WHEN DELETION WRITES: TARGET IMPRINTS IN LLM EDITING AND UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning and model editing aim to remove specific knowledge from language models while preserving unrelated knowledge. A common approach to evaluating deletion is to test whether the deleted knowledge can still be recovered, where successful recovery is interpreted as evidence that the knowledge has survived. In this work, we show that such an interpretation can be misleading, because the deletion process itself can create a recoverable trace of the target it suppresses. We demonstrate this by applying the same deletion procedure to random target tokens that the model has never associated with the subject, and we find that these tokens become recoverable after deletion. Since these tokens had no prior association with the subject, the signals that newly emerge for them cannot represent surviving knowledge. In other words, a deletion can write the very thing that it deletes. We refer to this phenomenon as a *loss-target imprint*. Across the mechanisms we study, the imprint appears when the deletion loss directly references the target token, increases with deletion strength, and remains detectable by probes adapted to the deletion method, with the target often ranking first or second among all vocabulary tokens. These findings expose a fundamental limitation of recovery-based evaluation: failure to recover a deleted target does not establish that the target has been removed, whereas successful recovery does not establish that the recovered signal survived deletion. Our results highlight the need for unlearning evaluations that distinguish knowledge surviving deletion from signals introduced by the deletion process itself.

1 INTRODUCTION

Machine unlearning and localized model editing aim to remove specific knowledge from a trained language model while preserving unrelated knowledge, at substantially lower cost than retraining (Fang et al., 2025; Liu et al., 2025). Such deletion is increasingly considered for data-erasure requests, copyrighted or memorized content, and hazardous knowledge (Li et al., 2024a; Shi et al., 2025). Yet verifying that the underlying knowledge has actually been removed remains difficult. A common evaluation is therefore to test whether the deleted target can still be recovered from the resulting checkpoint (Hu et al., 2025; Patil et al., 2024). If recovery succeeds, the usual interpretation is that the supposedly deleted knowledge survived.

This interpretation relies on an important assumption: a target recovered after deletion must reflect information that was already present before deletion. Prior work has shown that information suppressed at the output can remain recoverable from hidden states, model weights, or subsequent fine-tuning (Łucki et al., 2025; Zhang et al., 2025). However, recovery only reveals what can be extracted from the model *after* deletion, not where that signal came from. The deletion procedure itself may create a new signal associated with its target. If so, successful recovery can reflect either surviving knowledge or information written by the deletion itself. This raises our central question: *when a deleted target is recovered, does this establish that the knowledge survived deletion?*

We answer this question using a control whose provenance is known in advance. For each deletion, we replace the true target with a random token that the base model has never associated with the subject, apply the same deletion procedure, and attempt to recover it. Because this association did not exist before deletion, any newly emerging signal cannot represent surviving knowledge. Surprisingly, these never-known targets become strongly recoverable. A mechanism-adapted probe achieves top-5

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.

recovery of 0.92 for an erasure-optimized edit and 0.49 for gradient-ascent unlearning, compared with a 0.04 baseline. In other words, deletion can write the very thing that it deletes. We call this phenomenon a *loss-target imprint*.

The imprint is closely tied to what the deletion objective explicitly targets. When the loss directly references a token, the resulting update can leave a target-specific trace even as that token disappears from the model output. Across six editing and unlearning mechanisms, we find a simple pattern: target-referencing objectives produce recoverable imprints, whereas token-agnostic RMU does not under our audit (Li et al., 2024a). The imprint also grows with deletion strength, and stronger imprints are more easily recovered. Even explicitly removing the target direction from the parameter update does not eliminate the signal, which reappears in later layers. We provide a first-order explanation for this behavior in Section 4.3.

This distinction changes how recovery should be interpreted. For privacy, a recoverable target remains a disclosure regardless of whether the signal survived or was newly written. For deletion efficacy, however, provenance is essential. Successful recovery does not by itself establish that the original knowledge survived, while failure of one recovery method shows only that this method failed. Indeed, checkpoints that pass output and logit-lens deletion checks can still reveal the target to a mechanism-adapted probe, often at rank 1 or 2 over the full vocabulary. A probe trained on another deletion mechanism can miss the same signal by orders of magnitude in rank. Recovery-based evaluation should therefore distinguish whether the target is recoverable from whether the recovered signal survived deletion or was introduced by it.

This perspective suggests a simple remedy. Alongside the real deletion, an auditor can apply the same mechanism toward a random target that the model did not previously associate with the subject. If this never-known target also becomes recoverable, recovery of the real target cannot by itself be attributed to surviving knowledge. The same finding provides a design lesson: once an objective directly references the target, suppressing the resulting imprint can weaken or relocate it but does not reliably eliminate it. Among the mechanisms we study, token-agnostic RMU is the only one with no detectable loss-target imprint under our recovery methods, although it has other limitations (Łucki et al., 2025).

Our contributions are:

- **We identify a provenance confound in recovery-based deletion evaluation.** A random-target control shows that deletion can itself create the signal observed after deletion.
- **We characterize when deletion writes its target.** Across six mechanisms, recoverable imprints track whether the objective directly references the target, and their strength predicts recoverability.
- **We show how this changes deletion auditing.** Mechanism-adapted probes recover targets missed by output, logit-lens, and cross-mechanism probes, motivating a provenance control that distinguishes surviving knowledge from deletion-induced signals.

2 Related Work

Knowledge deletion and recovery. Localized editing modifies a small set of weights so that a model stops producing a fact (Fang et al., 2025; Meng et al., 2022; 2023), whereas unlearning optimizes on a forget set using gradient ascent (Jang et al., 2023), preference objectives (Zhang et al., 2024), or representation-level objectives that do not explicitly reference a token (Li et al., 2024a). Both can also be localized to components identified through mechanistic interpretability (Guo et al., 2025). We study six such mechanisms and focus on a property that cuts across this distinction: whether the deletion objective explicitly references the target token. Prior work has shown that deletion often suppresses rather than fully erases knowledge (Hu et al., 2025; Zhang et al., 2025). Deleted knowledge can reappear after unrelated fine-tuning or removal of activation-space directions (Deeb & Roger, 2024; Łucki et al., 2025), be identified by comparing representations before and after unlearning (Goel et al., 2026), or be extracted by contrasting post- and pre-deletion checkpoints (Wu et al., 2025). These approaches study information that existed before deletion. Our question is different: whether a signal recovered after deletion may instead have been introduced by the deletion procedure itself.

Deletion traces and verification. A related line studies traces left by editing and unlearning. Prior work detects whether a fact was edited (Youssef et al., 2025), identifies the type of edit (Li et al., 2024b), traces or reverses closed-form edits from model weights or updates (Holmov et al., 2026; Sun et al., 2026; Youssef et al., 2026), and detects unlearning from model outputs or activations (Chen et al., 2026). The latter is closest to our setting, but these methods do not ask whether the observed trace reflects pre-existing knowledge or information introduced by deletion. Unlearning-based fingerprinting (Xu et al., 2026) provides an intentional counterpart by deliberately writing a signal through unlearning. On the evaluation side, control tasks calibrate what a probe has learned (Hewitt & Liang, 2019), while existing unlearning benchmarks and verification methods evaluate whether training samples were removed (Naderloui et al., 2025; Shi et al., 2025; Sommer et al., 2022). None provides a provenance control for post-deletion recovery. Our random-target control instead applies the same deletion mechanism to a target that the model did not previously associate with the subject, allowing any resulting recoverable signal to be attributed to the deletion procedure. We additionally evaluate representation erasure with LEACE (Belrose et al., 2023b) in Section 6.2; concurrent probe-geometry alignment (Rupa & Andy, 2026) targets memorization detection rather than recovery of the deletion target.

3 Problem Statement

We study how to interpret information recovered from a language model after knowledge deletion. Following the convention, we represent a factual association as a triple (s, r, o), where s denotes the subject, r the relation, and o the corresponding object or value. For example, the fact "the capital of France is Paris" can be represented as $(\text{France}, \text{capital}, \text{Paris})$. In our setting, o is the target value to be deleted. Given a model with parameters θ, a deletion mechanism produces an updated model θ' that suppresses o for prompts expressing the relation (s, r) while aiming to preserve unrelated model behavior.

To evaluate whether a deletion is successful, a recovery method attempts to recover information about o from the post-deletion model θ'. Depending on the available access, recovery may use model outputs, intermediate representations, or model parameters. Successful recovery is commonly interpreted as evidence that the original knowledge survived deletion.

This interpretation, however, assumes that the recovered signal was already present before deletion. A signal observed in θ' may instead have been introduced by the deletion procedure itself. Evaluating a deleted model therefore requires separating two questions: **1. Recovery:** can the deleted target o still be recovered from θ'? **2. Provenance:** if o is recoverable, does the signal reflect knowledge that survived deletion, or was it introduced by the deletion procedure? Our main goal is to distinguish these two questions.

