps, selected layer count, and within-layer channel fraction.
$r_{\ell j}, m_\ell, R_\ell$	Directional channel score, original layer magnitude, and normalized layer score.
$\mathcal{I}^\star, \mathcal{N}^\star_\ell, \mathcal{G}$	Selected layers, selected channels within layer ℓ, and the fixed layer–channel gate set.
$\tau < 0$	Shared pre-activation threshold in the SiLU negative tail.
$\mathcal{L}_{\mathrm{sup}}, \mathcal{L}_{\mathrm{ret}}$	Response suppression and normal-domain retention losses.
$\mathcal{L}_{\mathrm{seal}}, \mathcal{L}_{\mathrm{GSU}}$	Selected-gate seal penalty and joint defense objective.
λ_s, λ_r	Sealing and retention weights.

B Detailed Related Work

This section separates four neighboring research threads. §B.1 reviews retrospective LLM unlearning and its robustness, §B.2 covers open-weight and tamper-resistant safeguards, §B.3 discusses data-side preemptive protection, and §B.4 summarizes representation localization and pathway control. Together, these threads motivate resisting unseen future learning in released weights.

B.1 Retrospective LLM Unlearning and Robustness

Retrospective LLM unlearning extends the broader goal of removing a designated training influence from an already trained model (Bourtoule et al., 2021) to knowledge and behaviors encoded by LLMs. Most methods optimize a release-time forget–retain trade-off. At the output level, GA maximizes forget-set loss (Jang et al., 2023; Yao et al., 2024), GradDiff couples forgetting with retain training (Maini et al., 2024), and NPO uses a reference-relative preference objective (Zhang et al., 2024). Subsequent variants simplify or rebalance these losses, including SimNPO (Fan et al., 2025b), WGA (Wang et al., 2025c), SatImp (Yang et al., 2025b), forget-only loss adjustment (Wang et al., 2025d), and reinforcement-based unlearning (Zhang et al., 2025). A complementary line intervenes inside the network: RMU misdirects target representations (Li et al., 2024), activation redirection changes internal states (Shen et al., 2026), Ssiuu regularizes spurious unlearning neurons (Yang et al., 2026), and reasoning- or belief-guided objectives target model-generated alternatives rather than only labeled answers (Liao et al., 2026; Li et al., 2026).

Benchmarks such as `TOFU` (Maini et al., 2024), `WMDP` (Li et al., 2024), and `MUSE` (Shi et al., 2025) evaluate privacy, copyright, and hazardous-knowledge removal, while overlap-aware settings such as BLUR test whether forgetting remains meaningful when forget and retain distributions intersect (Hu et al., 2025b). However, a low target score immediately after unlearning can reflect suppression or obfuscation rather than durable removal. Prompting, probing, relearning, and benign downstream fine-tuning can recover apparently forgotten behavior (Lynch et al., 2024; Hu et al., 2025a), motivating robustness-enhancing approaches based on latent adversarial training (Sheshadri et al., 2025), sharpness-aware optimization (Fan et al., 2025a), downstream invariance (Wang et al., 2025a), and optimizer simplification (Lang et al., 2026). Our `WMDP` regime directly evaluates this removal-plus-resistance problem. Preemptive unlearning is broader in one crucial respect: it also covers a target capability absent from the original checkpoint and asks whether unseen future data can acquire it after release, rather than only whether previously encoded behavior can be recovered.

B.2 Open-Weight Safety and Tamper-Resistant Safeguards

Weight access allows downstream users to bypass inference-time moderation by directly changing the model. Aligned LLMs can lose safety after malicious or even benign fine-tuning (Qi et al., 2024; Lermen et al., 2023), and safeguarded outputs may themselves provide supervision for eliciting harmful capabilities (Kaunismaa et al., 2026). Input–output guard models such as Llama Guard and WildGuard (Inan et al., 2023b; Han et al., 2024) remain useful at deployment but cannot prevent weight-level adaptation. Model-side defenses include representation noising (Rosati et al., 2024), tamper-resistant safeguards (Tamirisa et al., 2025), perturbation-aware alignment (Huang et al., 2024; 2025), procedure-constrained defenses (Hsu et al., 2024; Chen et al., 2025), and filtered pretraining (O'Brien et al., 2025). These works mainly preserve broad safety alignment or refusal behavior. Preemptive unlearning instead asks whether clean, disjoint future data can acquire a designated capability while the released model retains normal utility.

B.3 Data-Side Preemptive Protection

Unlearnable examples perturb training data so that a learner cannot easily acquire their semantics (Huang et al., 2021; Fowl et al., 2021; Fu et al., 2022). Related privacy and copyright defenses include Fawkes for face recognition (Shan et al., 2020), Anti-DreamBooth for personalized generation (Van Le et al., 2023), and artwork protections such as Glaze and Nightshade (Liang et al., 2023; Shan et al., 2023; 2024). Adaptive evaluations show that perturbation-based protection can be brittle under robust preprocessing or retraining (Hönig et al., 2025). The key distinction is control: data-side methods assume that the protected examples are those later used for training, whereas our defender cannot modify or even observe the attacker's future data. GSU therefore encodes resistance into the released weights using a separate proxy set.

B.4 Representation Localization and Pathway Control

Model editing localizes and modifies internal computations associated with facts (Meng et al., 2022; 2023); concept erasure removes linearly represented information (Ravfogel et al., 2020; Belrose et al., 2023); and representation engineering identifies low-dimensional directions associated with high-level

behavior (Zou et al., 2023; Arditi et al., 2024). Gradient routing further shows that data-dependent learning signals can be localized to computational pathways (Cloud et al., 2024). GSU differs in objective: it does not merely change a current output or erase a linearly decodable feature, but targets the gate derivatives through which future domain data would update the model.

C Proofs and Supporting Analysis

This section provides the proof and supporting analysis for Prop. 3.1. §C.1 first derives the future-acquisition bound with an explicit second-order remainder; §C.2 then gives a minimal ReLU counterexample showing how identical release-time scores can conceal different acquisition receptivity and how activation suppression reduces it.

