Distance Is Not Enough: Forget-Retain Alignment Gap Predicts LLM Relearning Robustness

Yi Chen[1*], Hanna Hsieh[1*], Shuhong Liu[2], Chuanbo Hua[1], Zihan Ma[1], Kun Wang[1], Joo-Young Kim[1]

[1]KAIST
[2]The University of Tokyo
{chenyi, hihahanaisme, cbhua, zihanma, walkerwang, jooyoung1203}@kaist.ac.kr
s-liu@mi.t.u-tokyo.ac.jp

Abstract

Machine unlearning aims to make a model forget specific data, yet unlearned LLMs often fail to stay unlearned: brief fine-tuning can revive removed knowledge. Existing robustness predictors rely on global weight-space displacement, but distance alone can be misleading when random or destructive updates collapse performance. We argue that relearning robustness depends on update structure: robust unlearning should affect forget-critical weights while sparing retain-critical ones. We introduce the Forget-Retain Alignment Gap (FRAG), a training-free predictor that scores an update's forget-retain alignment without running a relearning attack, and separates selective from dense updates more reliably than global distance. Building on the forget-critical, retain-sparing principle, Forget-Retain Pruning (FRP) improves relearning robustness. Our results suggest that weight selectivity better explains robustness than distance alone. Our code is available at https://github.com/Yi1-Chen/FRAG.

1 Introduction

Machine unlearning aims to remove the influence of specified data from a trained model while preserving its behavior on the remaining retain data (Maini et al., 2024; Li et al., 2024). However, for large language models (LLMs), forgetting at edit time is often fragile: even without access to the forgotten examples, subsequent fine-tuning on benign retain data can revive the supposedly removed knowledge, forming a relearning attack (Hu et al., 2025; Lynch et al., 2024). This relearning behavior exposes a central weakness of current unlearning methods: successful forgetting at edit time does not necessarily imply robustness against relearning attacks (Łucki et al., 2025; Che et al., 2025; Deeb and Roger, 2024).

Recent work argues that relearning robustness can be improved by moving the unlearned model farther from the original model in weight space, making the removed knowledge harder to recover under relearning attacks (Siddiqui et al., 2025). This motivates using the global ℓ_2 weight-space distance between the original and unlearned weights as a simple robustness predictor. However, distance alone can be misleading: it measures how far the weights move, but not which weights move. For example, a random or destructive update can yield a large ℓ_2 displacement and appear robust to a distance-based predictor, while collapsing retain or forget performance and producing a model that is far from the original but not meaningfully unlearned. Robust unlearning should depend not only on displacement magnitude, but on whether the update concentrates on forget-critical weights while sparing retain-critical ones.

This observation motivates a proxy that diagnoses where the unlearning update is concentrated, not only how large it is. We introduce the Forget–Retain Alignment Gap (FRAG), a training-free scalar proxy for predicting relearning robustness that measures whether the update aligns more with forget-critical than retain-critical weights. By penalizing retain-side disruption, FRAG avoids rewarding collapsed models and better reflects practical unlearning robustness.

Across diverse unlearning methods (Zhang et al., 2024a; Maini et al., 2024; Li et al., 2024; Pochinkov and Schoots, 2024; Jang et al., 2023), benchmarks, and model families, we show that FRAG correlates with empirical relearning robustness more reliably than global distance-based predictors. To show that this principle is actionable rather than merely diagnostic, we instantiate it as Forget-Retain Pruning (FRP), which selectively targets forget-critical weights while avoiding retain-critical ones. FRP improves robustness under relearning attacks, tracing a robustness–

*Equal contribution.

utility frontier that dominates strong baselines at every matched utility level, suggesting that which weights move matters more than distance alone.

Our contributions are summarized as follows:

- We revisit relearning robustness from a weight-selectivity perspective, showing that global distance alone cannot distinguish selective unlearning updates from random or retain-damaging perturbations.
- We introduce FRAG, a training-free proxy that predicts relearning robustness by diagnosing whether an update is forget-critical and retain-sparing.
- As an application of the same principle, we propose FRP, which improves relearning robustness at a controllable utility cost.

2 Related Work

Machine Unlearning for LLMs. LLM unlearning removes a forget set's influence while preserving retain utility, judged jointly on the two (Maini et al., 2024). Most methods optimize a forget-derived loss: likelihood suppression (GA, Jang et al., 2023; GradDiff, Liu et al., 2022), preference optimization (NPO, Zhang et al., 2024a; SimNPO, Fan et al., 2025b), and representation engineering (RMU; Li et al., 2024). Others edit forget-related weights directly (pruning, attribution) or drop the retain set (Wang et al., 2025) or act only at inference (Pawelczyk et al., 2024).

Relearning robustness. Unlearned LLMs recover forgotten knowledge under modest extra training (Hu et al., 2025; Lynch et al., 2024; Łucki et al., 2025; Schwinn et al., 2024; Patil et al., 2024; Deeb and Roger, 2024; Che et al., 2025), suggesting edit-time forgetting suppresses rather than removes knowledge. Proposed defenses include sharpness-aware unlearning (Fan et al., 2025a), latent adversarial training (Sheshadri et al., 2025), tamper-resistant safeguards (Tamirisa et al., 2025), and localized edits (Guo et al., 2025). Closest to us, Siddiqui et al. (2025) tie robustness to weight-space displacement; we show a scalar global ℓ_2 distance is insufficient: which weights move, not how far, governs robustness.

Weight Importance and Pruning. Unstructured LLM pruning scores weights by activations (Wanda; Sun et al., 2024), relative importance (RIA; Zhang et al., 2024b), or second-order reconstruction (SparseGPT; Frantar and Alistarh, 2023); a parallel line localizes knowledge to FFN memories (Geva et al., 2021), neurons (Dai et al., 2022), and MLP modules (Meng et al., 2022). For unlearning, Selective Pruning (Pochinkov and Schoots, 2024) uses forget–retain activation contrast, SSD (Foster et al., 2024) dampens forget weights via Fisher information (vision), SalUn (Fan et al., 2024) uses gradient-based weight saliency, and WAGLE (Jia et al., 2024) uses gradient attribution; Jia et al. (2023) show in vision that sparsity alone eases unlearning. None target the forget–retain alignment structure governing relearning robustness; FRP builds on the importance contrast of Selective Pruning and applies it to relearning robustness (§3.3).

3 Method

We develop a weight-selective view of relearning robustness. Our starting point is that a robust unlearned model should not merely move far from the original model; its update should be concentrated on forget-critical weights while avoiding retain-critical ones. Based on this property, we first define an attack-free robustness prediction problem, then introduce FRAG as a diagnostic proxy. Finally, we instantiate the same principle as FRP, which directly constructs more robust unlearned models by enforcing this selective update structure.

3.1 Problem Formulation

Let M_0 and M_u denote the original and unlearned models with parameters θ_0 and θ_u. Given a forget set $\mathcal{D}_f$ and retain set $\mathcal{D}_r$, unlearning aims to remove the influence of $\mathcal{D}_f$ while maintaining retain-side behavior on $\mathcal{D}_r$. After unlearning, M_u may face a relearning attack by fine-tuning on an attack set $\mathcal{D}_a$ drawn from $\mathcal{D}_r$, $\mathcal{D}_f$, or their mixture, producing an attacked model M_a.