3.1 Auditing Setting

We consider a first-party audit performed by the model owner. The auditor knows the deletion target o, the deletion mechanism, and the deletion set, and has access to the post-deletion checkpoint θ' and its intermediate representations. The goal is to determine whether o remains recoverable after deletion and, crucially, whether a recovered signal reflects surviving knowledge or information introduced by the deletion procedure itself.

Our main experiments consider single-fact deletion, where each checkpoint contains one deleted association. We study multi-target deletion separately in Section 6.3.

3.2 Deletion Mechanisms

We introduce Eras++, an erasure-optimized editing method that serves as a strong deletion baseline. Eras++ distributes rank-one updates across multiple layers and augments its suppression loss with a logit-lens penalty on the target, thereby removing o from both the model output and the ordinary logit lens. We evaluate it alongside the five existing mechanisms discussed in Section 2: ROME (Meng et al., 2022), MEMIT (Meng et al., 2023), GA (Jang et al., 2023), NPO (Zhang et al., 2024), and RMU (Li et al., 2024a). Implementation details for all six are provided in Appendix C.

4 DELETION WRITES WHAT IT DELETES

4.1 RANDOM-TARGET PROVENANCE CONTROL

A central difficulty in interpreting post-deletion recovery is that the true target o was already associated with the subject before deletion. If o remains recoverable afterward, the signal may reflect either surviving knowledge or information introduced by the deletion procedure.

To separate these possibilities, we introduce a *random-target provenance control*. For each factual association (s, r, o), we sample a random target o_{rand} from the candidate objects and apply the same deletion mechanism toward o_{rand} instead of o. The base model has no corresponding association between s and o_{rand}. We then use the same subject-disjoint evaluation protocol to test whether o_{rand} becomes recoverable after deletion.

Because the association with o_{rand} did not exist before deletion, any newly recoverable signal for it must have been introduced by the deletion procedure. As a no-op control, we apply the same recovery procedure to the unmodified model.

4.2 EVIDENCE OF DELETION-INDUCED IMPRINTS

Random targets become recoverable after deletion. We first report GA and Eras++, an erasure-optimized edit constructed for this audit that spreads rank-one updates over layers $\{9, \ldots, 23\}$ and adds a logit-lens penalty on the target to its suppression loss (Appendix C). The provenance control reveals a strong deletion-induced signal. For Eras++, the adapted probe recovers o_{rand} with top-5 accuracy 0.92, and for GA it reaches 0.49, compared with a 0.04 baseline. In contrast, applying the same procedure to the unmodified model remains at chance (0.02 against 0.05). Because o_{rand} had no prior association with the subject, this signal cannot represent surviving knowledge. It is created by the deletion itself.

Deletion also increases recovery of the real target. For the true target o, a probe trained and evaluated on the base model achieves recovery of 0.50. After deletion, a probe trained and evaluated on the corresponding deleted representations reaches 0.92 for Eras++ and 0.69 for GA, increases of $+0.42$ and $+0.18$. We call this matched increase in pre- to post-deletion recoverability ***RAISE***. By contrast, probes transferred between the base and deleted representation regimes fall to 0.08–0.09 for Eras++ and 0.18–0.24 for GA. Deletion therefore changes how the target is represented rather than simply preserving its original representation.

Together, these results show that post-deletion recovery can contain a target-specific signal created by the deletion procedure itself. We call this signal a *loss-target imprint*.

4.3 WHY THE IMPRINT APPEARS

A first-order view explains why target-referencing objectives create a privileged direction for the target. Consider a next-token loss for target token t,

$$\ell(h, t) = -\log p(t \mid h), \qquad \nabla_h \ell = W_U^\top (p - e_t) = \sum_j p_j W_U[j] - W_U[t], \tag{1}$$

where W_U is the unembedding matrix. Unlike the probability-weighted contribution from the rest of the vocabulary, the target's own unembedding row $W_U[t]$ appears as a separate term. Gradient ascent reverses the update direction but preserves this target-dependent component.

Thus, when a deletion objective directly references token t, its update contains a direction determined specifically by t. To first order, the resulting representation change, $\Delta h \approx J\Delta\theta$, can inherit this component. A token-agnostic objective has no corresponding explicit target term.

We measure this effect using the *aligned magnitude*,

$$\left\| \mathrm{Proj}_{W_U[o]} \Delta h \right\|,$$

which quantifies how strongly the post-deletion representation shift aligns with the target's unembedding direction.

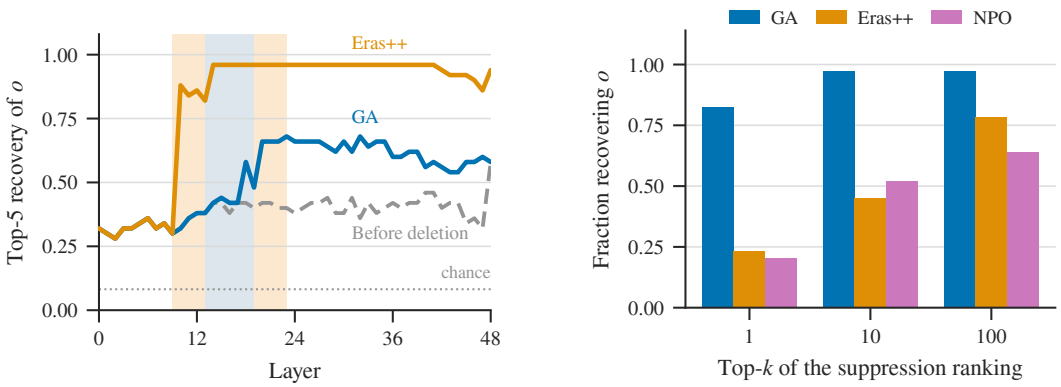


Figure 1: **Where the imprint appears and how it is read without probe training.** *Left:* layer-wise adapted-probe recovery of o before and after deletion; recovery rises around the edit band (K=61, n=50). *Right:* median rank of o among all 50k tokens when ranked by the drop in late-layer logit-lens value (n=69; chance median 25k).

The imprint emerges near the edited layers. Layer-wise recovery supports this account. In the base model, recovery of o remains around 0.36 through the middle layers. After deletion, recovery rises sharply around the layers modified by the deletion mechanism and eventually reaches 0.96 for Eras++ and 0.68 for GA (Fig. 1, left). The target-specific signal therefore emerges around the edited layers and remains recoverable downstream, even though deletion suppresses the target at the output.

This account makes a simple prediction: if the imprint is induced by the token referenced by the objective, changing which token the objective references should change which token becomes recoverable.

4.4 The Rule Across Mechanisms

We test this prediction across six editing and unlearning mechanisms. Across these mechanisms, the pattern is simple: *deletion tends to leave an imprint of the token explicitly referenced by its objective.*

Target-referencing mechanisms produce target-specific imprints. GA, Eras++, and whitened MEMIT-deletion directly reference the deletion target o, and all three produce a recoverable imprint. NPO likewise references o through its preference objective: at its utility-preserving operating point, it makes a never-known random target recoverable (0.17), while its imprint on the real target is too weak to distinguish from the unmodified model (RAISE -0.03, not significant). Increasing deletion strength raises this RAISE to $+0.46$ before fluency begins to break. ROME provides a loss-label control by targeting a different token, o_{new}, rather than the original object o. As expected, o_{new} becomes strongly recoverable (RAISE $+0.67$), and ROME aimed at a never-known random token makes that token recoverable at 0.69. In contrast, the recoverability of the untargeted o depends on what ROME installs: RAISE is $+0.10$ (not significant) when installing the same-relation CounterFact replacement, -0.14 for a random object, and -0.05 for a fixed dummy token. These controls show that the imprint follows the token named by the objective, rather than the token being removed or edited.

The effect follows the token named by the loss. We further test this relationship with a difference-in-differences control. For each fact, we apply the same edit either toward o or toward a decoy object with matched clean-model probability, and probe both resulting models for o under an object-held-out full-vocabulary evaluation. Targeting o increases its recoverability by approximately $+0.10$ more than targeting the matched decoy. GA shows the same approximately $+0.10$ contrast. The result persists when the decoy update is made larger than the deletion update and across four independently sampled decoys (Appendix A). The additional recoverability therefore follows which token enters the objective rather than edit magnitude alone.