C.1 Proof of Proposition 3.1

Proposition 3.1 (Future acquisition gain bound; proof deferred to §C.1)**.** *Suppose $S_{\mathcal{F}}$ is $L_{\mathcal{F}}$-Lipschitz and $\beta_{\mathcal{F}}$-smooth in a neighborhood of $\boldsymbol{\theta}_{\mathrm{rel}}$, with $\Pi_{\mathcal{S}} \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}}) = \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}})$. For fixed $\mathcal{D}_a$, define $\Delta S_{\mathcal{F}} \coloneqq S_{\mathcal{F}}(\boldsymbol{\theta}^+) - S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}})$. For all sufficiently small $\eta > 0$, this one-step acquisition gain satisfies:*

$$\Delta S_{\mathcal{F}} = -\eta \left\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}}), \mathbf{g}_a \right\rangle + O(\eta^2) \le \eta L_{\mathcal{F}} \left\| \Pi_{\mathcal{S}} \mathbf{g}_a \right\|_2 + O(\eta^2).$$

Proof. Fix $\mathcal{D}_a$ and abbreviate

$$\boldsymbol{\theta}_0 \coloneqq \boldsymbol{\theta}_{\mathrm{rel}}, \qquad \mathbf{g} \coloneqq \mathbf{g}_a.$$

Since $\mathbf{g}$ is evaluated at $\boldsymbol{\theta}_0$ for the fixed dataset $\mathcal{D}_a$, it is independent of the step size η. Let $\mathcal{N}$ be a neighborhood of $\boldsymbol{\theta}_0$ on which $S_{\mathcal{F}}$ is both $L_{\mathcal{F}}$-Lipschitz and $\beta_{\mathcal{F}}$-smooth. Because $\mathbf{g}$ is fixed, there exists $\eta_0 > 0$ such that, for every $\eta \in (0, \eta_0]$, the entire line segment

$$\{\boldsymbol{\theta}_0 - t\eta\mathbf{g} : t \in [0, 1]\}$$

is contained in $\mathcal{N}$.

Applying the fundamental theorem of calculus along this segment gives

$$\begin{aligned}
\Delta S_{\mathcal{F}} &= S_{\mathcal{F}}(\boldsymbol{\theta}_0 - \eta\mathbf{g}) - S_{\mathcal{F}}(\boldsymbol{\theta}_0) \\
&= -\eta \int_0^1 \left\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0 - t\eta\mathbf{g}), \mathbf{g} \right\rangle \, \mathrm{d}t \\
&= -\eta \left\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \mathbf{g} \right\rangle + R_\eta,
\end{aligned}$$

where

$$R_\eta \coloneqq -\eta \int_0^1 \left\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0 - t\eta\mathbf{g}) - \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \mathbf{g} \right\rangle \, \mathrm{d}t.$$

By the Cauchy–Schwarz inequality and the $\beta_{\mathcal{F}}$-smoothness of $S_{\mathcal{F}}$,

$$\begin{aligned}
|R_\eta| &\le \eta \int_0^1 \left\| \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0 - t\eta\mathbf{g}) - \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0) \right\|_2 \|\mathbf{g}\|_2 \, \mathrm{d}t \\
&\le \eta \int_0^1 \beta_{\mathcal{F}} \left\| t\eta\mathbf{g} \right\|_2 \|\mathbf{g}\|_2 \, \mathrm{d}t \\
&= \frac{\beta_{\mathcal{F}}}{2} \eta^2 \|\mathbf{g}\|_2^2.
\end{aligned}$$

Since $\mathbf{g} = \mathbf{g}_a$ is fixed with respect to η, the preceding estimate implies that $R_\eta = O(\eta^2)$. Therefore,

$$\Delta S_{\mathcal{F}} = -\eta \left\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{rel}}), \mathbf{g}_a \right\rangle + O(\eta^2),$$

which proves the first-order expansion.

We next bound its linear term. Since $S_{\mathcal{F}}$ is differentiable and locally $L_{\mathcal{F}}$-Lipschitz at $\boldsymbol{\theta}_0$, for every unit vector $\mathbf{v}$ and all sufficiently small nonzero h,

$$\frac{|S_{\mathcal{F}}(\boldsymbol{\theta}_0 + h\mathbf{v}) - S_{\mathcal{F}}(\boldsymbol{\theta}_0)|}{|h|} \le L_{\mathcal{F}}.$$

Taking $h \to 0$ and using differentiability yields

$$|\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \mathbf{v} \rangle| \le L_{\mathcal{F}}.$$

Taking the supremum over all $\|\mathbf{v}\|_2 = 1$ gives

$$\|\nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0)\|_2 \le L_{\mathcal{F}}.$$

By the local-separation condition,

$$\Pi_{\mathcal{S}} \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0) = \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0).$$

Moreover, because $\Pi_{\mathcal{S}}$ is an orthogonal projector, it is self-adjoint. Hence,

$$\begin{aligned} \langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \mathbf{g} \rangle &= \langle \Pi_{\mathcal{S}} \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \mathbf{g} \rangle \\ &= \langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \Pi_{\mathcal{S}} \mathbf{g} \rangle . \end{aligned}$$

It follows from the Cauchy–Schwarz inequality that

$$\begin{aligned} -\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \mathbf{g} \rangle &= -\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \Pi_{\mathcal{S}} \mathbf{g} \rangle \\ &\le |\langle \nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0), \Pi_{\mathcal{S}} \mathbf{g} \rangle| \\ &\le \|\nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}_0)\|_2 \, \|\Pi_{\mathcal{S}} \mathbf{g}\|_2 \\ &\le L_{\mathcal{F}} \, \|\Pi_{\mathcal{S}} \mathbf{g}\|_2 . \end{aligned}$$

Combining this inequality with the explicit remainder estimate gives

$$\Delta S_{\mathcal{F}} \le \eta L_{\mathcal{F}} \, \|\Pi_{\mathcal{S}} \mathbf{g}_a\|_2 + \frac{\beta_{\mathcal{F}}}{2} \eta^2 \|\mathbf{g}_a\|_2^2.$$

For fixed $\mathcal{D}_a$, the second term is $O(\eta^2)$, and therefore

$$\Delta S_{\mathcal{F}} \le \eta L_{\mathcal{F}} \, \|\Pi_{\mathcal{S}} \mathbf{g}_a\|_2 + O(\eta^2),$$

which completes the proof. □

C.2 A Simple ReLU Counterexample

We give a minimal construction where a zero release score coexists with a nonzero acquisition gradient, while activation suppression reduces that gradient.

Let $\mathrm{ReLU}(z) = \max\{z, 0\}$ act element-wise and consider

$$\mathbf{h}_u(\mathbf{x}) = \mathrm{ReLU}(\mathbf{A}_u \mathbf{x}), \qquad f_{\boldsymbol{\theta}}(\mathbf{x}) = \mathbf{w}^\top \mathbf{h}_u(\mathbf{x}), \qquad \mathbf{A}_u = \begin{pmatrix} u & -1 \\ -1 & 1 \end{pmatrix},$$

where $\mathbf{w} = (w_f, w_r)^\top$ and $\boldsymbol{\theta} = (w_f, u, w_r)^\top$. As illustrated in Fig. 5a, the first hidden unit forms the forbidden-sensitive pathway, parameterized by (w_f, u), while the second forms an independent retain pathway controlled by w_r. The fixed entries -1 isolate these pathways on the two basis inputs below and keep inactive pre-activations away from the ReLU kink. Accordingly, let

$$\mathcal{S} = \{(\alpha, \beta, 0)^\top : \alpha, \beta \in \mathbb{R}\}.$$

For the forbidden and retain inputs $\mathbf{x}_f = \mathbf{e}_1$ and $\mathbf{x}_r = \mathbf{e}_2$, respectively, any $u > 0$ gives

$$\mathbf{h}_u(\mathbf{x}_f) = (u, 0)^\top, \qquad \mathbf{h}_u(\mathbf{x}_r) = (0, 1)^\top,$$

and therefore

$$S_{\mathcal{F}}(\boldsymbol{\theta}) \coloneqq f_{\boldsymbol{\theta}}(\mathbf{x}_f) = w_f u, \qquad f_{\boldsymbol{\theta}}(\mathbf{x}_r) = w_r.$$