A robust unlearned model should resist recovery of forgotten knowledge while maintaining retain-side utility. Thus, robustness is not captured by post-attack forgetting alone; a utility-collapsed model is not meaningfully robust. Our goal is to identify attack-free weight-space properties that predict such robustness. Given M_0, M_u, and small calibration sets $\mathcal{D}_f^{\text{cal}}, \mathcal{D}_r^{\text{cal}}$, we seek an attack-free scoring function

$$\phi : (M_0, M_u, \mathcal{D}_f^{\text{cal}}, \mathcal{D}_r^{\text{cal}}) \mapsto \mathbb{R}, \qquad (1)$$

where $\mathbb{R}$ denotes the real numbers and higher scores indicate stronger predicted robustness under relearning attacks. Unlike global ℓ_2 distance

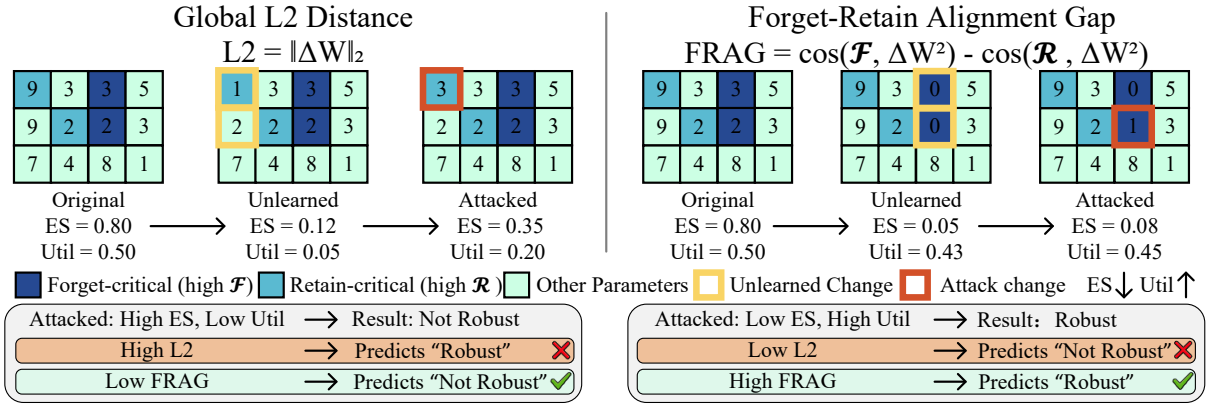

Figure 1: Illustration of global L_2 and FRAG as attack-free predictors of relearning robustness. Large L_2 can falsely suggest robustness when edits hit retain-critical weights, while small L_2 can miss robust forget-critical edits. FRAG captures both cases by measuring whether updates target forget-critical while sparing retain-critical weights.

$\|\theta_u - \theta_0\|_2$, which measures only update magnitude, we focus on where the update is concentrated.

3.2 Predicting Robustness: FRAG

A weight is *forget-critical* if its magnitude and input-channel activation indicate greater importance on forget than on retain data; *retain-critical* is defined symmetrically. Both denote *relative*, data-dependent importance rather than weights exclusive to one set: nearly every weight carries some of both, and what matters is the ratio.

For layer ℓ, let W_0^ℓ and W_u^ℓ be the original and unlearned weights. The unlearning update is

$$\Delta W^\ell = W_u^\ell - W_0^\ell. \tag{2}$$

To assess whether the update is robustly structured, we compare $(\Delta W^\ell)^2$ with forget- and retain-critical weight importance.

For each input channel j, we collect activation norms on forget and retain calibration data, denoted $x_j^{f,\ell}$ and $x_j^{r,\ell}$. Following the weight importance (Sun et al., 2024), we define

$$\mathcal{F}_{ij}^\ell = |(W_0^\ell)_{ij}| \, \frac{x_j^{f,\ell}}{x_j^{r,\ell} + \epsilon}, \tag{3}$$

$$\mathcal{R}_{ij}^\ell = |(W_0^\ell)_{ij}| \, \frac{x_j^{r,\ell}}{x_j^{f,\ell} + \epsilon}. \tag{4}$$

Here, $\mathcal{F}^\ell$ and $\mathcal{R}^\ell$ denote forget- and retain-critical weight importance, respectively. Let $D = (\Delta W)^2$ denote the squared update after aggregating selected layers. We use cosine similarity because it is scale-invariant, measuring alignment rather than update magnitude:

$$\begin{aligned} A_f &= \cos(\mathcal{F}, D), \\ A_r &= \cos(\mathcal{R}, D), \\ \mathrm{FRAG} &= A_f - \gamma A_r. \end{aligned} \tag{5}$$

Algorithm 1: Forget–Retain Pruning (FRP)

Input: $\theta_0, \mathcal{D}_f, \mathcal{D}_r$, modules $\mathcal{M}$, sparsity ρ, retain penalty β, magnitude weight λ
Output: θ_u

for $m \in \mathcal{M}$ *with* $W \in \mathbb{R}^{d_o \times d_i}$ **do**
  $x^f, x^r \leftarrow$ input-channel norms on $\mathcal{D}_f, \mathcal{D}_r$;
  $\mathcal{F}_{ij} \leftarrow |W_{ij}| x_j^f / (x_j^r + \epsilon)$; $\mathcal{R}_{ij} \leftarrow |W_{ij}| x_j^r / (x_j^f + \epsilon)$;
  for $i = 1, \ldots, d_o$ **do**
    $S_{ij} \leftarrow \mathrm{rank}_j(\mathcal{F}_{ij}) - \beta\,\mathrm{rank}_j(\mathcal{R}_{ij}) + \lambda\,\mathrm{rank}_j(|W_{ij}|)$;
    $\mathcal{P}_i \leftarrow \mathrm{TopK}(S_{i,:}, \lfloor \rho d_i \rfloor)$;
    $W_{i,\mathcal{P}_i} \leftarrow 0$;
return θ_u;

Here, A_f and A_r are forget- and retain-update alignment scores. A high FRAG indicates the update aligns with forget-critical weights while avoiding retain-critical ones. The retain term ($\gamma = 1$; App. A.1) prevents forget-only alignment from rewarding destructive updates that damage utility.

Fine-tuning can only move a weight that the fine-tuning data actually uses: for a linear layer, the gradient of W_{ij} carries a factor x_j, the activation of its input channel. Weights that fire on forgotten content but not on retain content are therefore inert under a retain-only relearning attack, and an edit placed there survives it. FRAG scores exactly this placement, which is why an attack-free score can anticipate the attack. The argument is local and first-order, and it weakens once the attacker also holds forget data, which reactivates those channels; we therefore evaluate FRAG under that stronger attack (Table 3).

Computing FRAG only requires calibration forward passes and a weight comparison between M_0 and M_u; it needs no relearning attack or additional optimization. Unless specified otherwise, we score attention and MLP projection layers. Because FRAG measures directional alignment, diffuse dense updates align weakly and receive