Table 1: **The loss-target rule across deletion mechanisms.** RAISE is the increase in adapted-probe recoverability from before to after deletion; Gain is the recoverability added by post-deletion representations over the early-layer baseline; Random-target is recovery when the same mechanism targets a never-known random token (a property of the mechanism, so listed once per mechanism). All rows use K=94, n=103, except ROME's o_{new} row (K=84, n=57). Statistics in Appendix A.

Mechanism	Audited token	RAISE	Gain	Random-target
Audited token is named by the objective				
Eras++ (ours)	o	+0.42	+0.53	0.92
GA	o	+0.18	+0.27	0.49
NPO	o	−0.03	+0.07	0.17
MEMIT-deletion	o	+0.20	+0.30	0.34
ROME	o_{new}	+0.67	+0.74	0.69
Audited token is not named by the objective				
ROME, installs o_{new}	o	+0.10	+0.21	—
ROME, installs a random object	o	−0.14	−0.02	—
ROME, installs a fixed dummy token	o	−0.05	+0.07	—
RMU	o	−0.18	−0.09	0.04

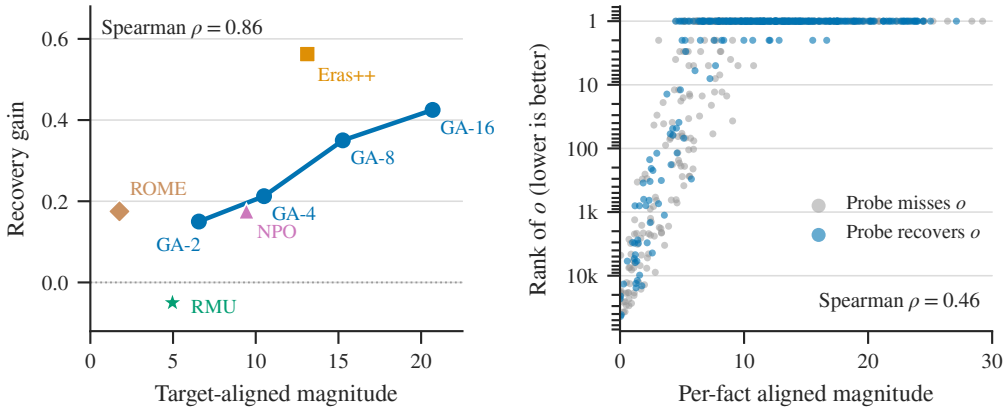


Figure 2: **Target-aligned representation change predicts recovery.** *Left:* larger target-aligned shifts across deletion configurations are associated with larger adapted-probe gains. *Right:* across individual facts, larger aligned magnitude is associated with a better differential-recovery rank for o.

Token-agnostic deletion does not show the same imprint. RMU steers activations toward a random direction without referencing a vocabulary target. At a utility-preserving operating point (forget 0.95, utility 0.95, perplexity ×1.00), random-target recovery stays at chance (0.04 against 0.06), and recovery of the real target falls by 0.18 after deletion rather than rising. We therefore find no detectable loss-target imprint for RMU, although prior work shows that knowledge removed by RMU remains recoverable through lightweight fine-tuning (Łucki et al., 2025).

Stronger deletion produces a stronger imprint. Within GA, increasing the number of ascent steps from 2 to 16 raises the aligned magnitude from 6.6 to 20.7 and the probe gain from +0.15 to +0.42 (Table 3). The same relationship appears across mechanisms. At the individual-fact level, larger aligned shifts are associated with better differential-recovery ranks for o. Across deletion configurations, they also track larger adapted-probe gains (Fig. 2).

The pattern also holds for MEMIT-deletion. At the published MEMIT band and update weight, deletion reaches forget score 0.97 without measurable utility or perplexity degradation, while the adapted probe gains +0.30 and the differential method ranks o first. Sweeping the update weight follows the same deletion-strength–imprint-strength trend. Corpus-covariance whitening changes edit specificity but not this behavior: at matched deletion strength, the whitened and isotropic-ridge variants produce nearly identical gains (+0.34 versus +0.32; Appendix A).

Table 2: **Passing standard deletion checks does not imply that the target is unrecoverable.** All checkpoints satisfy their output-level deletion criterion. B is the edit-invariant early-layer baseline; Transferred is a probe trained on a different deletion mechanism (ROME, which is therefore n/a in its own row); Adapted is trained on examples from the mechanism being audited. For full-vocabulary recovery, lower median rank indicates stronger recovery.

		Top-5 recovery ↑			Median vocabulary rank ↓	
Model	Deletion	B	Transferred	Adapted	Transferred	Adapted
Primary evaluation						
GPT-2 XL	ROME	0.39	n/a	0.61	n/a	3
GPT-2 XL	GA	0.39	0.23	0.66	331	2
GPT-2 XL	NPO	0.39	0.28	0.46	61	6
GPT-2 XL	Eras++	0.39	0.26	0.92	699	1
Cross-architecture GA						
Pythia-2.8B	GA	0.39	0.33	0.57	76	3
Gemma-4-E2B	GA	0.27	0.24	0.69	197,399	1
Qwen2.5-7B	GA	0.33	0.19	0.66	52	2

5 What Recovery Can and Cannot Establish

Section 4 shows that deletion can itself create a recoverable signal of its target. This changes how post-deletion recovery should be interpreted. We first show that standard deletion checks can miss recoverable traces. We then distinguish recoverability from provenance. Finally, we show how the random-target control can be incorporated into deletion auditing.

5.1 Standard Deletion Checks Can Miss Recoverable Traces

A checkpoint can satisfy standard deletion checks while leaving its target strongly recoverable from internal representations. For GA, the target is outside the output top-5 for every fact after deletion, its ordinary logit-lens evidence is driven close to zero, retained top-1 accuracy on 100 held-out known facts remains 0.73, and neutral-text perplexity rises by 15%. Nevertheless, a mechanism-adapted probe recovers the target at 0.66, compared with an edit-invariant baseline of 0.39. For Eras++, recovery reaches 0.92.

The conclusion also depends strongly on the recovery method used by the auditor. A probe trained on ROME and transferred unchanged to GA and Eras++ recovers only 0.23 and 0.26, respectively, below even the edit-invariant baseline. Over the full vocabulary, the mechanism-adapted probe ranks the deleted target at median rank 2 for GA and 1 for Eras++, whereas the transferred probe ranks it at 331 and 699. Thus, failure of an output check, ordinary logit lens, or transferred probe shows only that the tested recovery method failed. It does not establish that the target is no longer recoverable.

The result is not explained by a small candidate set or probe memorization. When test object tokens are entirely excluded from probe training, the adapted probe still ranks the target at median rank 16 for Eras++ and 38 for GA over the full vocabulary, compared with 2440 for the matched early-layer baseline. Recovery also persists across paraphrases of the deletion prompt. Additional probe and prompt controls are reported in the appendix.

The imprint can also be exposed without training a probe. Ranking vocabulary tokens by how strongly deletion suppresses their late-layer logit-lens value places the target at median ranks 1, 27, and 8 for GA, Eras++, and NPO, respectively (Fig. 1, right). The ordinary logit lens misses this signal because it measures the positive evidence that deletion suppresses, whereas the differential comparison measures the suppression itself.

The same pattern appears across architectures and datasets. For example, a Qwen2.5-7B GA checkpoint removes every target from the output while retaining 0.81 known-fact utility, yet the adapted probe places the target at median rank 2 over a vocabulary of 152,064 tokens. We report the full generalization results in the appendix.

5.2 Recovery Does Not Establish Survival

The results above establish that a deleted target can remain recoverable after standard deletion checks pass. They do not establish that the recovered signal represents knowledge that survived deletion.

The random-target control in Section 4.1 makes this distinction explicit. After Eras++, the mechanism-adapted probe recovers the true target at 0.92. The same deletion procedure makes a random target that the base model never associated with the subject recoverable at the same rate, 0.92. GA similarly makes a never-known random target recoverable at 0.49. Because these random associations did not exist before deletion, their recovery cannot reflect surviving knowledge. The signal was introduced by the deletion procedure.

Recovery of a real target can therefore have at least two sources. One is information that survived deletion. The other is a loss-target imprint introduced by the deletion procedure. Observing that the target is recoverable after deletion does not distinguish between them. Successful recovery therefore establishes *recoverability*, but not *provenance*.

The converse also requires care. Failure of a particular recovery method shows only that this method did not recover the target. As the results above show, output and logit-lens checks can fail to recover a target that a mechanism-adapted probe readily exposes. A negative recovery result therefore does not establish that no recoverable signal remains.

This distinction matters differently depending on the goal of the audit. For privacy, a recoverable target can still constitute information exposure regardless of whether the signal survived deletion or was newly introduced. For evaluating deletion efficacy, however, provenance is essential. Recovery alone cannot determine whether the original knowledge survived.