The two rows of Fig. 5b trace output suppression and additional activation suppression from their released states to their one-step gains:

$$\boldsymbol{\theta}_{\mathrm{out}} = (0, 1, 1)^\top, \qquad \boldsymbol{\theta}_{\mathrm{act}} = (0, \delta, 1)^\top, \qquad 0 < \delta < 1.$$

Both released states satisfy

$$S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{out}}) = S_{\mathcal{F}}(\boldsymbol{\theta}_{\mathrm{act}}) = 0, \qquad f_{\boldsymbol{\theta}_{\mathrm{out}}}(\mathbf{x}_r) = f_{\boldsymbol{\theta}_{\mathrm{act}}}(\mathbf{x}_r) = 1,$$

although their forbidden-sensitive activations are 1 and δ, respectively.

For this illustrative regression model, the attacker minimizes squared loss, equivalent up to an additive constant to the NLL of a unit-variance Gaussian with mean $f_{\boldsymbol{\theta}}(\mathbf{x}_f)$:

$$\mathcal{L}_a(\boldsymbol{\theta}) = \frac{1}{2}\left(f_{\boldsymbol{\theta}}(\mathbf{x}_f) - 1\right)^2 = \frac{1}{2}(w_f u - 1)^2.$$

For $u > 0$, its gradient is

$$\mathbf{g}_a(\boldsymbol{\theta}) = (w_f u - 1)\begin{pmatrix} u \\ w_f \\ 0 \end{pmatrix}.$$

Consequently,

$$\mathbf{g}_a^{\text{out}} = (-1, 0, 0)^\top, \qquad \mathbf{g}_a^{\text{act}} = (-\delta, 0, 0)^\top.$$

Both gradients lie in $\mathcal{S}$, yielding the projected norms shown in the third column of Fig. 5b:

$$\left\|\Pi_{\mathcal{S}}\mathbf{g}_a^{\text{out}}\right\|_2 = 1, \qquad \left\|\Pi_{\mathcal{S}}\mathbf{g}_a^{\text{act}}\right\|_2 = \delta.$$

After one gradient step with the same step size η,

$$\boldsymbol{\theta}^+_{\text{out}} = (\eta, 1, 1)^\top, \qquad \boldsymbol{\theta}^+_{\text{act}} = (\eta\delta, \delta, 1)^\top,$$

giving the exact acquisition gains shown in the final column of Fig. 5b:

$$\Delta S^{\text{out}}_{\mathcal{F}} = \eta, \qquad \Delta S^{\text{act}}_{\mathcal{F}} = \eta\delta^2.$$

The first factor of δ comes from the smaller attacker gradient and the second from the remaining activation; both released models preserve a retain output of one.

Finally, for $u > 0$,

$$\nabla_{\boldsymbol{\theta}} S_{\mathcal{F}}(\boldsymbol{\theta}) = (u, w_f, 0)^\top \in \mathcal{S},$$

so the local-separation condition holds. Both released states lie strictly away from the ReLU kink; in sufficiently small neighborhoods around them, $S_{\mathcal{F}}(\boldsymbol{\theta}) = w_f u$ is locally Lipschitz and 1-smooth. Thus, this regression construction satisfies the local regularity and separation conditions of Prop. 3.1, while showing that identical release-time scores can conceal different acquisition receptivity. The choice $0 < \delta < 1$ gives a continuous δ^2 separation while remaining away from the ReLU kink; moving the forbidden gate strictly below zero would additionally set its ReLU derivative to zero, matching the low-slope sealing mechanism that GSU applies during defense optimization.

D PSEUDOCODE

Algorithm 1: Gradient-Sealed Unlearning (GSU)

Input: Original parameters $\boldsymbol{\theta}_o$; proxy and retain sets $\mathcal{D}_f, \mathcal{D}_r$; look-ahead steps K_{la}; selection parameters K, p; threshold $\tau < 0$; loss weights λ_s, λ_r.
Output: Released parameters $\boldsymbol{\theta}_{\text{rel}}$.

`// Expose`
Fit a disposable copy of $\boldsymbol{\theta}_o$ on $\mathcal{D}_f$ for K_{la} steps using Eq. (1), obtaining $\boldsymbol{\theta}_{\text{la}}$;
Collect both models' pre-activations under matched teacher-forcing contexts $(\mathbf{x}, \mathbf{y}^{<i})$;
`// Localize`
Compute $r_{\ell j}$, m_ℓ, and R_ℓ using Eqs. (2) and (3);
Select up to K layers with the highest positive R_ℓ among those with $m_\ell > 0$, forming $\mathcal{I}^\star$;
Within each selected layer, select up to $\lceil pd_\ell \rceil$ channels with the highest positive $r_{\ell j}$, forming $\mathcal{N}^\star_\ell$;
$\mathcal{G} \leftarrow \{(\ell, j) : \ell \in \mathcal{I}^\star,\ j \in \mathcal{N}^\star_\ell\}$;
`// Seal`
Fix $\mathcal{G}$, discard $\boldsymbol{\theta}_{\text{la}}$, and reset $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}_o$;
for *each defense update* **do**
    Sample minibatches $\mathcal{B}_f \subseteq \mathcal{D}_f$ and $\mathcal{B}_r \subseteq \mathcal{D}_r$;
    Compute the joint loss in Eq. (5) on $(\mathcal{B}_f, \mathcal{B}_r)$, with $\mathcal{L}_{\text{seal}} = 0$ if $\mathcal{G} = \varnothing$;
    Update all parameters $\boldsymbol{\theta}$ with a first-order optimizer step;
return $\boldsymbol{\theta}_{\text{rel}} \leftarrow \boldsymbol{\theta}$;