Method	Unlearned		Retain Attack			Forget Attack			Forget+Retain Attack			Average			Predictor	
	ES ↓	Util ↑	ES ↓	ΔES ↓	Util ↑	ES ↓	ΔES ↓	Util ↑	ES ↓	ΔES ↓	Util ↑	$\overline{\text{ES}}$ ↓	$\overline{\Delta\text{ES}}$ ↓	$\overline{\text{Util}}$ ↑	L2 ↑	FRAG ↑
LLaMA-3.2-1B																
Retain	0.064	0.596	0.063	-0.001	0.597	0.081	0.017	0.580	0.071	0.007	0.597	0.072	0.008	0.591	-	-
GA	0.086	0.199	0.284	0.198	0.598	0.153	0.067	0.359	0.401	0.315	0.598	0.279	0.193	0.518	0.875	0.002
GradDiff	0.123	0.498	0.256	0.134	0.602	0.264	0.142	0.591	0.441	0.318	0.600	0.320	0.198	0.598	0.569	0.003
NPO	0.126	0.487	0.215	0.089	0.603	0.197	0.071	0.547	0.331	0.205	0.600	0.248	0.122	0.583	0.748	0.000
RMU	0.105	0.561	0.412	0.307	0.602	0.420	0.315	0.579	0.742	0.637	0.601	0.525	0.420	0.594	0.769	0.004
SP	0.124	0.486	0.171	0.047	0.521	0.200	0.076	0.497	0.237	0.113	0.520	0.203	0.079	0.513	120.6	0.393
FRP (β=0.00)	0.055	0.384	0.079	0.024	0.464	0.071	0.015	0.407	0.104	0.049	0.463	0.085	0.029	0.444	111.5	3.159
FRP (β=0.15)	0.099	0.494	0.133	0.034	0.526	0.138	0.039	0.498	0.201	0.102	0.526	0.157	0.058	0.517	81.4	3.020
LLaMA-3.2-3B																
Retain	0.064	0.658	0.071	0.007	0.655	0.085	0.022	0.639	0.082	0.019	0.651	0.080	0.016	0.649	-	-
GA	0.120	0.384	0.331	0.211	0.671	0.222	0.102	0.548	0.497	0.376	0.673	0.350	0.230	0.630	1.363	0.002
GradDiff	0.195	0.584	0.380	0.184	0.658	0.454	0.259	0.654	0.553	0.358	0.660	0.462	0.267	0.657	0.937	0.005
NPO	0.082	0.663	0.108	0.026	0.665	0.175	0.093	0.649	0.281	0.198	0.664	0.188	0.106	0.660	1.506	0.002
RMU	0.054	0.664	0.123	0.068	0.664	0.139	0.085	0.663	0.755	0.700	0.664	0.339	0.284	0.664	1.488	0.007
SP	0.187	0.589	0.299	0.112	0.613	0.431	0.244	0.597	0.443	0.257	0.618	0.391	0.204	0.610	191.9	0.941
FRP (β=0.00)	0.068	0.499	0.096	0.028	0.569	0.107	0.039	0.522	0.147	0.079	0.570	0.117	0.049	0.553	182.2	3.538
FRP (β=0.05)	0.093	0.555	0.127	0.034	0.598	0.132	0.039	0.561	0.195	0.102	0.600	0.151	0.058	0.587	162.3	3.738

Table 1: Relearning robustness on TOFU averaged over forget-set sizes. We report post-attack ES, ΔES, and utility under retain, forget, and forget+retain attacks, along with attack-free predictors. FRP achieves the lowest post-attack ES/ΔES with favorable robustness-utility tradeoff. FRAG better identifies robust updates than global ℓ_2 distance.

Method	Unlearned		Retain Attack			Predictor	
	Acc ↓	MMLU ↑	Acc ↓	ΔAcc ↓	MMLU ↑	L2 ↑	FRAG ↑
Ref	0.583	0.788	0.581	-0.002	0.792	—	—
RMU	0.479	0.782	0.552	+0.073	0.791	26.3	0.185
SP	0.515	0.764	0.513	-0.002	0.771	464.1	3.149
FRP (β=0)	0.429	0.668	0.417	-0.012	0.707	443.4	9.198

Table 2: Cross-family validation on WMDP-cyber with Qwen2.5-14B-Instruct.

substantially smaller scores; it therefore separates selective from dense updates reliably, while resolving differences *among* dense methods only coarsely. See Appendix A.1 for details.

3.3 Achieving Robustness: FRP

The same weight-selective principle can be used to construct robust unlearned models. We propose FRP, which prunes weights that are forget-important, retain-unimportant, and large enough to induce a meaningful edit.

For each target module, FRP computes the weight-aware importance scores $\mathcal{F}$ and $\mathcal{R}$ from Eq. (3)–(4). For each output row i, it scores each weight W_{ij} by

$$S_{ij} = \operatorname{rank}_j(\mathcal{F}_{ij}) - \beta \operatorname{rank}_j(\mathcal{R}_{ij}) + \lambda \operatorname{rank}_j(|W_{ij}|). \quad (6)$$

Here, $\operatorname{rank}_j(\cdot)$ ranks entries within the same output row, with larger values receiving larger ranks. The three terms favor forget-critical weights, penalize retain-critical weights, and add a magnitude prior so that pruning produces a nontrivial edit. The hyperparameters β and λ control the retain penalty and magnitude prior, respectively. This weight-level, rank-space scoring distinguishes FRP from Selective Pruning (Pochinkov and Schoots, 2024), which thresholds a raw importance ratio at the neuron level and is not evaluated for relearning robustness. Finally, FRP prunes the top $\lfloor \rho d_i \rfloor$ weights in each row according to $S_{i,:}$, as shown in Algorithm 1. See Appendix B for details and ablation study.

4 Experiments

Evaluation Setups. All experiments use OpenUnlearning (Dorna et al., 2025). We evaluate TOFU (Maini et al., 2024) on LLaMA-3.2-1B/3B (Grattafiori et al., 2024) across `forget01/05/10`, using retain, forget, and forget+retain relearning attacks, and WMDP-cyber (Li et al., 2024) on Qwen2.5-14B-Instruct (Qwen Team, 2024); Appendix C.6 adds MUSE-News (Shi et al., 2025). Baselines include GA (Jang et al., 2023), GradDiff (Liu et al., 2022), NPO (Zhang et al., 2024a), RMU (Li et al., 2024), and SP (Pochinkov and Schoots, 2024). We report ES/ΔES/utility on TOFU and Acc/ΔAcc/MMLU on WMDP, and compare global ℓ_2 distance (Siddiqui et al., 2025) with FRAG as robustness predictors. Best results are shaded first, second, third; details are in Appendix B.

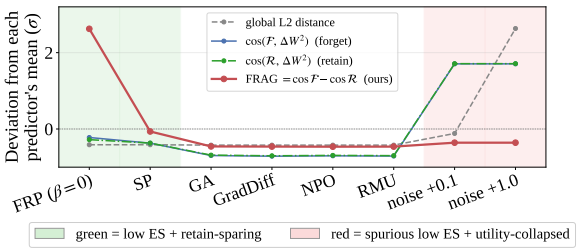

Figure 2: Curves are normalized by their own mean/std across unlearned checkpoints and noise controls. Global ℓ_2 and individual cosine terms peak on utility-collapsed perturbations, while FRAG peaks on the low-ES, retain-sparing FRP checkpoint.

	Global ℓ_2		FRAG	
	all	w/o	all	w/o
1B	-0.56	-0.36	$\mathbf{-0.92}$	$\mathbf{-0.85}$
3B	-0.17	$+0.13$	$\mathbf{-0.72}$	$\mathbf{-0.71}$
Pooled	-0.36	-0.10	$\mathbf{-0.78}$	$\mathbf{-0.74}$

Table 3: Spearman ρ between each predictor and ΔES under the forget+retain relearning attack, over healthy checkpoints only (5 methods $\times$ 3 splits $\times$ 2 models; $n = 30$ pooled). Collapsed and noise controls are excluded. More negative is better; "w/o" drops all FRP checkpoints ($n = 24$) to rule out circularity.

Main Results. Table 1 shows that FRP consistently achieves the best average post-attack ES and ΔES on TOFU across both model sizes and three relearning attacks. Although SP obtains very large global ℓ_2 distance, its performance is worse than FRP, showing that distance alone misranks update quality. Table 2 confirms the same trend on WMDP-cyber: FRP achieves the lowest cyber accuracy after retain-set relearning, and FRAG assigns it the highest robustness score despite SP having larger ℓ_2 distance. This comes at a cost: FRP's MMLU drop exceeds that of RMU and SP, so FRP traces a robustness–utility frontier rather than dominating on both axes. See Appendix B.3 for guidance on choosing β and ρ, and Appendix C for more results.

Predictor Analysis. Figure 2 isolates predictor behavior on TOFU forget10 checkpoints with noise-based collapsed controls. After per-predictor normalization, global ℓ_2 and individual cosine terms peak on collapsed perturbations, while FRAG peaks on the low-ES, non-collapsed FRP checkpoint. This shows why retain-aware alignment is necessary. Table 3 makes the comparison quantitative over healthy checkpoints only, with collapsed and noise controls removed. FRAG reaches $\rho = -0.78$ pooled against -0.36 for global ℓ_2, and the gap widens at 3B, where ℓ_2 falls to -0.17. Dropping every FRP checkpoint leaves FRAG at -0.74 while ℓ_2 falls to -0.10, reversing sign at 3B, so the ranking power does not come from FRAG scoring the method built on it. See more in Appendix A.2.