5.3 A Provenance-Aware Deletion Audit

The random-target control provides a simple way to account for this ambiguity. Alongside the real deletion, the auditor applies the same mechanism at the same operating point toward a random target o_{rand} that the base model did not previously associate with the subject. The auditor then applies the same recovery method and evaluation protocol to both conditions.

Recovery of o_{rand} answers a direct provenance question: *can the deletion procedure itself create a recoverable signal of a target that was not previously associated with the subject?* In our main evaluation, random-target recovery reaches 0.92 for Eras++, 0.49 for GA, and 0.17 for NPO. For token-agnostic RMU, it remains at chance.

The random-target result should not be subtracted from recovery of the real target. The two conditions have different pre-deletion baselines and therefore do not provide an additive decomposition of surviving knowledge and deletion-induced signal. Instead, the control establishes whether the deletion mechanism itself is sufficient to produce target recovery. If a never-known target becomes recoverable after the same operation, recovery of the real target cannot by itself be attributed to surviving knowledge.

Therefore, we recommend reporting a matched random-target control alongside recovery of the real deletion target. Without such a provenance control, recovery tells the auditor what can be extracted from the post-deletion model, but not where that signal came from.

6 Can We Delete Without Writing the Target?

In this section, we evaluate three ways to weaken the imprint of a target-referencing objective and then consider avoiding the target token altogether. The imprint can be diluted or attenuated, but not reliably removed without weakening deletion or utility.

6.1 Diluting the Imprint With Decoy Targets

A simple strategy is to distribute the gradient-ascent objective across the true target o and randomly sampled decoys. Increasing the number of decoys reduces the target-aligned magnitude from 15.3 to 10.0, 7.9, 5.8, and 4.6, and weakens differential recovery, raising the median rank of o from 1 to

2, 35, and 80. On a separate evaluation, four decoys raise the median suppression rank from 3 to 2627 over the full vocabulary. Yet the mechanism-adapted probe retains a recovery gain of $+0.16$ to $+0.23$, while top-5 forget drops from 1.00 to 0.92, 0.82, 0.74, and 0.67. A larger optimization budget does not resolve this trade-off, as it restores the target-aligned signal while degrading utility. Therefore, decoys weaken the imprint without eliminating it.

6.2 Removing the Target Direction

A more direct strategy is to remove the target-aligned component from the parameter update. For GA, we project every edited-layer update onto the orthogonal complement of

$$u \propto W_U[o] \odot \gamma_{\ln_f},$$

with projection onto a random direction as a control.

The intervention removes most of the target-aligned component immediately above the edit band, while deletion performance remains unchanged. Yet the signal reappears in later layers. At the recovery layers, the aligned magnitude remains 12.3, compared with 15.3 without projection; the differential method still ranks o first; and adapted-probe gain remains essentially unchanged ($+0.33$ versus $+0.35$).

We also remove a rank-20 object-related subspace from the post-deletion representations using a LEACE-style intervention (Belrose et al., 2023b). This reduces the adapted-probe gain only from $+0.53$ to $+0.44$, while retained utility falls from 1.00 to 0.27.

The imprint is therefore not a single localized direction that can simply be projected away. Removing it locally does not prevent a recoverable target-specific signal from reappearing downstream.

6.3 Batching Attenuates Some Imprints but Not Others

When k facts are deleted jointly with GA, recovery gain decreases from $+0.27$ at $k = 1$ to $+0.20$, $+0.14$, and $+0.08$ (not significant) at $k = 2$, 4, and 8, while deletion remains successful. The attenuation is consistent with individual imprints being superposed in a shared update (Appendix E.1). Corpus-covariance-whitened MEMIT deletion behaves differently. Run at a wider band than in Table 1, its recovery gain remains between $+0.40$ and $+0.46$ from $k = 1$ through $k = 720$, and the differential method continues to rank the deleted target first. Although utility declines at larger k, the $k = 40$ checkpoint still retains 0.83 known-fact utility with a recovery gain of $+0.46$. Batching therefore attenuates an imprint only when the joint update causes target-specific signals to interfere; it does not provide a general route to removing the imprint.

6.4 Deleting Without Naming the Target

The loss-target rule suggests a direct alternative: keep the target token out of the deletion objective. RMU does so, and as shown in Section 4.4, our audit finds no detectable loss-target imprint for it, although the removed knowledge remains recoverable through fine-tuning. Among the mechanisms we study, its token-agnostic objective is therefore the only one that avoids the imprint. This addresses the failure mode identified here, but does not by itself establish that the underlying knowledge has been removed.

7 Conclusion

Deletion writes what it deletes. A random token the model never knew becomes as recoverable as a real target when a deletion names it, the write is the target's unembedding direction with a sign that records suppression, and it survives every attempt we made to remove it short of not naming the target at all. Two things follow for practice. Certification should quantify over a family of recovery methods at a stated access level, because output, logit-lens, and transferred probes pass checkpoints that an adapted probe reads at median rank 1 to 2 over the vocabulary, and that a training-free lens reads at rank 1 for GA. Unlearning evaluation should carry a placebo arm, because a recovery attack alone cannot separate what a model kept from what its deletion wrote. Until a deletion can be shown not to write its target, a checkpoint that has forgotten a fact is a checkpoint that has recorded it. Limitations, including what the audit does not decide and where the imprint is faint, are discussed in Appendix G.

AI USE STATEMENT

Generative AI was used in this work to assist with coding and to polish the writing of the paper. All technical content, including the research ideas, methodology, experimental design, and interpretation of results, was developed, reviewed, and verified by the authors. We reviewed all AI-assisted edits to ensure that they preserve the intended meaning. The authors take full responsibility for the final content of this work, including any text produced with the assistance of generative AI.

ETHICS STATEMENT

This paper describes a method that recovers a deleted token from a checkpoint whose owner believes the token to be gone, so it is dual-use in the ordinary sense of security research. We think disclosure is the right call: the recovery methods we field are cheap and use only public checkpoints and standard probing, the affected deletion methods are already deployed and already certified by the very attacks we show to be insufficient, and a defense follows directly from the mechanism we identify (keep the target out of the loss's direct gradient). No human subjects, private data, or personally identifying information were involved; every fact we delete comes from a public benchmark of encyclopedic triples. The practical implication is a warning rather than a capability: a deployed checkpoint certified by an output or logit-lens audit may still disclose the very token it was asked to delete, and operators handling deletion requests under a legal right-to-erasure should not read such a certificate as compliance.

REPRODUCIBILITY STATEMENT

All experiments use publicly available checkpoints (GPT-2 XL, Pythia-2.8B, Gemma-4-E2B, Qwen2.5-7B) and public datasets (CounterFact, T-REx, RWKU). The audit setting, the access and supervision levels, and the definitions of the recovery methods are stated in full in Appendix F.1. The mechanisms and their losses are in Appendix C, and Appendix C.2 lists every deviation of our implementations from the published ones, together with the numerical checks used to confirm the closed-form algebra. The audit protocol – probe architecture, read layers, data splits, the shuffled-label and no-op controls, and the baseline recovery methods – is in Appendix F. The preregistered difference-in-differences, its magnitude matching, and the paired statistics used throughout are described with the results they support in Appendix A.

REFERENCES

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023a.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. In *Advances in Neural Information Processing Systems*, volume 36, 2023b.

Yiwei Chen, Soumyadeep Pal, Yimeng Zhang, Qing Qu, and Sijia Liu. Unlearning isn't invisible: Detecting unlearning traces in LLMs from model outputs. In *International Conference on Learning Representations*, 2026.

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

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AlphaEdit: Null-space constrained knowledge editing for language models. In *International Conference on Learning Representations*, 2025.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12216–12235, 2023. doi: 10.18653/v1/2023.emnlp-main.751.

Anmol Goel, Alan Ritter, and Iryna Gurevych. Auditing language model unlearning via information decomposition. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 808–826, 2026.

Phillip Huang Guo, Aaquib Syed, Abhay Sheshadri, Aidan Ewart, and Gintare Karolina Dziugaite. Mechanistic unlearning: Robust knowledge unlearning and editing via mechanistic localization. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 20964–20992, 2025.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 2733–2743, 2019. doi: 10.18653/v1/D19-1275.

Ali Holmov, Paul Youssef, Nandi Schoots, and Christin Seifert. One mask to rule them all: On hidden facts after editing and how to find them. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 11163–11181, 2026.

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

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 *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14389–14408, 2023. doi: 10.18653/v1/2023.acl-long.805.

Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. RWKU: Benchmarking real-world knowledge unlearning for large language models. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*, volume 37, 2024.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Ariel Herbert-Voss, Cort Breuer, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam Hunt, Justin Tienken-Harder, Kevin Shih, Kemper Talley, John Guan, Ian Steneker, David Campbell, Brad Jokubaitis, Steven Basart, Stephen Fitz, Ponnurangam Kumaraguru, Kallol Karmakar, Uday Tupakula, Vijay Varadharajan, Yan Shoshitaishvili, Jimmy Ba, Kevin Esvelt, Alexandr Wang, and Dan Hendrycks. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning*, pp. 28525–28550, 2024a.

Xiaopeng Li, Shasha Li, Shangwen Wang, Shezheng Song, Bin Ji, Huijun Liu, Jun Ma, and Jie Yu. Identifying knowledge editing types in large language models. *arXiv preprint arXiv:2409.19663*, 2024b.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 7(2): 181–194, 2025. doi: 10.1038/s42256-025-00985-0.

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

Syed Naveed Mahmood, Md. Rezaur Rahman Bhuiyan, Tasfia Zaman, Jareen Tasneem Khondaker, Md. Sameer Sakib, K. M. Shadman Wadith, Nazia Tasnim, and Farig Sadeque. Representation-aware unlearning via activation signatures: From suppression to entity-signature erasure. *arXiv preprint arXiv:2601.10566*, 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372, 2022.

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

Nima Naderloui, Shenao Yan, Binghui Wang, Jie Fu, Wendy Hui Wang, Weiran Liu, and Yuan Hong. Rectifying privacy and efficacy measurements in machine unlearning: A new inference attack perspective. In *34th USENIX Security Symposium*, pp. 5545–5564, 2025.

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

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 2463–2473, 2019. doi: 10.18653/v1/D19-1250.

Anamika Paul Rupa and Anietie Andy. Probe-geometry alignment: Erasing the cross-sequence memorization signature below chance. *arXiv preprint arXiv:2605.01699*, 2026.

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 *International Conference on Learning Representations*, 2025.

David M. Sommer, Liwei Song, Sameer Wagh, and Prateek Mittal. Athena: Probabilistic verification of machine unlearning. *Proceedings on Privacy Enhancing Technologies*, 2022(3):268–290, 2022. doi: 10.56553/popets-2022-0072.

Zhiyu Sun, Minrui Luo, Yu Wang, Zhili Chen, and Tianxing He. Reverse-engineering model editing on language models. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026.

Xiaoyu Wu, Yifei Pang, Terrance Liu, and Zhiwei Steven Wu. Unlearned but not forgotten: Data extraction after exact unlearning in LLM. In *Advances in Neural Information Processing Systems*, 2025.

Zhenhua Xu, Haobo Zhang, Zhebo Wang, Qichen Liu, Haitao Xu, Wenpeng Xing, and Meng Han. ForgetMark: Stealthy fingerprint embedding via targeted unlearning in language models. *arXiv preprint arXiv:2601.08189*, 2026.

Paul Youssef, Zhixue Zhao, Christin Seifert, and Jörg Schlötterer. Has this fact been edited? detecting knowledge edits in language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 9768–9784, 2025. doi: 10.18653/v1/2025.naacl-long.492.

Paul Youssef, Zhixue Zhao, Christin Seifert, and Jörg Schlötterer. Tracing and reversing edits in llms. In *International Conference on Learning Representations*, 2026.

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

Zhiwei Zhang, Fali Wang, Xiaomin Li, Zongyu Wu, Xianfeng Tang, Hui Liu, Qi He, Wenpeng Yin, and Suhang Wang. Catastrophic failure of LLM unlearning via quantization. In *International Conference on Learning Representations*, 2025.

A PROVENANCE AND THE LOSS-TARGET RULE

Numbers use the main pool (K=94, n=103) unless stated; per-row statistics for Table 1 are in Appendix A.4, and per-experiment pools in Table 8.

A.1 THE CANARY, THE NO-OP CONTROL, AND THE REGIME GRID

The random-target canary recovers o_{rand} at 0.92 (Eras++) and 0.49 (GA), gains $+0.88$ and $+0.45$ over a 0.04 baseline (paired McNemar $p{=}8{\times}10^{-28}$ and $3.5{\times}10^{-13}$); NPO reaches 0.17 vs. 0.02 ($p{=}6{\times}10^{-5}$). The no-op control stays at chance (0.02 vs. 0.05), so the recovery is caused by the deletion. True- and random-target accuracies are not subtracted to bound a residual: their baselines (0.39 vs. 0.04) differ and they are not additive.

The $2{\times}2$ regime grid (an adapted probe trained$\times$tested on {base, deleted} activations, same facts and labels) reads o from the unmodified model at 0.50 and from the post-deletion checkpoint at 0.92/0.69 (Eras++/GA; GA's 0.69 matches the 0.66 of Appendix B.1 up to probe-seed variation). Off-diagonal cells collapse (0.08/0.09 for Eras++, 0.18/0.24 for GA), exhibiting the distribution shift that underlies the method-relative false negative. The lift concentrates in the edit band (Fig. 1, left).

A.2 LOSS-LABEL TEST AND PREREGISTERED DIFFERENCE-IN-DIFFERENCES

In-loss. Standard ROME installs o_{new} as top-1 for 0.99 of facts; the adapted probe recovers o_{new} at 0.95 over a 0.21 baseline (gain $+0.74$, 95% CI $[+0.61, +0.84]$, McNemar $p{=}5{\times}10^{-12}$), and ROME aimed at a never-known random token makes that token recoverable at 0.69. That o_{new} is read out almost perfectly is expected; the test uses it only to confirm that the recovered token follows the objective, the activation-level counterpart of the weight-space imprints of Sun et al. (2026); Youssef et al. (2026).

Out-of-loss. For o, which ROME never names, the adapted probe's gain is $+0.21$ (CI $[0.12, 0.31]$, $p{=}10^{-4}$; the per-mechanism audit re-measures it at $+0.22$, one test fact apart), about half of which is the unmodified model's intrinsic trace ($+0.11$). The remainder follows the installed token rather than o (RAISE values in Appendix A.4): ROME writes the token its loss names and none of the token it replaces.

Difference-in-differences. Because the early probe precedes every edited layer, $B_{\text{IN}}{=}B_{\text{OUT}}$ exactly (verified to machine precision), so the contrast is carried by the adapted probe. For **Eras++** (primary endpoint; $n{=}87$ over 37 held-out objects) the DiD is $+0.103$ (object-clustered 95% CI $[+0.019, +0.211]$, $p{=}0.028$; relation-clustered $p{=}0.008$; McNemar 12:3; gain $+0.126$ vs. $+0.023$). For **GA** (confirmation; $n{=}207$ over 43 objects) it is $+0.097$ (object-clustered $p{=}3{\times}10^{-4}$, relation-clustered $p{=}8{\times}10^{-4}$; McNemar 24:4; gain $+0.130$ vs. $+0.034$). The recovery claims rest separately on per-mechanism gains at $p{<}10^{-5}$, far below any family-wise threshold.

Magnitude controls. Exact probability matching is vocabulary-limited (decoy 0.009 vs. o's 0.037), so an up-weighted-decoy arm makes the OUT edit *stronger* than the IN edit (targeted log-prob drop: Eras++ 85.1 vs. 81.9; GA 53.2 vs. 52.6). The DiD stays positive and significant (Eras++ $+0.118$, $[+0.030, +0.214]$, $p{=}0.010$; GA $+0.101$, $[+0.009, +0.209]$, $p{=}0.035$; $n{=}119$ over 41 objects). Redrawing GA's decoy four times from each fact's probability-matched band ($n{=}77$ over 38 objects) gives a positive DiD on every draw ($+0.130/+0.078/+0.091/+0.117$; mean $+0.104$, s.d. 0.021) against the shared IN gain $+0.143$. Because Eras++ is our own construction, the field-standard evidence for the rule rests on GA and NPO.

A.3 RMU AND NPO: A PROVENANCE DISSOCIATION

On a second pool ($K{=}87$; $n{=}80$ for RMU, $n{=}85$ for NPO) the regime grid places the two modern unlearning methods on opposite sides of the rule. *RMU* (forget 0.96, utility 0.94, perplexity $\times 1.00$) lowers o's decodability (base 0.33 vs. RMU 0.16, RAISE -0.16), its conditional gain is not positive (-0.09, $p{=}0.09$), the canary sits at chance (0.07 vs. 0.06), and the differential lens leaves o at median rank 599. *NPO* at a credible operating point (full deletion, perplexity within 1% of base) gives RAISE $+0.08$ on this pool (base 0.42 vs. NPO 0.51; gain $+0.12$, 95% CI $[0.00, 0.24]$; recovery still partly memory-dependent, 0.62 vs. 0.48), against -0.03 on the main pool (Appendix A.4). Pushed until it damages fluency (perplexity $266\times$), NPO reaches RAISE $+0.46$ and memory-independent recovery (0.88=0.88). The rule thus holds directionally, while imprint magnitude is a property of the edit. Whether probe recovery evidences survival is mechanism-specific: for RMU yes; for editing no, although weight-space reversal (Holmov et al., 2026; Youssef et al., 2026) still recovers the association from the weights; for NPO it depends on how hard the model was pushed.