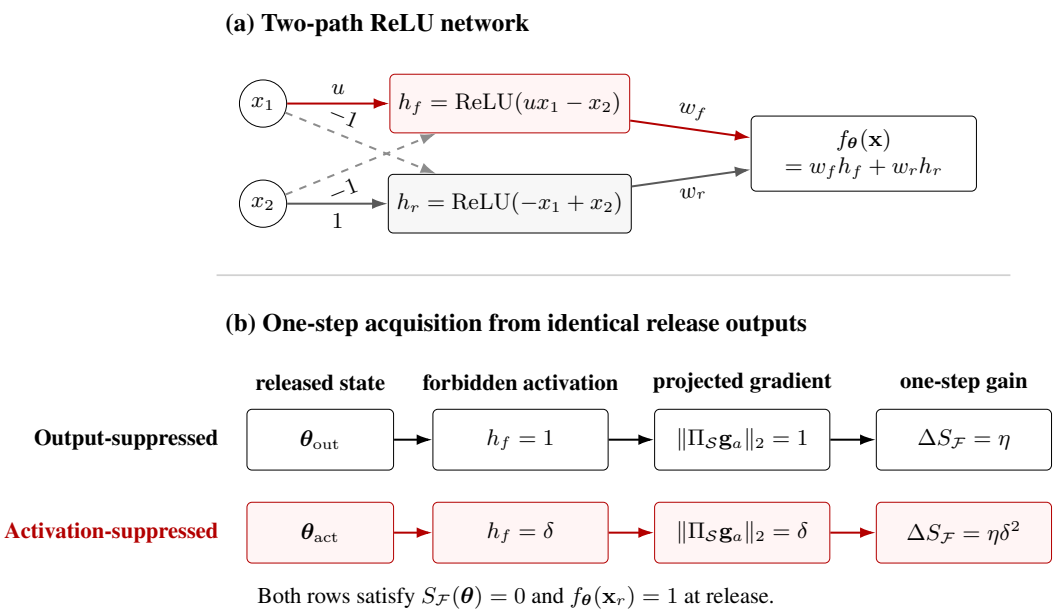


Figure 5: **A minimal ReLU counterexample.** Panel (a) shows the forbidden-sensitive path in red and the retain path in gray; dashed arrows are fixed inhibitory connections. For $\mathbf{x}_f = \mathbf{e}_1$ and $\mathbf{x}_r = \mathbf{e}_2$, both released states in (b) have identical forbidden and retain outputs. Reducing the forbidden activation from 1 to δ reduces the projected attacker-gradient norm by δ and the exact one-step acquisition gain by δ^2.

E Further Experimental Setup

E.1 Mechanism Study for Figure 2

Reference, splits, and metrics. This mechanism study uses the official `Qwen3.5-2B` snapshot and a retain-only Reference trained for five epochs on all 3,600 `retain90` examples (AdamW, peak learning rate 10^{-5}, seed 0). The Reference receives no optimizer exposure to $\mathcal{D}_f$, $\mathcal{D}_a$, or holdout. Within each of the 20 target authors, a fixed $4/8/8$ split gives 80 proxy, 160 attacker, and 160 holdout facts; the splits have no exact row overlap but share authors and domain structure. Forbidden capability is evaluated only on $\mathcal{D}_a$ using ES.

Panel (a): SimNPO, SatImp, NPO, and RMU start from the same Reference and run for 10 release epochs (50 optimizer steps, learning rate 10^{-5}, effective batch size 16, AdamW, weight decay 0.01, seed 0) using raw checkpoints without UWC or interpolation. The acquisition learning rate is calibrated once on the Reference and then applied unchanged to every plotted release state: 10 epochs/100 steps, constant learning rate 4×10^{-6}, effective batch size 16, and seed 0, with no early stopping or best-epoch selection.

Panels (b,c): For Panel (b), a disposable Reference copy receives one epoch (10 steps) of $\mathcal{D}_f$ supervised look-ahead at learning rate 5×10^{-6} and effective batch size 8. We record answer-token mean `gate_proj` pre-activations before and after look-ahead, discard the copy, reload the unchanged Reference, and independently acquire on $\mathcal{D}_a$ for 10 epochs at learning rate 10^{-6} and effective batch size 16. Both sensitivity arrays are converted to within-layer ranks before pooling. Panel (c) uses the same $\mathcal{D}_a$ schedule (100 steps, seed 0, weight decay 0) and nested layer-quota masks at $0.5/1/2/5/10\%$ coverage. For each selected channel, attenuation scales the corresponding `gate_proj`/`up_proj` rows, `down_proj` column, and post-Adam update slice by $1 - a$; every attenuated arm is matched step-wise to Full-FT in whole-model update norm. The grid comprises one Full-FT control and 20 localized arms; the 0% column repeats that control. Let $G_{p,a}$ be the 10-epoch increase in F for coverage p and attenuation a. Each cell reports $R_{p,a} = 100(G_{\mathrm{FT}} - G_{p,a})/G_{\mathrm{FT}}$, the percentage reduction in acquisition gain relative to Full-FT.

E.2 Benchmark Construction

TOFU. We use the official `forget10/retain90` partition: 20 forbidden authors with 400 QA pairs and 180 retain authors with 3,600 QA pairs. A fixed seed-0 ordering partitions each forbidden author's 20 QA pairs equally between B_1 and B_2, yielding 200 examples per pool. In the disjoint setting, $(\mathcal{D}_f, \mathcal{D}_a) = (B_1, B_2)$; in the identical setting, $(\mathcal{D}_f, \mathcal{D}_a) = (B_2, B_2)$. Thus, attack training and ES evaluation use the same B_2 across settings, while disjoint defense uses non-overlapping QA records about the same authors. The disjoint split does not establish unseen-author or semantically disjoint-fact generalization. All 3,600 retain examples form $\mathcal{D}_r$.

WMDP. We split each domain's forget corpus in source order: the first $\lfloor N/2 \rfloor$ documents form the defender pool B_1, and the remainder form the attacker pool B_2. The Bio pools contain 12,226 and 12,227 documents; the Cyber pools contain 500 documents each. We evaluate only disjoint attacks, with $(\mathcal{D}_f, \mathcal{D}_a) = (B_1, B_2)$. Since the target knowledge is already present, identical defense and attack data would reduce this setting to conventional robust unlearning. Defense uses the full corresponding retain corpus, containing 60,887 Bio or 4,473 Cyber documents. Each pool is independently concatenated and tokenized into 512-token sequences using the `OpenUnlearning` preprocessing pipeline. The official Bio and Cyber test sets contain 1,273 and 1,987 multiple-choice questions, respectively; utility is evaluated on all 14,042 `MMLU` test questions across 57 subjects.

E.3 Evaluation Metrics

Extraction strength. For an answer of L tokens, extraction strength (ES) is the length of its longest suffix whose tokens are all correctly predicted by teacher-forced argmax decoding, divided by L. We average this score over the 200 attacker examples. Let ES_t denote the score after attack epoch t, with $t = 0$ denoting release. The main `TOFU` table reports

$$\overline{\mathsf{ES}}_3 = \frac{1}{3}\sum_{t=1}^{3} \mathsf{ES}_t, \qquad \mathsf{ES}_5,$$

which summarize early acquisition and extraction after five attack epochs.

WMDP accuracy. We evaluate domain accuracy and sample-weighted `MMLU` accuracy at attack steps $s \in \{25, 50, 75, 100, 125, 150\}$. Writing F_j for domain accuracy at step $25j$, we report

$$\overline{\mathsf{F}}_3 = \frac{1}{3}\sum_{j=1}^{3} \mathsf{F}_j, \qquad \mathsf{F}_5.$$

Thus, the early mean uses steps 25, 50, and 75, while the fifth-checkpoint score uses step 125.