5 Conclusion

We present a weight-selective view of relearning robustness: robust unlearning depends on which weights move, not distance alone. We introduce FRAG, a training-free predictor of forget-critical and retain-sparing updates, and FRP, a direct pruning-based application of the same principle. Together, they show that forget-retain alignment provides a more reliable basis for predicting and improving relearning robustness.

Limitations

Our experiments cover TOFU, WMDP-cyber and MUSE-News across several model families and scales; broader benchmarks, multilingual data, and larger architectures would give a more complete picture. FRAG also has limited resolution within dense unlearning methods: their updates receive scores an order of magnitude smaller than selective edits, so it separates dense from selective updates far more sharply than it ranks dense methods among themselves. FRP is instantiated as unstructured pruning; the same principle may extend to structured pruning, low-rank editing, and other parameter-efficient interventions. Studying relearning attacks also inevitably shows how easily unlearned knowledge can be recovered, which could inform adversaries seeking to restore hazardous content (e.g., WMDP); we use only public benchmarks and attack protocols, frame these attacks as tools for building more robust unlearning (FRP strengthens, not weakens, resistance), and release no model with restored hazardous capabilities.

Acknowledgments

This work was partly supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. RS-2025-02264029, Integration and Validation of an AI Semiconductor-Based Data Center Training and Inference System) and (No. RS-2023-00228255, PIM-NPU Based Processing System Software Developments for Hyper-scale Artificial Neural Network Processing).

References

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Zora Che, Stephen Casper, Robert Kirk, Anirudh Satheesh, Stewart Slocum, Lev E. McKinney, Rohit Gandikota, Aidan Ewart, Domenic Rosati, Zichu Wu, Zikui Cai, Bilal Chughtai, Yarin Gal, Furong Huang, and Dylan Hadfield-Menell. 2025. Model tampering attacks enable more rigorous evaluations of LLM capabilities. *Transactions on Machine Learning Research*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

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

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. 2025. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. 2025a. Towards LLM unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. In *International Conference on Machine Learning (ICML)*.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. 2025b. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. 2024. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations (ICLR)*.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. 2024. Fast machine unlearning without retraining through selective synaptic dampening. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pages 12043–12051.

Elias Frantar and Dan Alistarh. 2023. SparseGPT: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning (ICML)*.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5484–5495, Online and Punta Cana, Dominican Republic.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The Llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Phillip Huang Guo, Aaquib Syed, Abhay Sheshadri, Aidan Ewart, and Gintare Karolina Dziugaite. 2025. Mechanistic unlearning: Robust knowledge unlearning and editing via mechanistic localization. In *International Conference on Machine Learning (ICML)*.

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. 2025. Unlearning or obfuscating? jogging the memory of unlearned LLMs via benign relearning. In *International Conference on Learning Representations (ICLR)*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 14389–14408, Toronto, Canada.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. 2023. Model sparsity can simplify machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Jinghan Jia, Jiancheng Liu, Yihua Zhang, Parikshit Ram, Nathalie Baracaldo, and Sijia Liu. 2024. WAGLE: Strategic weight attribution for effective and modular unlearning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, and 1 others. 2024. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning (ICML)*.

Bo Liu, Qiang Liu, and Peter Stone. 2022. Continual learning and private unlearning. In *Conference on Lifelong Learning Agents (CoLLAs)*, pages 243–254.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. 2025. An adversarial perspective on machine unlearning for AI safety. *Transactions on Machine Learning Research*.

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

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

Kevin Meng, David Bau, Alex J. Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Vaidehi Patil, Peter Hase, and Mohit Bansal. 2024. Can sensitive information be deleted from LLMs? objectives for defending against extraction attacks. In *International Conference on Learning Representations (ICLR)*.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. 2024. In-context unlearning: Language models as few-shot unlearners. In *International Conference on Machine Learning (ICML)*, pages 40034–40050.

Nicholas Pochinkov and Nandi Schoots. 2024. Dissecting language models: Machine unlearning via selective pruning. *Preprint*, arXiv:2403.01267.

Qwen Team. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Leo Schwinn, David Dobre, Sophie Xhonneux, Gauthier Gidel, and Stephan Günnemann. 2024. Soft prompt threats: Attacking safety alignment and unlearning in open-source LLMs through the embedding space. In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

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

Shoaib Ahmed Siddiqui, Adrian Weller, David Krueger, Gintare Karolina Dziugaite, Michael Curtis Mozer, and Eleni Triantafillou. 2025. From dormant to deleted: Tamper-resistant unlearning through weight-space regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Mingjie Sun, Zhuang Liu, Anna Bair, and J. Zico Kolter. 2024. A simple and effective pruning approach for large language models. In *International Conference on Learning Representations (ICLR)*.

Rishub Tamirisa, Bhrugu Bharathi, Long Phan, Andy Zhou, Alice Gatti, Tarun Suresh, Maxwell Lin, Justin Wang, Rowan Wang, Ron Arel, Andy Zou, Dawn Song, Bo Li, Dan Hendrycks, and Mantas Mazeika. 2025. Tamper-resistant safeguards for open-weight LLMs. In *International Conference on Learning Representations (ICLR)*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Lewis Tunstall, Edward Emanuel Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro Von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2024. Zephyr: Direct distillation of LM alignment. In *Conference on Language Modeling (COLM)*.

Yaxuan Wang, Jiaheng Wei, Chris Yuhao Liu, Jinlong Pang, Quan Liu, Ankit Shah, Yujia Bao, Yang Liu, and Wei Wei. 2025. LLM unlearning via loss adjustment with only forget data. In *International Conference on Learning Representations (ICLR)*.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024a. Negative preference optimization: From catastrophic collapse to effective unlearning. In *Conference on Language Modeling (COLM)*.

Yingtao Zhang, Haoli Bai, Haokun Lin, Jialin Zhao, Lu Hou, and Carlo Vittorio Cannistraci. 2024b. Plug-and-play: An efficient post-training pruning method for large language models. In *International Conference on Learning Representations (ICLR)*.

Appendix

The appendix expands three threads from the main text: Appendix A gives FRAG's computational recipe and the direction-blind perturbation control that motivates it. Appendix B reports FRP's implementation, evaluation protocol, and ablations over scoring, mixing weight, and sparsity. Appendix C provides the cross-family WMDP-cyber extension and the per-split TOFU breakdowns supporting the averaged main-text table.

A FRAG: Forget–Retain Alignment Gap

This appendix complements Section 3.2 with FRAG's full computational recipe (§A.1) and the direction-blind perturbation control that motivates the retain-penalty term (§A.2).

A.1 Computational Recipe

Module scope. FRAG scores every linear projection inside each transformer block (seven per block in Llama/Qwen): the four attention projections `q_proj`, `k_proj`, `v_proj`, `o_proj`, and the three MLP projections `gate_proj`, `up_proj`, `down_proj`. For other architectures the predictor falls back to all `nn.Linear` modules; reported results use this projection list.

Activation norms. Following the Wanda-style calibration of Sun et al. (2024), forward hooks on each selected module accumulate per-input-channel ℓ_2 norms of the input activation:

$$x_j^{f,\ell} \;=\; \frac{1}{N_f}\sum_{i=1}^{N_f}\big\|X_{:,:,j}^{(i),\ell}\big\|_2, \tag{7}$$

and analogously $x_j^{r,\ell}$. Both forward passes use W_0; W_u is read but never executed. Cosine similarity in Equation (5) is invariant to the per-channel norm convention.