Table 3: **Deletion strength sets the aligned magnitude and the recovery.** Aligned magnitude is $\|\mathrm{Proj}_{W_U[o]}\Delta h\|$ at the read layers; gain is adapted-probe conditional gain. This pool has $K{=}125$ and $n{=}80$, so its levels sit above those of Table 1. MEMIT rows use the main pool.

Mechanism	Setting	Aligned magnitude	Gain
GA	2 steps	6.6	$+0.15$
GA	4 steps	10.5	$+0.21$
GA	8 steps	15.3	$+0.35$
GA	16 steps	20.7	$+0.42$
Eras++	default	13.1	$+0.56$
NPO	default	9.4	$+0.17$
RMU	default	5.0	-0.05
ROME	default	1.8	$+0.17$
MEMIT-deletion	weight 60000	2.2	$+0.06$ (n.s.)
MEMIT-deletion	weight 20000	6.6	$+0.30$
MEMIT-deletion	weight 8000	9.0	$+0.38$

A.4 Pools and statistics for Table 1

Every row is measured on the main pool ($K{=}94$, $n{=}103$, chance 0.05, $B{=}0.39$), except ROME's o_{new} row, whose labels are the installed objects ($K{=}84$, $n{=}57$, $B{=}0.21$; on a larger pool with $K{=}226$, $n{=}115$ it gives RAISE $+0.73$ and gain $+0.75$, $p{=}6{\times}10^{-25}$). An unmodified model gives an intrinsic gain of $+0.11$, and a probe adapted to the unmodified model recovers o at 0.50; RAISE is measured against it on the same test facts, with a paired bootstrap interval and McNemar test.

Rows whose RAISE is near zero: NPO -0.03 (95% CI $[-0.11, +0.05]$, $p{=}0.63$; gain $+0.07$, n.s.); ROME installing o_{new} $+0.10$ ($[+0.01, +0.19]$, $p{=}0.064$); ROME installing a random object -0.14 ($[-0.21, -0.06]$, $p{=}0.003$); ROME installing a fixed dummy token ("banana") -0.05 ($[-0.14, +0.03]$, $p{=}0.36$). The out-of-loss nulls bound, rather than exclude, a small residual. MEMIT-deletion runs at the band and update weight published by Meng et al. (2023) (forget 0.97, utility 1.00, aligned magnitude 6.6, lens rank 1). RMU runs in its published form at a utility-preserving operating point (forget 0.95, utility 0.95, perplexity $\times 1.00$).

A.5 The alignment law

Table 3 sweeps deletion strength on a separate pool. Within GA, the read-layer aligned magnitude $\|\mathrm{Proj}_{W_U[o]}\Delta h\|$ and the gain rise together. Eras++ sits high and NPO on the trend; RMU writes the least aligned magnitude among the deletions at the lowest gain. ROME writes the smallest aligned component of all, consistent with its o recovery being carried by the object it installs. The small residual alignment of ROME and RMU is the downstream shadow of output-level deletion, which is why provenance (canary and loss-label test), not alignment magnitude, separates surviving memory from a written imprint. Pooling facts$\times$configurations, per-fact aligned magnitude predicts the differential-lens rank of o (Spearman $\rho{=}{+}0.46$, $p{=}2{\times}10^{-31}$, $n{=}640$; recovered facts carry 11.8 vs. 8.9), and across the eight configurations above the rule it tracks the probe gain ($\rho{=}{+}0.86$, $p{=}0.023$; Fig. 2).

A.6 Whitening sets specificity, not what is written

Calibrated to the same output-deletion strength on the main pool (forget 0.97 each), the whitened and isotropic-ridge MEMIT-deletion editors write the same imprint: gain $+0.34$ vs. $+0.32$ (paired subject-clustered $\Delta{=}-0.02$, 95% CI $[-0.05, +0.00]$, $p{=}0.92$), aligned magnitude 7.7 vs. 6.9, with the whitened editor retaining more utility (1.00 vs. 0.95–0.97). At the published band and update weight the editor is recovered at $+0.30$ ($p{=}1.2{\times}10^{-7}$); the update-weight sweep is in Table 3. Comparisons across editors must therefore match deletion strength: a gentler whitened setting (forget 0.78) writes proportionally less ($+0.17$, $p{=}0.003$).

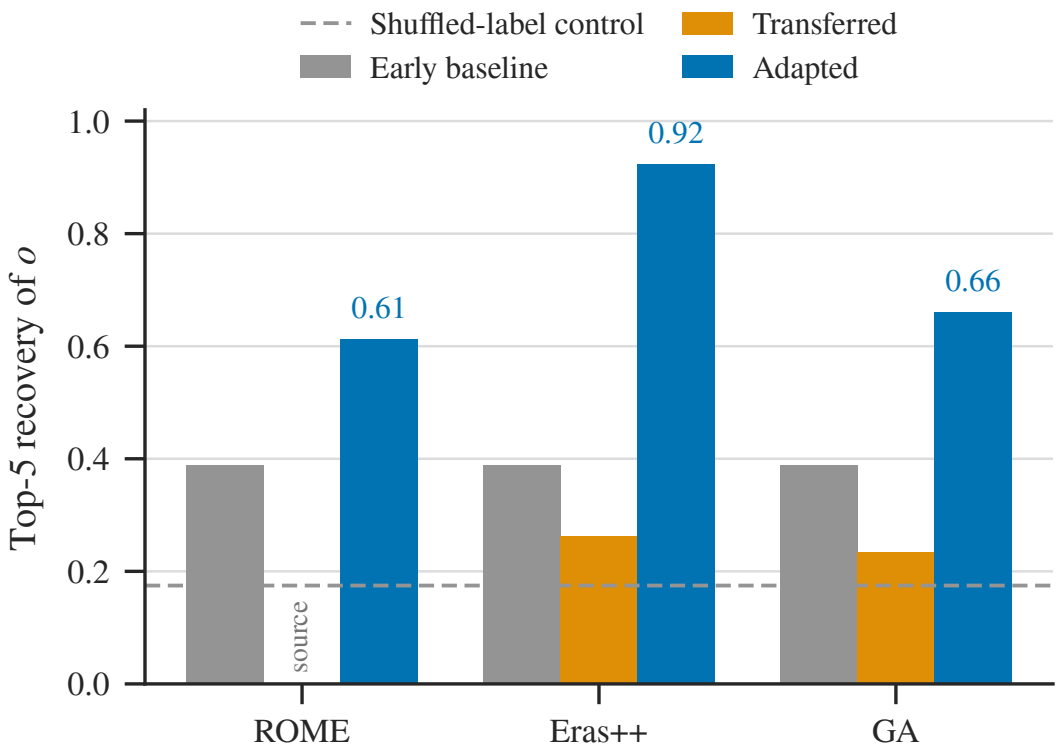


Figure 3: **Output/lens deletion is not representational deletion.** The adapted probe (blue) recovers the target far above the edit-invariant baseline (grey); the transferred probe (orange) falls below the baseline. Dashed: shuffled-label control. The GA checkpoint keeps o out of the output top-5 for 100% of facts (utility 1.00→0.73).

B Recovery results

B.1 Adapted-probe recovery and the false negatives of standard methods

On the main pool the adapted probe recovers o at 0.92 (CI $[0.85, 0.96]$) for Eras++ and 0.66 ($[0.56, 0.74]$) for GA, conditional gains of $+0.53$ ($[.44, .63]$, p=5.6×10^{-17}, discordant 55:0) and $+0.27$ ($[.17, .38]$, p=1.5×10^{-5}, 35:7) over B=0.39. On the items B misses it recovers 0.87 and 0.56. ROME's gain ($+0.22$, $[.13, .32]$, 27:4) is not an imprint of o (Appendix A.2).

For GA the output top-5 excludes o for 100% of facts and the lens reads ≈ 0, so both certify deletion. The transferred probe gives 0.23 (GA) and 0.26 (Eras++), below B itself: adding post-deletion layers through a mismatched decoder is worse than not reading them.