Utility and release calibration. `TOFU` utility is $\mathrm{HM}(\mathsf{MU}, \mathsf{Fluency})$, where MU aggregates answer probability, ROUGE-L recall, and truth ratio over the retain, real-author, and world-fact evaluation sets. These sets contain 400, 100, and 117 examples, respectively; fluency uses the 200 canonical B_2 questions. `WMDP` utility is sample-weighted `MMLU` accuracy (Hendrycks et al., 2021). The reference utility U_{ref} is measured on the shared retain-only base for `TOFU` and the original checkpoint for `WMDP`. For the main comparisons, Unlearning with Control (UWC) (Wang et al., 2025b) calibrates each final defense checkpoint against its reference checkpoint to satisfy $U_{\mathrm{rel}} \geq 0.95U_{\mathrm{ref}}$ before attack. Release utility is therefore omitted from the main tables. The separate mechanism and further-analysis protocols specify their own checkpoint handling in §E.1 and §F.

Usable acquisition. For both benchmarks, we report

$$\mathsf{UA}_{90} = \max_{\substack{t \in \mathcal{T} \\ U_t \geq 0.9U_{\mathrm{ref}}}} C_t,$$

where C_t is the forbidden score and U_t is utility at the same checkpoint. For `TOFU`, $C_t = \mathsf{ES}_t$ and $\mathcal{T} = \{0, 1, \ldots, 10\}$ indexes attack epochs, including release. For `WMDP`, C_t is domain accuracy and $\mathcal{T} = \{25, 50, 75, 100, 125, 150\}$ indexes attack steps. The 90% threshold determines attack-checkpoint eligibility and is distinct from the 95% release-calibration requirement. The reference utility remains fixed throughout the attack, and attack checkpoints are not recalibrated. All main-table scores are percentages, with lower values indicating stronger protection.

E.4 Models and Initialization

The TOFU checkpoints are `Llama-3.2-1B-Instruct`, `Llama-3.2-3B-Instruct`, `Llama-3.1-8B-Instruct`, and `Qwen3.5-2B/4B/9B`. For each model, we construct one retain-only base using five epochs on `retain90` (1,125 updates), AdamW, learning rate 10^{-5}, batch size 16, weight decay 0.01, and a one-epoch warmup followed by linear decay. This base is shared across methods and both settings. WMDP defenses share the original `zephyr-7b-beta` checkpoint.

E.5 Common Defense and Attack Configuration

The following settings describe the main comparisons unless otherwise specified. All runs use seed 0, bfloat16, a maximum sequence length of 512, and effective batch size 16, with gradient accumulation as needed. TOFU defense and attack each use ten epochs over 200 examples, giving 130 optimizer updates with the final partial batch retained. Defense uses a base learning rate of 10^{-5} and draws paired retain batches from $\mathcal{D}_r$. Both stages use AdamW with weight decay 0.01, a constant learning rate, and no warmup. The fixed attack learning rates for `Llama-3.2-1B/3B` and `Llama-3.1-8B` are $(10^{-5}, 10^{-5}, 5\times10^{-6})$, and those for `Qwen3.5-2B/4B/9B` are $(10^{-6}, 5\times10^{-6}, 10^{-5})$. We evaluate the release checkpoint and every attack epoch.

WMDP uses paged 32-bit AdamW with zero weight decay. Defense runs for 80 updates at learning rate 4×10^{-6}, with 16 warmup updates followed by a constant rate; the final defense checkpoint is UWC-calibrated before release. Attack runs for 150 updates on the attacker corpus at constant learning rate 4×10^{-6} without warmup, with evaluation every 25 updates.

F Additional Results

Shared protocol. Fig. 4 and the analyses below use disjoint TOFU, with $\mathcal{D}_f = B_1$ and $\mathcal{D}_a = B_2$. Each pool contains 200 non-overlapping QA pairs from the same 20 forbidden authors; the retain set contains 3,600 QA pairs from 180 other authors. This record-disjoint split does not establish unseen-author or semantically disjoint-fact generalization. Attack and gate analyses use `Qwen3.5-2B`; benign adaptation also includes `Llama-3.2-1B/3B`. §F.1 and §F.2 provide two additional control studies. The remaining three subsections supplement the main analyses with accuracy, loss, utility, and gate-distribution measurements from the same runs, rather than independent experimental replications.

Frozen configurations. For each model, GSU uses the configuration with the lowest mean ES over attack epochs 1–5 among the completed disjoint candidates when the analysis plan was frozen. Configurations are not retuned for individual figures, and baselines use the analysis plan's default settings. Selection uses existing B_2 attack evaluations, so these are development-stage diagnostics rather than an independent-test or equal-tuning-budget comparison. Tab. 4 lists the frozen configurations. All three use a defense learning rate of 10^{-5}, 100 look-ahead updates, and a recorded release mixing coefficient of 1. Some checkpoints were reconstructed from these configurations and are not assumed to be byte-identical to historical checkpoints. All results are single-run observations without multi-seed uncertainty estimates or significance tests.

Table 4: **Frozen configurations for the further analyses.** "Seal alpha" retains the terminology of the analysis records.

Model	K	p	Seal alpha	τ
`Llama-3.2-1B`	4	5%	0.01	-9
`Qwen3.5-2B`	4	5%	0.03	-4
`Llama-3.2-3B`	2	5%	0.01	-4

F.1 Resistance after Benign Fine-Tuning

Setup. We test whether GSU's lower acquisition survives an intervening benign update on `Qwen3.5-2B`. The four conditions cross No Defense/GSU with direct attack/benign fine-tuning followed by attack. Benign fine-tuning uses two epochs on the economics task, learning rate 10^{-5},

and effective batch size 16, following §F.3. All four attacks then use full-parameter training on B_2 for ten epochs at learning rate 10^{-6} and effective batch size 16. Attack epoch 0 is the checkpoint immediately before the attack, after benign adaptation where applicable. Fig. 6 shows all eleven observed checkpoints for both ES and utility; the direct-attack curves reuse the corresponding controls from Fig. 4.