Per-element importance and update tensors. For each layer ℓ with $W_0^\ell \in \mathbb{R}^{d_{\text{out}}\times d_{\text{in}}}$, we broadcast the channel norms along the output dimension:

$$\begin{aligned}\mathcal{F}_{ij}^\ell &= |(W_0^\ell)_{ij}| \cdot \frac{x_j^{f,\ell}}{x_j^{r,\ell}+\epsilon},\\ \mathcal{R}_{ij}^\ell &= |(W_0^\ell)_{ij}| \cdot \frac{x_j^{r,\ell}}{x_j^{f,\ell}+\epsilon},\end{aligned} \tag{8}$$

and form $D^\ell = (W_u^\ell - W_0^\ell)^2$.

Cross-layer aggregation. A_f and A_r are computed jointly across all selected layers by streaming three inner products and three squared norms, keeping memory at $O(d_{\text{out}}\times d_{\text{in}})$:

$$A_f \;=\; \frac{\sum_\ell \langle \mathcal{F}^\ell, D^\ell\rangle}{\sqrt{\sum_\ell \|\mathcal{F}^\ell\|^2}\,\sqrt{\sum_\ell \|D^\ell\|^2}}. \tag{9}$$

This single global cosine, rather than per-layer cosines that are then averaged, preserves the relative magnitude across layers so updates concentrated in a few high-importance layers are rewarded correctly.

Defaults. $\epsilon = 10^{-6}, \gamma = 1, N_f = N_r = 128$ calibration sequences of length 256, kept unchanged across all reported results. The smaller calibration here vs. FRP's $N = 400$ (Table 4) reflects that cosine alignment is robust to per-channel norm noise, while FRP's per-row top-k thresholds require more stable estimates. Computation runs in bf16 with fp32 accumulation; W_u is read shard-by-shard from safetensors so peak memory stays comparable to a single transformer layer.

Compute cost. On a single A6000, end-to-end wall-clock is 25 s on LLaMA-3.2-1B (112 projections), 1 min on 3B (196 projections), and $\sim$4 min on Qwen-14B. The shortest relearning attack we report (retain-1ep, batch 32, lr=10^{-5}) costs $\sim$30 min on 1B and over an hour on 14B per checkpoint, and yields only a single post-attack ES point. FRAG is $\sim$60$\times$ cheaper at 1B and $\geq$15$\times$ cheaper at 14B, and assigns a continuous score from weights alone.

A.2 Direction-blind Perturbation Ablation

The strongest test of FRAG against L_2 is a perturbation that performs no targeting at all: isotropic $\mathcal{N}(0,\sigma^2)$ noise added to every MLP weight of Qwen2.5-14B-Instruct (seed 42). Attention layers are untouched, no calibration data is used, and every weight is perturbed equally. We choose σ so that the resulting L_2 displacement straddles FRP's operating range: σ=0.002 gives $L_2\approx$202, and σ=0.004 gives $L_2\approx$404, the latter matching FRP (443.4) within 10%.

The noise rows of Table 8 make the distance/direction distinction sharp. At σ=0.004 the noise edit not only matches FRP's L_2 but produces a more negative ΔAcc (-0.027 vs. -0.012); a predictor that watches either signal would rank it as at least as robust as FRP. The key observation is that pre-attack Acc is essentially unchanged from

Ref (0.562 vs. 0.583) and MMLU actually *exceeds* FRP (0.740 vs. 0.668) – the noise did not unlearn, so there is nothing for the attack to undo, and the apparent robustness is vacuous. FRAG correctly demotes both noise settings to a flat 0.39 ($\times 100$), well below FRP's 9.20 on the same panel. This is the failure mode that motivates the retain term in Eq. (5): random perturbations hit forget- and retain-critical weights with equal intensity, so A_f and A_r are both large and roughly equal, and the gap collapses. With $\gamma = 0$, $A_f \approx 38\%$ exceeds FRP's 22%, the predictor would rank random destruction as more robust.

B FRP: Forget–Retain Pruning

B.1 Implementation Details

All experiments share the configuration in Table 4. Baseline-specific deviations from OpenUnlearning defaults are minimal: RMU on WMDP targets `down_proj` layers 5–7 with steering coefficient 2; SP uses `mlp_frac`=0.05, `cos_threshold`=0.5.

Component		Setting
FRP	Sparsity	3% (MLP only)
	β	0.05
	Damage mode	zero out (W'=0)
	Calibration	400 seq $\times$ 256 tok
	Precision	bf16 ($\leq$3B), fp16 (7B+)
Setup	Baselines	OpenUnlearning defaults
	Hardware	4$\times$RTX A6000 48 GB
	Seed	42 (all stages)

Table 4: FRP and infrastructure settings used across all benchmarks. Attack and evaluation protocols are in Appendix B.2.

B.2 Evaluation Protocol

Forgetting and utility. On TOFU, forgetting is measured by Extraction Strength (Carlini et al., 2021) (ES) as implemented in the OpenUnlearning evaluator. For a sequence y of length $|y|$ following prompt x, ES is one minus the normalized minimal prefix length k at which greedy continuation from $[x, y^{<k}]$ reproduces the remaining suffix:

$$\mathrm{ES} = 1 - \frac{1}{|y|} \min_k \left\{ k \;\middle|\; f\big([x, y^{<k}]; \theta\big) = y^{>k} \right\}. \tag{10}$$

ES $= 1$ indicates trivial extractability; ES $= 0$ indicates the suffix cannot be recovered. We report the per-split mean. Utility is the OpenUnlearning Model Utility scalar (geometric mean of nine retain-side metrics). On WMDP-cyber, ES is undefined for multiple-choice items, so we substitute the benchmark's 4-way accuracy via `lm-eval-harness`; the general-capability proxy is 5-shot MMLU.

Three relearning attacks. We follow the "Jogging the Memory" threat model (Hu et al., 2025): after unlearning, the adversary fine-tunes the released checkpoint for one epoch at lr=10^{-5} with AdamW (8-bit on $\geq$7B). We instantiate the attacker with three data sources to cover the realistic spectrum from white-box leakage to mask-free recovery:

- **Retain attack**: adversary fine-tunes on the retain split only. Tests whether normal continued use of the model surfaces the forgotten content. This is the canonical Hu et al. protocol and the default reported in the main text.
- **Forget attack**: adversary has direct access to the forget split itself (worst case: the leak that motivated unlearning also reveals the forget set). Smallest split, $\sim$50 optimization steps, but the strongest gradient signal toward the target content.
- **Forget+retain attack**: full white-box adversary that fine-tunes on the union, the most aggressive setting.

For each, we report post-attack ES, the change $\Delta\mathrm{ES} = \mathrm{ES}_{\mathrm{post}} - \mathrm{ES}_{\mathrm{pre}}$, and post-attack Utility. A robust unlearner should keep ES near its unlearned value across all three attacks; a method that relies on loss-surface suppression rather than knowledge removal will see ΔES inflate sharply on the forget and forget+retain attacks.

Averaging convention. Table 1 averages over the three forget splits ($1\%/5\%/10\%$) within each model. The split-wise breakdown is reported in Appendix C.2.

B.3 Ablation

We ablate FRP's three design choices on LLaMA-3.2-1B / TOFU forget10: the scoring rule (Table 5), the rank-space mixing weight β (Table 6), and the sparsity ρ (Table 7).

Scoring rule. All three magnitude-driven baselines collapse utility to zero: their top-scoring entries are the largest weights, which the retain set

also relies on, so the resulting low ES reflects degraded output rather than targeted forgetting — their FRAG scores are correspondingly near zero. Only FRP's rank-space combination of activation ratio r and $|W|$ ($\beta = 0.05$) preserves utility above 0.4 and earns a positive FRAG.

Score	Unlearned		Retain Attack			Predictor	
	ES ↓	U ↑	ES ↓	ΔES ↓	U ↑	L2 ↑	FRAG (%) ↑
Ref	0.060	0.593	0.059	-0.001	0.593	—	—
$\|W\|$ only†	0.033	0.000	0.033	0.001	0.013	236.4	-0.1
$\|W\| \cdot \|X^f\|$†	0.033	0.000	0.033	0.000	0.008	214.9	-0.1
$\|W\| \cdot r$†	0.033	0.000	0.033	0.000	0.000	235.9	+0.1
FRP (β=0.05)	0.062	0.405	0.099	0.037	0.485	111.3	+2.1

Table 5: Scoring ablation at fixed 3% MLP sparsity on TOFU/LLaMA-3.2-1B forget10. †Magnitude-driven scores collapse utility (U ≤ 0.013 after attack) and are excluded from ranking.