B.2 Full-vocabulary and object-held-out recovery

Over the entire 50,257-token vocabulary (Table 4), the adapted probe places o at median rank 1 (Eras++) and 2 (GA), against the early baseline's 48 and the transferred probe's 699/331. Because the object token can still appear as a training label for other subjects, we also hold objects out entirely (Table 5). Exact top-1 then falls to 0.006/0.006 (shuffled 0, uniform chance 2×10^{-5}), but the median rank stays far below chance: the deletion-shaped representation carries object identity that generalizes to unseen classes, while exact top-1 additionally uses the seen class.

B.3 Capacity controls

The shuffled-label probe recovers 0.16–0.20 across mechanisms (mean 0.18; above chance because it guesses frequent objects), far below the adapted probe. A linear-only probe, capacity-matched

Table 4: **Full-vocabulary recovery of** o ($|V|$=50,257, subject-disjoint, n=103). The early baseline is identical across mechanisms. Chance median rank 25,128; chance top-10 2×10^{-4}.

Recovery method	top-1	top-10	med. rank
B (early, edit-invariant)	0.12	0.36	48
Eras++ adapted $B+L$	**0.83**	**0.96**	**1**
Eras++ transferred	0.13	0.25	699
GA adapted $B+L$	**0.48**	**0.82**	**2**
GA transferred	0.10	0.24	331

Table 5: **Object-held-out full-vocabulary recovery** (51/29 train/test objects, n=158). Median rank of o over $|V|$=50,257 (chance 25,128; object-clustered 95% CI for the adapted probe). The early baseline and GA's transferred probe still beat chance, but far more weakly.

Recovery method (same split)	Eras++	GA
Adapted $B+L$ rank	**16** $[4, 65]$	**38** $[11, 303]$
(top-10 / top-100)	0.42 / 0.72	0.30 / 0.58
Early baseline rank	2440	2440
Transferred rank	28,780	20,946
Shuffled rank	42,483	29,473

between B and $B+L$, recovers nearly as well as the linear+MLP ensemble (gain $+0.50$ vs. $+0.53$ for Eras++, $+0.21$ vs. $+0.27$ for GA; $p\leq3\times10^{-4}$). An embedding-only probe on $\mathrm{hidden\text{-}states}[0]$ recovers 0.26, below B=0.39: the early baseline already carries subject→object structure, so the conditional gain is measured above a conservative floor, not a bare frequency prior.

B.4 Credibility of the deleted checkpoints

Across 40 single-fact checkpoints per mechanism, each fully deletes o while retaining top-1 accuracy on 100 held-out known facts (disjoint from deleted and retain facts) of 0.92 (Eras++), 0.84 (NPO) and 0.73 (GA); neutral-text perplexity ratios 1.01/1.01/1.15; and CounterFact neighborhood specificity 0.89/0.73/0.56, each at its utility-preserving learning rate (a larger step deletes no better but degrades fluency). The recovered target is an imprint on a working checkpoint, not an artifact of a broken one. Once forgetting is complete, a GA aggressiveness sweep trades utility for recoverability (gain 0.00→0.36 as utility 1.00→0.74), while Eras++ reaches gain 0.58 at utility 0.92.

B.5 The training-free differential lens

The deletion's residual shift points *away* from $W_U[o]$: its signed projection is negative for every GA, Eras++, NPO, RMU and ROME configuration of Table 3. Ranking all $|V|$ tokens by how much the edit suppressed their late-layer logit-lens value (base minus edited, no training) places o at median rank 1 (GA), 27 (Eras++) and 8 (NPO) (top-10 0.97/0.45/0.52; n=69 base-known facts). The absolute lens fails because it reads positive evidence, which the edit drove to ≈ 0; the differential lens reads the suppression. Its cost is one extra read of the public base model. It is strongest for gradient unlearning, which suppresses o's logit most directly; the trained probe's advantage is largest for Eras++ (rank 1 vs. 27).

B.6 Paraphrase and wrong-relation controls

On the paraphrase-eligible subset (n=95; lower absolute levels than Appendix B.1), the edit-trained probe recovers o at similar levels from the edit prompt and a paraphrase (Eras++ 0.68 vs. 0.66; GA 0.46 vs. 0.49; levels, not an equivalence test), above the early baseline (0.28 edit, 0.24 paraphrase). A probe trained on paraphrase activations also succeeds (Eras++ 0.64 $[0.54, 0.73]$; GA 0.44 $[0.35, 0.54]$). Reading the probe in a prompt for a *different* relation still recovers o (0.66 vs. correct-relation 0.67; baseline 0.24): at the probed position the imprint is *subject-keyed*. Because decoding is at the subject token under causal attention, relation words after the subject are not visible; this control varies the

relation context around the subject, and a fully after-the-relation read-out is left to future work. A recovery method therefore needs only s, which is also why the scope condition of Appendix F.1 is required.

C MECHANISMS, THEIR LOSSES, AND IMPLEMENTATION FIDELITY

C.1 MECHANISMS AND WHAT THEIR LOSSES REFERENCE

We say a mechanism's loss *names* a token t when it has a term whose gradient flows *directly* through $W_U[t]$ (the token's unembedding row), rather than only through the softmax partition function; this is the operational content of the loss-target rule.

ROME (Meng et al., 2022): a single-layer whitened rank-one edit (layer 17) optimizing $-\log p(o_{\text{new}})$; it names o_{new} only, so o enters solely through the partition function. ***MEMIT-deletion***: the same closed-form family with keys whitened by the corpus second moment C_0 (Meng et al., 2023), plus an explicit suppression term $-\log p(o_{\text{new}}) + \lambda_{\text{sup}} \log p(o)$ that names o (ours: published MEMIT installs a new object and has no deletion variant). It is run at the band published for this model ($\{13, \ldots, 17\}$, target optimized once at the band's last layer, residual spread across the band) and, for the whitening comparison, at a wider band $\{9, \ldots, 23\}$ with a value optimized per layer. ***Eras++***: an erasure-optimized edit (ours, not a published method) spreading rank-one updates over a band up to $\{9, \ldots, 23\}$ with the suppression term plus a hand-designed logit-lens penalty on o (strongest setting). ***GA*** (Jang et al., 2023): localized gradient-ascent unlearning with a retain term on the down-projections of layers $\{13, 17, 19\}$; its objective $\log p(o)$ is label-conditioned (gradient through $W_U[o]$). The retain set is 8 known facts at retain weight 3. ***NPO*** (Zhang et al., 2024): the preference objective on the GA band, 12 steps at learning rate 10^{-4} (utility-preserving; Appendix B.4). ***RMU*** (Li et al., 2024a): 40 steps at learning rate 10^{-4}, steering the forget activation at layer 17 toward a fixed random unit direction scaled by c=6, over the down-projections of the contiguous layers $\{15, 16, 17\}$, with retain weight 100; its loss never touches labels, logits or the vocabulary.

Eras++, MEMIT-deletion, GA and NPO name o; ROME names only o_{new}; RMU names no token. All localized bands lie above the early probe layers $\{2, 4, 6\}$.

C.2 FIDELITY TO THE PUBLISHED IMPLEMENTATIONS

Every mechanism was checked against its paper and official repository, and the property the rule depends on—which token each loss names—is correct for all six. The closed-form editors run at the band and update weight published for GPT-2 XL, and the rank-one algebra is the published one (verified numerically: the update reproduces the optimized value at the key to 10^{-3}, and is invariant to rescaling the inverse covariance by 10^3). ROME's value step is the published one (weight decay $\lambda\|\delta\|/\|v_0\|^2$, norm clamp at $4\|v_0\|$, at most 20 steps with early stopping, essence divergence read at the subject token), and each edit takes effect: the edited model's top-1 prediction on the edit prompt is the installed object for 0.99 of facts, and for 0.98–0.99 when the installed token is a random object or a fixed dummy (n=103). NPO's loss, inverse temperature and frozen reference match the released implementation; because the reference never changes, precomputing its log-probability is equivalent to recomputing it. Remaining deviations are listed below.

C.3 LOCALIZATION CHECK: FULL-PARAMETER GRADIENT ASCENT

Standard full-parameter GA at a utility-preserving operating point (SGD, forget 1.00, perplexity $\times$1.06; larger rates lobotomize the model) still recovers o at $B{+}L$=0.37 over chance 0.07 (K=71; gain +0.10, CI $[-0.02, +0.22]$, vs. +0.21 for the localized edit on the same candidate set). This understates the effect because full-parameter GA also perturbs B. On the K=125 pool the contrast is significant and sits on the alignment law: at equal forget, full-parameter GA writes less aligned magnitude than the localized edit (10.8 vs. 15.3) and is recovered less (+0.19, $p{=}7{\times}10^{-4}$, vs. +0.35; Fig. 6). The imprint survives full-parameter optimization, attenuated as the first-order account predicts when the $W_U[o]$ gradient is spread across all layers.