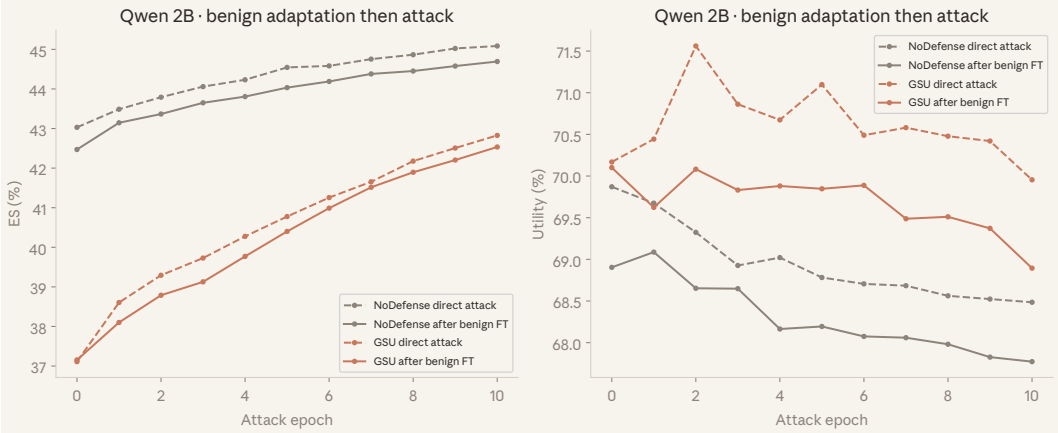


Figure 6: **Attacks after benign adaptation.** ES (left) and utility (right) on `Qwen3.5-2B` under disjoint `TOFU`. Dashed lines denote direct attacks; solid lines denote attacks following two epochs of benign fine-tuning. GSU retains lower extraction than the adapted undefended control, while utility reveals the cost relative to direct attack.

Results. After benign adaptation and ten attack epochs, GSU reaches 42.534% ES, compared with 44.693% for No Defense, a gap of about 2.16 percentage points; their utilities are 68.894% and 67.772%, respectively. Thus, GSU retains lower acquisition than the matched control after this benign update. Compared with direct attack on GSU, however, benign adaptation slightly lowers both final ES (42.826% to 42.534%) and utility (69.955% to 68.894%). It therefore does not uniformly improve protection and utility. This result covers the specified benign task and budgets, not arbitrary post-release fine-tuning.

F.2 Paired Sealing Ablation

Setup. We compare full GSU with the variant omitting the sealing term on `Qwen3.5-2B` under disjoint `TOFU`. Both conditions complete defense, release evaluation, and the same ten-epoch attack protocol used in §F.1. Fig. 7 reports ES and utility at all eleven checkpoints. This is a paired sealing ablation, not a full loss-component or localization-control suite. The full-GSU trajectory is the same run used in the other attack analyses.

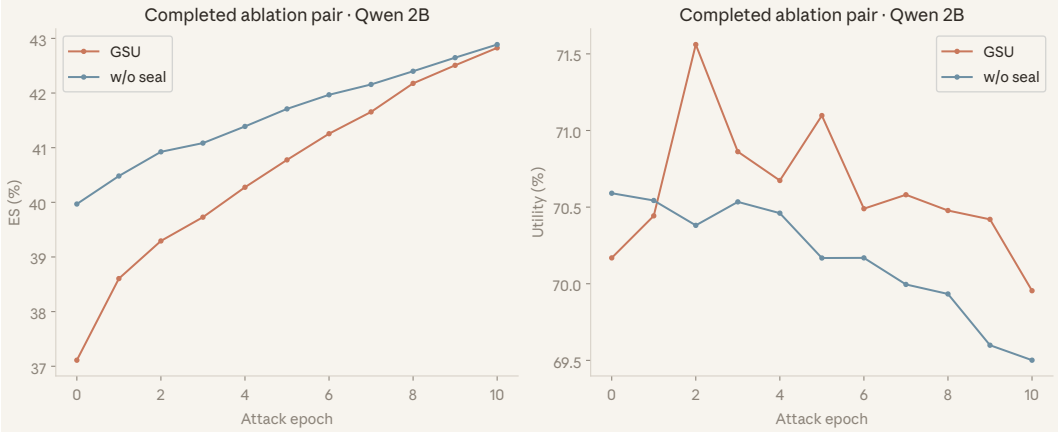


Figure 7: **Contribution of the sealing term.** Full GSU versus the variant without sealing on `Qwen3.5-2B` under disjoint `TOFU`. The ES gap is largest near release and narrows over longer attacks; utility is reported alongside extraction. This comparison does not include other loss or gate-selection ablations.

Results. Full GSU has lower ES at release (37.111% versus 39.969%) and attack epoch 5 (40.776% versus 41.710%). By epoch 10, the values are 42.826% and 42.887%, leaving a gap of only 0.061 percentage points. Final utility is 69.955% for GSU and 69.501% without sealing. This comparison supports an incremental contribution of sealing to lower early-budget extraction, but the advantage narrows substantially under longer attacks. Because the initial scores also differ, the absolute trajectories do not establish a slower acquisition rate or permanent resistance.

F.3 Benign Adaptation: Accuracy and Validation Loss

Setup and metrics. We extend Fig. 4c with two complementary measurements from the same six benign fine-tuning runs: No Defense and GSU on `Llama-3.2-1B`, `Qwen3.5-2B`, and `Llama-3.2-3B`. Fine-tuning uses two epochs of full-parameter AdamW, learning rate 10^{-5}, effective batch size 16, weight decay 0.01, bfloat16, maximum sequence length 512, a constant learning-rate schedule, and seed 0. The learning rate is used directly without calibration, and no defense hooks or sealing penalties are active. Each model's paired runs use the same 128-example validation set; token counts may differ across tokenizers. We evaluate at epochs 0, 1, and 2. Economics accuracy is the equally weighted mean of the `MMLU` subjects `high_school_macroeconomics` and `high_school_microeconomics`, rather than validation NLL or ES.

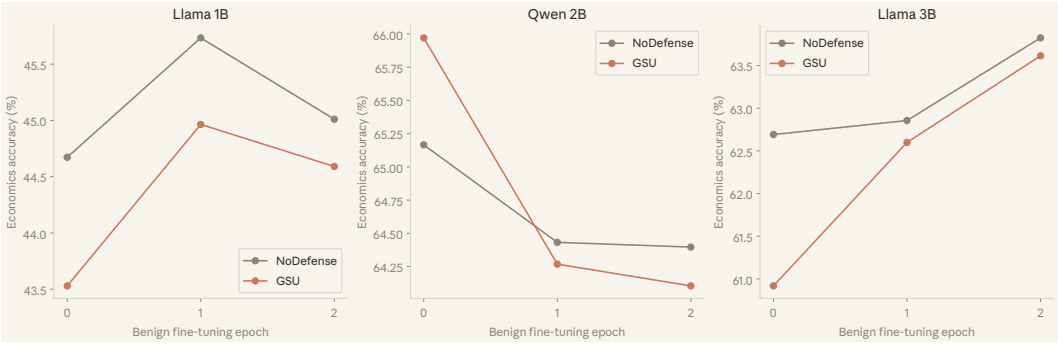


Figure 8: **Economics accuracy during benign fine-tuning.** Accuracy averages two `MMLU` economics subjects with equal weight. The two Llama-based GSU models improve from their initial accuracy, whereas `Qwen3.5-2B` declines under both initializations. These are supplementary behavioral measurements of the same runs used for the benign-loss analysis, not independent replications.