Mixing weight β. At fixed 3% sparsity, β traces a smooth, monotonic utility/forgetting trade-off. We fix $\beta = 0.05$ as the smallest mixing weight that matches the Retain ES_{pre} floor.

β	Unlearned		Retain Attack			Predictor	
	ES ↓	U ↑	ES ↓	ΔES ↓	U ↑	L2 ↑	FRAG (%) ↑
Ref	0.060	0.593	0.059	−0.001	0.593	—	—
0.00	0.086	0.481	0.121	+0.035	0.527	89.4	+1.9
0.01	0.082	0.464	0.112	+0.031	0.513	94.1	+1.9
0.05⋆	0.062	0.408	0.100	+0.037	0.482	111.3	+2.1
0.10	0.053	0.359	0.080	+0.027	0.443	127.0	+2.1
0.25	0.046	0.295	0.078	+0.032	0.393	147.8	+2.2
0.50	0.045	0.220	0.070	+0.025	0.324	162.7	+2.2
1.00	0.041	0.167	0.065	+0.024	0.301	177.1	+2.2

Table 6: β sweep at 3% MLP sparsity on TOFU/LLaMA-3.2-1B forget10. ⋆Headline, chosen as the smallest β matching the Retain ES_{pre} floor.

sp (%)	Unlearned		Retain Attack			Predictor	
	ES ↓	U ↑	ES ↓	ΔES ↓	U ↑	L2 ↑	FRAG (%) ↑
Ref	0.060	0.593	0.059	-0.001	0.593	—	—
0.5	0.163	0.544	0.219	0.055	0.562	60.95	+1.5
1.0	0.107	0.520	0.147	0.041	0.548	77.83	+1.7
1.5	0.090	0.477	0.126	0.036	0.521	89.07	+1.8
2.0	0.078	0.461	0.114	0.036	0.508	97.65	+1.9
2.5	0.070	0.431	0.108	0.038	0.497	105.15	+1.9
3.0	0.062	0.408	0.100	0.037	0.482	111.34	+2.1
4.0	0.055	0.375	0.084	0.029	0.451	122.77	+2.1
5.0	0.051	0.316	0.076	0.025	0.421	133.59	+2.1
7.0	0.045	0.245	0.068	0.023	0.359	152.75	+2.2
10.0	0.038	0.143	0.064	0.026	0.263	177.58	+2.4

Table 7: LLaMA-3.2-1B FRP sparsity sweep on TOFU forget10. 3.0% is the headline operating point; *Ref* is the retain90 gold model.

Sparsity ρ. At fixed $\beta = 0.05$, sparsity sweeps a Pareto frontier between forgetting and utility: ES_{pre} falls monotonically from 0.163 at 0.5% to 0.038 at 10% while utility falls from 0.544 to 0.143. We adopt 3% as the headline operating point, the smallest sparsity that matches the Retain ES_{pre} floor (0.062 vs. 0.060) while preserving utility above 0.4. The joint $(\beta, \rho) = (0.05, 3\%)$ point is selected consistently by the same criterion along both axes. In practice, ρ and β should be increased only until the target forgetting level is reached, since robustness gains beyond that point are paid for in retain-side utility. Utility-critical deployments should keep $\beta \leq 0.05$ and use the smallest sparsity that reaches the desired ES_{pre}.

C Extended Results

C.1 Cross-Family on WMDP-cyber

Table 8 extends the WMDP-cyber evaluation to Zephyr-7B-β (Tunstall et al., 2024) and reports gradient-method baselines; FRAG and post-attack accuracy agree on the ordering, while ℓ_2 is inflated by direction-blind noise.

Method	Unlearned		Retain Attack		Predictor	
	Acc ↓	MMLU ↑	Acc ↓	MMLU ↑	L2 ↑	FRAG (%) ↑
Zephyr-7B-β						
Ref	0.446	0.586	0.430	0.577	—	—
GA†	0.243	0.247	0.254	0.256	25.7	0.063
GradDiff†	0.246	0.255	0.246	0.255	27.5	0.052
NPO†	0.245	0.251	0.246	0.255	26.8	0.182
RMU	0.270	0.574	0.424	0.577	4.7	0.105
SP	0.414	0.572	0.407	0.561	52.1	2.148
FRP	0.344	0.533	0.376	0.530	49.9	6.311
Qwen2.5-14B-Instruct						
Ref	0.583	0.788	0.581	0.792	—	—
GA†	0.255	0.270	0.264	0.241	288.5	0.054
GradDiff†	0.245	0.726	0.275	0.693	507.1	0.004
RMU	0.479	0.782	0.552	0.791	26.3	0.185
SP	0.515	0.764	0.513	0.771	464.1	3.149
Noise σ=.002	0.582	0.778	0.557	0.782	201.9	0.394
Noise σ=.004	0.562	0.740	0.535	0.752	403.8	0.394
FRP	0.429	0.668	0.417	0.707	443.4	9.198

Table 8: Cross-family validation on WMDP-cyber (Zephyr-7B-β, Qwen2.5-14B-Instruct); MMLU as the general-capability proxy. † marks degenerate baselines excluded from ranking.

C.2 Full TOFU Results

Table 9 expands Table 1 along two axes the main text compresses: it separates the three relearning attacks instead of averaging them, and it reports each

forget split with its own retain-trained gold reference. Two observations follow. First, under the forget+retain attack FRP has the lowest attacked ES in every split at both scales; the ordering is less stable under the weaker retain-only and forget-only attacks. Second, the three attacks are not interchangeable: forget+retain recovers the most for every method, and a checkpoint can appear robust under the retain-only attack yet return most of the forgotten content once the attacker also holds forget data—the same asymmetry the first-order argument in §3.2 predicts.

C.3 Relearning Attack Variants

The main text fixes one attack configuration across its three attack sets. Here we vary that configuration along four axes on TOFU `forget10` with LLaMA-3.2-1B; every cell reports attacked ES / utility. The RMU and NPO checkpoints in this section differ from those in Table 9. Not every setting is an effective attack: where no method recovers meaningfully (all ΔES below 0.10), nothing was taken back from any of them and the setting says nothing about robustness. Excluding those leaves eight effective variants, and FRP has the lowest attacked ES among the unlearned methods in every one of them.

Learning rate. Too small a step recovers nothing and too large a one destroys the model, so only the middle of the range is informative (Table 10).

lr	gold	RMU	NPO	FRP
pre-attack	.060/.593	.035/.581	.085/.568	.062/.405
5e−6	.062/.590	.047/.584	.104/.565	.095/.467
1e−5	.068/.588	.461/.587	.137/.576	**.120**/.484
2e−5	.103/.582	.611/.592	.259/.590	**.178**/.499
5e−5	.135/.541	.358/.550	.270/.550	**.207**/.474
1e−4	.109/.446	.134/.445	.134/.442	**.129**/.337

Table 10: Relearning attack across learning rates (forget+retain attack set, 1 epoch, AdamW), as attacked ES / utility; lower ES means less recovery. Recovery is negligible at 5e−6; from 1e−5 to 5e−5, FRP has lower attacked ES than NPO and RMU. At 1e−4 every method loses substantial utility.

Optimizer. The attacker's optimizer matters more than its learning rate: three of the five settings fail to constitute an attack at all (Table 11).

Optimizer	gold	RMU	NPO	FRP
pre-attack	.060/.593	.035/.581	.085/.568	.062/.405
AdamW	.109/.446	.134/.445	.134/.442	.129/.337
SGD	.059/.592	.035/.581	.091/.571	.064/.416
SGD+mom.	.060/.592	.035/.574	.096/.571	.076/.439
Adafactor	.083/.297	.088/.274	.086/.280	.079/.108
Adagrad	.083/.583	.597/.590	.208/.586	**.137**/.489

Table 11: Relearning attack across optimizers (forget+retain attack set, 1 epoch, lr 1e−4), as attacked ES / utility. SGD recovers nothing—every method, including the retain-trained gold model, stays at its pre-attack ES—and Adafactor collapses utility for all methods, so neither probes robustness. "SGD+mom." uses momentum 0.9. Adagrad is the only additional effective attack, and there FRP remains far less recovered than RMU and NPO.

Attack data. Access to the forgotten examples is what makes an attack strong; indirect substitutes recover far less (Table 12).

Attack set	gold	RMU	NPO	FRP
pre-attack	.060/.593	.035/.581	.085/.568	.062/.405
retain only	.059/.594	.039/.598	.097/.585	.099/.485
forget only	.066/.573	.038/.576	.093/.535	.090/.437
forget+retain	.068/.588	.461/.587	.137/.576	**.120**/.484
paraphrase	.065/.572	.037/.576	.084/.520	.075/.425
50/50 mix	.067/.568	.087/.567	.111/.544	.101/.452
forget+general	.069/.642	.047/.597	.099/.575	.097/.454

Table 12: Relearning attack across attack-data mixtures (1 epoch, lr 1e−5, AdamW), as attacked ES / utility. Single-source and indirect mixtures recover little; "paraphrase" rewrites the forget set and "50/50 mix" balances forget and retain; forget+retain is the strongest attack, and it is there that FRP shows the lowest attacked ES among the unlearned models.

Horizon. Given enough epochs every method eventually gives the knowledge back, so the question is how fast (Table 13).

Horizon	gold	RMU	NPO	FRP
pre-attack	.060/.593	.035/.581	.085/.568	.062/.405
1 ep	.068/.588	.461/.587	.137/.576	**.120**/.484
2 ep	.084/.588	.649/.594	.200/.584	**.161**/.496
3 ep	.097/.586	.792/.596	.255/.584	**.203**/.502
5 ep	.126/.585	.936/.596	.404/.585	**.318**/.508
10 ep	.365/.575	.999/.587	.820/.566	**.752**/.504

Table 13: Relearning attack across fine-tuning horizons (forget+retain attack set, lr 1e−5, AdamW), as attacked ES / utility. FRP stays less recovered than NPO and RMU at every horizon; after 10 epochs RMU reaches .999 and NPO .820 while FRP remains at .752.

Method	Unlearned		Retain Attack			Forget Attack			Forget+Retain Attack			Predictor	
	ES↓	U↑	ES↓	ΔES↓	U↑	ES↓	ΔES↓	U↑	ES↓	ΔES↓	U↑	L2↑	FRAG (%)↑
TOFU forget01													
LLaMA-3.2-1B													
Retain	0.069	0.598	0.067	-0.002	0.597	0.096	+0.027	0.593	0.073	+0.003	0.601	2.781	-
GA	0.187	0.594	0.185	-0.002	0.601	0.309	+0.122	0.591	0.298	+0.112	0.600	0.361	-0.002
GradDiff	0.178	0.587	0.240	+0.062	0.602	0.304	+0.126	0.601	0.348	+0.170	0.599	0.314	+0.003
NPO	0.181	0.595	0.148	-0.033	0.600	0.292	+0.110	0.593	0.260	+0.079	0.600	0.353	-0.002
RMU	0.150	0.556	0.433	+0.283	0.604	0.485	+0.335	0.587	0.792	+0.642	0.601	0.284	+0.003
SP	0.079	0.456	0.104	+0.025	0.495	0.151	+0.073	0.465	0.150	+0.072	0.494	121.7	+0.689
FRP(β=0.05)	0.036	0.344	0.052	+0.015	0.445	0.041	+0.005	0.369	0.071	+0.035	0.439	111.8	+5.238
LLaMA-3.2-3B													
Retain	0.067	0.663	0.088	+0.021	0.663	0.099	+0.033	0.658	0.094	+0.028	0.663	4.760	-
GA	0.237	0.667	0.244	+0.007	0.659	0.402	+0.165	0.664	0.417	+0.180	0.658	0.576	+0.002
GradDiff	0.318	0.661	0.357	+0.039	0.657	0.501	+0.184	0.668	0.484	+0.166	0.658	0.503	+0.008
NPO	0.119	0.655	0.138	+0.019	0.657	0.182	+0.063	0.658	0.187	+0.068	0.653	1.032	+0.004
RMU	0.075	0.660	0.237	+0.162	0.663	0.240	+0.165	0.662	0.721	+0.646	0.661	0.877	+0.016
SP	0.123	0.588	0.214	+0.091	0.618	0.351	+0.228	0.592	0.330	+0.206	0.620	192.3	+1.628
FRP(β=0.05)	0.046	0.458	0.056	+0.010	0.540	0.065	+0.019	0.473	0.083	+0.037	0.545	182.3	+5.782
TOFU forget05													
LLaMA-3.2-1B													
Retain	0.063	0.598	0.063	-0.000	0.600	0.076	+0.013	0.580	0.070	+0.007	0.602	2.975	-
GA	0.039	0.002	0.276	+0.237	0.596	0.045	+0.006	0.078	0.418	+0.379	0.592	0.936	+0.003
GradDiff	0.108	0.467	0.258	+0.150	0.602	0.174	+0.066	0.578	0.462	+0.354	0.601	0.614	+0.003
NPO	0.101	0.464	0.212	+0.111	0.600	0.135	+0.033	0.504	0.339	+0.237	0.600	0.819	+0.003
RMU	0.108	0.551	0.512	+0.404	0.602	0.356	+0.248	0.577	0.733	+0.625	0.600	0.749	+0.006
SP	0.163	0.504	0.210	+0.047	0.535	0.234	+0.072	0.513	0.285	+0.122	0.532	119.8	+0.222
FRP(β=0.05)	0.067	0.399	0.086	+0.018	0.464	0.081	+0.013	0.420	0.120	+0.053	0.460	111.5	+2.156
LLaMA-3.2-3B													
Retain	0.061	0.660	0.061	-0.001	0.658	0.082	+0.021	0.649	0.077	+0.016	0.651	4.994	-
GA	0.091	0.484	0.316	+0.224	0.664	0.165	+0.074	0.584	0.491	+0.400	0.669	1.473	+0.002
GradDiff	0.162	0.562	0.385	+0.223	0.659	0.384	+0.222	0.643	0.543	+0.382	0.659	1.046	+0.003
NPO	0.070	0.662	0.101	+0.031	0.662	0.138	+0.068	0.640	0.259	+0.189	0.667	2.024	+0.001
RMU	0.054	0.665	0.073	+0.019	0.665	0.102	+0.048	0.664	0.733	+0.678	0.663	2.272	+0.001
SP	0.219	0.590	0.341	+0.122	0.611	0.486	+0.267	0.600	0.494	+0.275	0.618	191.7	+0.615
FRP(β=0.05)	0.078	0.512	0.116	+0.038	0.579	0.132	+0.054	0.539	0.183	+0.105	0.575	182.3	+2.537
TOFU forget10													
LLaMA-3.2-1B													
Retain	0.060	0.593	0.059	-0.001	0.593	0.071	+0.012	0.568	0.070	+0.010	0.589	3.255	-
GA	0.033	0.000	0.392	+0.359	0.596	0.104	+0.072	0.407	0.486	+0.453	0.601	1.311	+0.005
GradDiff	0.082	0.442	0.271	+0.189	0.602	0.315	+0.233	0.595	0.512	+0.430	0.600	0.773	+0.003
NPO	0.096	0.402	0.285	+0.190	0.609	0.164	+0.069	0.544	0.396	+0.300	0.601	1.065	+0.001
RMU	0.056	0.577	0.291	+0.235	0.599	0.420	+0.363	0.574	0.701	+0.644	0.601	1.264	+0.002
SP	0.131	0.499	0.200	+0.069	0.533	0.213	+0.082	0.513	0.275	+0.144	0.535	120.1	+0.269
FRP(β=0.05)	0.062	0.408	0.100	+0.037	0.482	0.090	+0.027	0.432	0.121	+0.059	0.488	111.3	+2.083
LLaMA-3.2-3B													
Retain	0.063	0.651	0.064	+0.001	0.645	0.075	+0.012	0.611	0.075	+0.012	0.638	5.282	-
GA	0.033	0.000	0.433	+0.401	0.689	0.098	+0.066	0.395	0.581	+0.549	0.692	2.041	+0.003
GradDiff	0.107	0.529	0.397	+0.290	0.656	0.478	+0.371	0.652	0.632	+0.525	0.663	1.257	+0.003
NPO	0.057	0.674	0.085	+0.027	0.677	0.204	+0.147	0.650	0.396	+0.339	0.672	2.559	+0.000
RMU	0.034	0.667	0.058	+0.024	0.665	0.075	+0.041	0.664	0.810	+0.776	0.669	3.082	+0.003
SP	0.217	0.587	0.341	+0.123	0.612	0.456	+0.238	0.600	0.507	+0.289	0.617	191.8	+0.580
FRP(β=0.05)	0.081	0.528	0.117	+0.036	0.588	0.125	+0.044	0.553	0.175	+0.094	0.589	181.9	+2.295