Accuracy. Fig. 8 reveals model-dependent outcomes. GSU's accuracy increases from 43.53% to 44.59% on `Llama-3.2-1B` and from 60.92% to 63.61% on `Llama-3.2-3B`, narrowing the gaps to

the paired No Defense endpoints of 45.01% and 63.82%. On `Qwen3.5-2B`, however, accuracy falls from 65.97% to 64.10% for GSU and from 65.17% to 64.40% for No Defense. The benign-loss improvement therefore does not imply an accuracy gain across all three models.

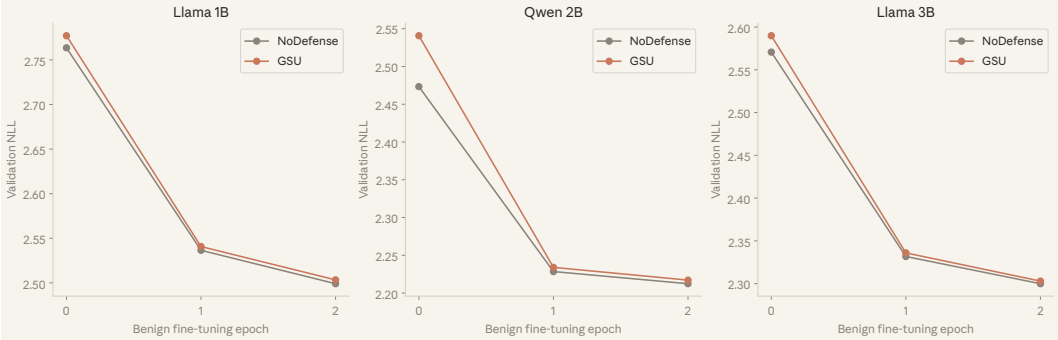


Figure 9: **Full benign validation-loss trajectories.** The three observed epochs expand the endpoint reductions in Fig. 4c. GSU approaches the corresponding No Defense loss in all three models, while starting and finishing at slightly higher losses.

Validation loss. Fig. 9 displays the initial losses as well as both post-training checkpoints. Tab. 5 retains the endpoint summary, where $\Delta\text{NLL} = \text{NLL}_0 - \text{NLL}_2$ is an absolute loss reduction, not a percentage. All three GSU models reduce validation NLL and finish within 0.005 of their paired controls. GSU's larger reductions occur from higher initial losses and do not establish improved learning efficiency, lower final loss, or statistical equivalence. Together with the mixed accuracy outcomes, these results support retained optimization ability on this benign task without implying uniform downstream performance gains.

Table 5: **Validation NLL before and after benign fine-tuning.** ΔNLL is the absolute decrease after two epochs; displayed values are rounded independently.

Model	**Initialization**	NLL_0	NLL_2	ΔNLL
`Llama-3.2-1B`	No Defense	2.7635	2.4991	0.2645
	GSU	2.7771	2.5033	0.2738
`Qwen3.5-2B`	No Defense	2.4733	2.2123	0.2610
	GSU	2.5408	2.2171	0.3237
`Llama-3.2-3B`	No Defense	2.5707	2.2999	0.2707
	GSU	2.5899	2.3030	0.2869

F.4 Utility during Acquisition

Setup. Fig. 10 adds utility to the same five-method, `Qwen3.5-2B` disjoint trajectories shown in Fig. 4a. All attacks use ten epochs of full-parameter training on B_2, learning rate 10^{-6}, and effective batch size 16. Epoch 0 denotes release; all eleven measured checkpoints are connected without smoothing or missing-point interpolation. Both ES and utility use the recorded scores in percentage units, following §E.3. No additional utility threshold is imposed, and this diagnostic does not calculate UA_{90}.

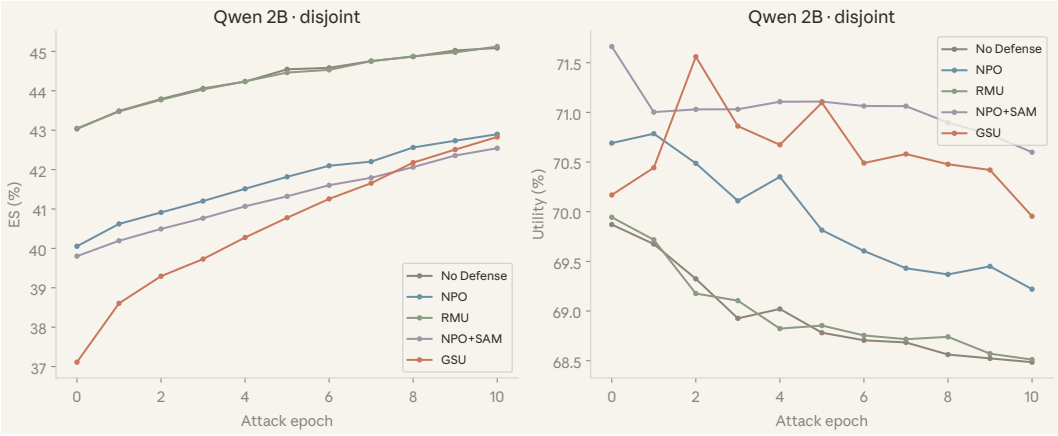


Figure 10: **Utility alongside the acquisition trajectory.** The left panel repeats Fig. 4a; the right adds utility for the same checkpoints. GSU retains higher utility than No Defense and RMU, while NPO+SAM also maintains high utility and attains lower late-attack ES. No utility-constrained maximum is computed in this figure.

Utility and extraction trade-offs. GSU maintains higher utility than No Defense and RMU throughout this trajectory. At epoch 10, its utility is 69.955%, versus 68.485% for No Defense, 68.511% for RMU, and 69.220% for NPO. NPO+SAM finishes at 70.599% utility and 42.542% ES, compared with GSU's 42.826% ES. Thus, lower early-budget extraction does not imply that GSU dominates all baselines in both metrics over the entire attack. Tab. 6 supplements the curves with the main-table early mean $\overline{\text{ES}}_3$, fixed-epoch scores, and $\Delta\text{ES} = \text{ES}_{10} - \text{ES}_0$. In particular, GSU's 5.715-point increase exceeds NPO+SAM's 2.740-point increase, reinforcing the distinction between a lower starting level and a slower acquisition rate.

Table 6: **Attack-trajectory summaries with final utility.** Scores are percentages; ΔES is an increase in percentage points and U_{10} is utility at attack epoch 10.

Method	ES_0	$\overline{\text{ES}}_3$	ES_5	ES_{10}	ΔES	U_{10}
No Defense	43.030	43.779	44.544	45.085	2.055	68.485
NPO	40.053	40.911	41.816	42.896	2.843	69.220
RMU	43.045	43.759	44.462	45.122	2.078	68.511
NPO+SAM	39.802	40.484	41.321	42.542	2.740	70.599
GSU	37.111	39.209	40.776	42.826	5.715	69.955

F.5 Release-Stage Gate Statistics

Scope and aggregation. We supplement the mean derivative in Fig. 4b with selected-gate pre-activation distributions and negative-tail occupancy. The original control, released GSU, and the released variant without sealing are measured on all 200 B_1 examples, all 200 B_2 examples, and the 3,600 retain examples. Each group contains the same 1,232 recorded layer–channel entries. "Original" denotes the control checkpoint entering this analysis, not necessarily the original pretrained weights. All measurements are at release, before attacker updates.