Table 9: Per-method tamper resistance on TOFU forget01/05/10. *Retain* is the gold reference (unranked).

C.4 Matched Controls

Unlearning methods differ simultaneously in utility, forgetting depth, sparsity, and update magnitude, so a raw comparison cannot attribute FRP's robustness to selective placement rather than to one of these confounds. We therefore address each in turn; the advantage survives all four, which is what attributes it to *where* the edit is placed.

Matched utility. Sweeping each method's strength knob and comparing only checkpoints of equal retain-side utility removes the possibility that FRP simply trades utility for robustness (Table 14).

	forget01	forget05	forget10
utility ≈ 0.52			
RMU	.486	.279	.304
FRP	**.101**	**.086**	**.156**
utility ≈ 0.45			
RMU	—	.263	.228
FRP	**.052**	**.056**	**.075**
utility ≈ 0.38			
RMU	—	.257	—
FRP	**.035**	**.053**	**.058**

Table 14: ΔES at matched utility (TOFU-1B, forget+retain attack, 1 epoch). Each method's strength knob is swept—steering coefficient for RMU, sparsity for FRP—and checkpoints are grouped into utility bands; "—" marks a band with no RMU checkpoint. Within every band where both methods appear, utilities differ by at most 0.01 or FRP's is higher, and FRP recovers 1.9–4.9$\times$ less.

Forgetting strength. Comparing each method's pre-attack ES against the gold level rules out the possibility that FRP merely forgets more deeply to begin with: RMU forgets deeper still (0.056 vs. 0.062) and yet recovers an order of magnitude more (Table 15).

Method	pre-ES	pre-utility	ΔES
Retain (gold)	0.060	0.593	+0.008
RMU	0.056	0.577	+0.644
GradDiff	0.082	0.442	+0.430
NPO	0.096	0.402	+0.300
SP	0.131	0.499	+0.144
FRP	0.062	0.408	**+0.059**

Table 15: Forgetting strength on `forget10`, compared against the gold pre-attack ES (0.060). FRP lands closest to the gold level (0.062) and still recovers the least; RMU forgets even more deeply (0.056) yet recovers about ten times as much.

Matched sparsity. Comparing FRP and SP at the same pruning budget isolates the scoring rule from the amount of pruning (Table 16).

Budget	Method	pre-ES	pre-utility	ΔES
2.5%	SP	0.274	0.539	+0.173
2.5%	FRP	**0.068**	0.434	**+0.067**
5%	SP	0.131	0.499	+0.144
5%	FRP	**0.050**	0.315	**+0.043**

Table 16: Matched sparsity: FRP and SP remove the same fraction of MLP weights. At equal budget FRP reaches 2.6–4.0$\times$ lower pre-attack ES and 2.6–3.3$\times$ lower ΔES. Its lower utility at the same budget reflects a stronger intervention, which is why we also report the matched-utility comparison in Table 14.

Matched update norm. Comparing the two at near-identical $\|\Delta W\|_2$ rules out displacement magnitude as the explanation (Table 17).

Edit	$\|\Delta W\|_2$	pre-ES	pre-utility	ΔES
SP 2.5%	86.3	0.274	0.539	+0.173
FRP 1.5%	89.1	**0.090**	0.483	**+0.079**
SP 5%	120.1	0.131	0.499	+0.144
FRP 4%	122.8	**0.054**	0.366	**+0.049**

Table 17: Matched update norm: FRP and SP compared at near-identical ℓ_2 update magnitudes (gaps $\leq 3\%$). FRP reaches 2.4–3.0$\times$ lower pre-attack ES and 2.2–2.9$\times$ lower ΔES, so displacement magnitude alone does not explain the gain.

C.5 Predictor Design Choices

Two ingredients of FRAG could have been chosen differently: the importance term it is built on, and the weight-space quantity it competes against. We check both.

Alternative importance measures. FRAG scores importance from weight magnitude and input activation. Swapping that term for gradient-, curvature-, or influence-based alternatives, and leaving the rest of FRAG untouched, weakens the predictor in every case (Table 18).

Importance term	Requires	ρ
Fisher g^2	backward	$+0.57$
Influence surrogate	backward	$+0.25$
Hessian $H_{jj}W^2$ (OBD)	backward	-0.79
Integrated grad. $\lvert W\bar{g}\rvert$	backward	-0.84
Activation $\lvert W\rvert\lvert X_f\rvert$ (FRAG)	forward	$\mathbf{-0.92}$

Table 18: Replacing only FRAG's importance term, evaluated on the same 15 TOFU-1B checkpoints and attack as Table 3. Negative is the correct sign. Fisher and the influence surrogate get the sign wrong; the Hessian and integrated-gradient variants are weaker and need backward passes. The activation-based term is both the strongest and the only forward-only choice.

Linear mode connectivity. Siddiqui et al. (2025) pair weight-space distance with a linear mode connectivity barrier in the vision setting. Ported to TOFU, the barrier carries almost no signal (Table 19).

Family	Checkpoints	B_f
Dense methods and FRP	42	0.000
SP (2.5–15%)	4	0.076–0.387

Table 19: Linear mode connectivity barrier B_f between θ_0 and the unlearned model, ported from Siddiqui et al. (2025) to TOFU-1B. Dense methods are GradDiff, NPO and RMU. For every healthy non-SP checkpoint the loss curve shows no upward bump at all, so there is nothing to rank by; only SP produces nonzero barriers, and there FRAG gives the same ordering.

C.6 Cross-Benchmark Check

TOFU and WMDP-cyber differ in domain but share an extraction-style evaluation. As a third setting we run MUSE-News (Shi et al., 2025), whose forget set is natural news text rather than synthetic profiles or hazardous procedures, on LLaMA-2-7B (Touvron et al., 2023) (Table 20). The ordering among utility-preserving methods carries over; broader benchmarks and architectures remain future work.

Method	Utility ↑	pre-ES ↓	ΔES ↓
GradDiff	0.048	0.008	$+0.313$
RMU	0.496	0.084	$+0.205$
SP	0.432	0.085	$+0.072$
FRP	0.379	**0.059**	$\mathbf{+0.067}$

Table 20: Cross-benchmark check on MUSE-News (LLaMA-2-7B, retain-ROUGE utility, gold $= 0.557$; retain-only relearning attack, 1 epoch). GradDiff collapses in utility; among the utility-preserving methods FRP has the lowest ΔES.