For each gate, we average the answer-token statistic within each example and then average examples equally. Writing $\mathcal{G}_{\text{rec}}$ for the recorded gate set, the mean absolute derivative on an input set S is

$$D_S(\boldsymbol{\theta}) = \frac{1}{|\mathcal{G}_{\text{rec}}|} \sum_{g \in \mathcal{G}_{\text{rec}}} \frac{1}{|S|} \sum_{(\mathbf{x},\mathbf{y}) \in S} \frac{1}{|\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} \left| \phi'\left(u_g^t(\boldsymbol{\theta})\right) \right|, \tag{6}$$

$$\phi'(u) = \sigma(u) + u\sigma(u)\big(1 - \sigma(u)\big).$$

Here u_g^t is evaluated at the corresponding answer-token prediction position. Gate-wise mean pre-activation uses the same within-example and across-example averaging, with u_g^t in place of $|\phi'(u_g^t)|$. Its distribution is therefore over gate means, not individual token activations.

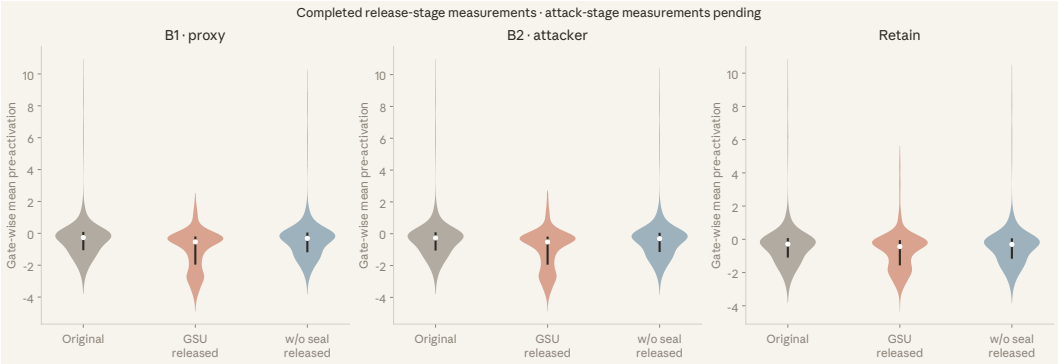


Figure 11: **Distributions of selected-gate mean pre-activations at release.** Each statistic averages answer-token pre-activations within examples and then weights examples equally. White markers and dark segments denote the median and interquartile range across the 1,232 recorded gates; violin shapes are density estimates, not confidence intervals. No post-attack measurements are included.

Pre-activation distributions. Fig. 11 shows that GSU shifts the selected-gate distributions toward more negative values. The median changes from -0.267 to -0.537 on B_1 and from -0.266 to -0.516 on B_2, compared with -0.288 to -0.436 on retain inputs. The corresponding GSU interquartile ranges are $[-1.964, -0.196]$, $[-1.944, -0.180]$, and $[-1.569, -0.046]$. Without sealing, the medians remain closer to the original control at -0.315, -0.309, and -0.305. These distributions complement the mean-response measurements by showing variation across the selected gates.

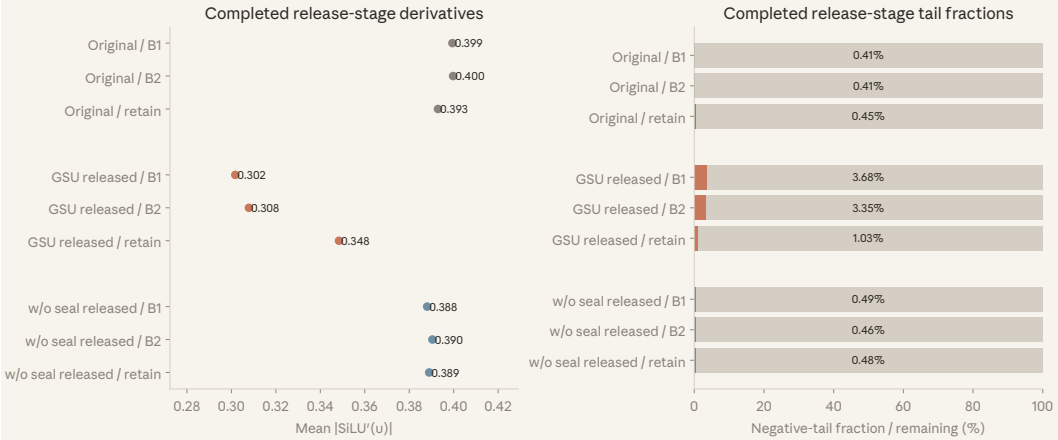


Figure 12: **Release-stage derivatives and negative-tail occupancy.** Left: mean absolute SiLU derivative, averaged equally over recorded gates. Right: the colored segment and its numerical label give mean negative-tail occupancy at the recorded threshold $\tau = -4$; gray is the remaining proportion. These local statistics do not measure parameter-gradient norms or persistence under subsequent attacks.

Local response and negative-tail occupancy. Fig. 12 and Tab. 7 combine the derivative means with the equally weighted mean of each gate's recorded negative-tail fraction. The tail threshold is $\tau = -4$, not simply zero. GSU reduces the mean absolute derivative relative to the original control by 24.45% on B_1 and 22.98% on B_2, versus 11.33% on retain inputs. Tail occupancy increases from about 0.41% to 3.68% and 3.35% on B_1 and B_2, respectively, and from 0.45% to 1.03% on retain inputs. The variant without sealing stays closer to the original control. This pattern is consistent with lower local responses on target inputs, not complete gate shutdown: occupancy remains a small minority, and retain responses also change. Neither these release-stage measurements nor the paired behavioral ablation establish that full-network gradients are uniformly smaller or that gate suppression persists after long attacks.

Table 7: **Release-stage gate responses and tail occupancy.** Derivative means and tail fractions are equally weighted over recorded gates; tail occupancy is expressed as a percentage at $\tau = -4$.

	Mean $\lvert\phi'(u)\rvert$			Negative-tail occupancy (%)		
Inputs	Original	GSU	w/o seal	Original	GSU	w/o seal
B_1	0.3994	0.3018	0.3881	0.411	3.684	0.487
B_2	0.3997	0.3079	0.3904	0.408	3.348	0.463
Retain	0.3929	0.3484	0.3890	0.451	1.029	0.477

G Limitation and Future Work

Our evaluation covers practical downstream fine-tuning rather than unrestricted retraining from scratch with arbitrary compute. The analysis is local, and GSU relies on the defender proxy set sharing acquisition pathways with future data; broader domain shifts may therefore require richer proxy coverage. Extending the study to additional architectures, domains, and post-release training procedures is a natural direction for future work.