Table 6: Deviations from the published implementations. None changes which token a loss names.

mechanism	deviation
ROME	Keys and the value optimized on the bare prompt rather than averaged over the published context prefixes; covariance from wikitext with a small ridge rather than from Wikipedia without one.
MEMIT	The suppression term that turns the editor into a deletion is ours. Keys from the bare prompt rather than averaged over context prefixes.
GA, NPO	Localized to the down-projections of three layers rather than updating the full model, and applied to the target token rather than every token of the forget sequence (our targets are single tokens by construction). The retain term makes them the gradient-difference and NPO+RT variants of the published baselines. We adopt the editing literature's locus deliberately and control for it with a full-parameter run (Appendix C.3).
RMU	Retain weight and step size retuned for this model, since the published values are set for a 7B model; the steering vector is applied to the prompt's tokens.
Eras++	Ours; not a published method.

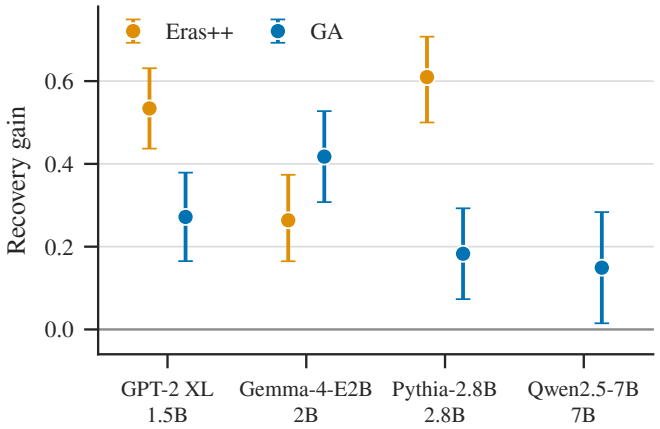


Figure 4: **The audit generalizes across architectures.** Conditional recovery gain with 95% paired-bootstrap intervals, ordered by effective size. Pythia-2.8B's GA is the clean-utility checkpoint (retained top-1 0.90).

D GENERALITY ACROSS ARCHITECTURES AND DATASETS

Architectures. Fig. 4 runs the same audit on Pythia-2.8B, Gemma-4-E2B and Qwen2.5-7B. On **Pythia-2.8B**, Eras++ reaches $B{+}L{=}1.00$ (gain $+0.61$, $B{=}0.39$, $p{<}2{\times}10^{-15}$; transferred 0.15). Its clean GA checkpoint forgets o from the output top-5 for 100% of facts while retaining 0.90 top-1 on 61 held-out facts, and is still recovered at $+0.18$ ($p{=}0.003$; $B{+}L{=}0.57$; transferred 0.33); the aggressive default schedule gives $+0.61$ but only 0.27 utility, and the 16-configuration grid shows utility falling monotonically with aggressiveness. On **Gemma-4-E2B**[1] (edited in its text decoder), Eras++ and GA give gains $+0.27$ and $+0.42$ ($B{=}0.27$; $B{+}L$ 0.54/0.69; $p{\leq}8{\times}10^{-6}$; transferred 0.36/0.24). On **Qwen2.5-7B**, top-5 GA gain is $+0.15$ ($p{=}0.052$; $B{+}L{=}0.57$ vs. transferred 0.33, below the 0.42 baseline; NPO $+0.10$; utility $1.00{\rightarrow}0.83$). This top-5 attenuation is a ceiling effect of the higher early baseline: on a credible GA checkpoint (forget 1.00, utility $1.00{\rightarrow}0.81$) the full-vocabulary probe places o at median rank 2 of 152,064 (95% subject-clustered CI $[1, 4]$; top-1 0.47, top-10 0.76) versus early-baseline rank 68 and transferred rank 52 (chance 76,032). The shallower Qwen3.5-2B saturates the early baseline ($B{=}0.98$) and is inconclusive.

The canary on a second architecture. On Pythia-2.8B (bands rescaled to depth) each mechanism's random target is recovered at $B{+}L = 0.88/0.96/0.78$ (Eras++/GA/NPO) from a 0.11 baseline (gains $+0.77/{+}0.85/{+}0.68$, each McNemar $p < 2{\times}10^{-15}$). Eras++ matches GPT-2 XL; GA and NPO write their random target more strongly here than on GPT-2 XL (0.49 and 0.17).

[1] "E" denotes *effective* ($\approx$ 2B) parameters; the checkpoint carries $\approx$ 5.1B total.

Datasets. On **T-REx** (LAMA (Petroni et al., 2019); Pythia-2.8B), with a validation-selected GA schedule and disjoint selection- and reporting-utility sets, the clean GA checkpoint forgets 99% of targets while retaining 0.90 utility (NLL 2.99, unchanged). Over $B{=}0.46$ the gain is significant for GA ($+0.17$, $p{=}0.035$) and Eras++ ($+0.52$, $p{<}10^{-8}$; transferred 0.22 vs. adapted 0.98). zsRE is uninformative for the same reason ($B{=}0.84$). On **RWKU** (Jin et al., 2024), adapted to single tokens (298 base-known cloze targets, read at the prediction token), GA deletion (forget 0.95) leaves o at full-vocabulary median rank 452 (chance $\approx 25{,}000$) against the early baseline's 807 (top-10 0.22 vs. 0.17): the imprint reproduces, attenuated, because the free-form prompt already lifts the baseline.

E Batching and defenses

E.1 Multi-target checkpoints

Naive *sequential* multi-fact deletion forms no viable checkpoint: utility collapses for GA ($1.00{\to}0.00$ at 100 deletions) and Eras++ ($\to$ 0.00 at 20), and even whitened ROME falls to 0.65 after 100 sequential edits. *Jointly batched* GA, the deployment mode for gradient unlearning, is viable (forget 1.00) and dilutes the per-fact imprint: gain $+0.27$ ($k{=}1$), $+0.20$ (2), $+0.14$ (4), $+0.08$ (8, n.s.). The decay is superposition, not probe collapse: the batched weight update stays close to the sum of single-fact updates ($\cos(\Delta W_{\text{batch}}, \sum_i \Delta W_{\text{single},i}) = 1.00/0.87/0.80/0.76$ at $k{=}1/2/4/8$, $k{=}1$ being the identity check), while each target's own aligned imprint $\langle \Delta h, \hat{W}_U[o] \rangle$ falls to $1.00/0.86/0.64/0.53$ of its single-fact value, with forget ≥ 0.94 (Fig. 5c,d).

Batched MEMIT. We build N/k checkpoints that each co-delete k facts, calibrate the regularizer per k to matched strength, and train and test on matched-k checkpoints. With faithful whitening (Meng et al., 2023) (top-5 forget ≥ 0.95 at every k, achieved 0.99), the per-fact imprint does not decay: $+0.46$ ($k{=}1$; this pool's calibrated value, versus $+0.34$ on the main pool), $+0.46$ (40), $+0.40$ (180), $+0.43$ (360), $+0.40$ (720, a single checkpoint co-deleting every fact; 95% CI $[+0.29, +0.51]$). Aligned magnitude is equally flat ($11.3{\to}10.1$) and the differential lens ranks o first at every k. Perplexity stays at $\times 1.00$–1.03, but retained known-fact accuracy falls ($1.00{\to}0.83/0.67/0.56/0.46$ at $k{=}40/180/360/720$), so $k{=}40$ is the deployment-credible case. Whitening makes co-edits near-orthogonal in the key metric and so preserves each aligned write, the activation-level counterpart of KSTER's (Sun et al., 2026) finding that the MEMIT update's rank equals the edit count. The isotropic-ridge form ($K{=}40$, $n{=}35$) loses forget rate and utility along with the imprint as k grows, so it forms no credible multi-target checkpoint to test.

E.2 Decoy randomization

Equal-weight gradient ascent on $\{o\} \cup \{4 \text{ random decoys}\}$ removes o from the output top-1 (0.95) at no utility cost (retained top-1 $0.81{\to}0.83$, perplexity $\times 1.2$) and defeats the differential lens: o's median suppression rank rises from 3 to 2627, scaling with decoy count ($72/515/3676/13000$ for $1/2/4/8$). On the $K{=}125$ pool, aligned magnitude falls with decoys ($15.3{\to}10.0/7.9/5.8/4.6$) and the lens rank with it ($1{\to}1/2/35/80$), faster than strength alone explains, since plain GA at aligned magnitude 6.6 still ranks o first: decoys add competing suppressed tokens that a ranking-based method cannot resolve.

Two limits. First, the supervised adapted probe, trained across decoy-randomized checkpoints (the family-level audit of Appendix F.1), is not reliably defeated: its gain falls from $+0.21$ to -0.05 (n.s.) on the first pool, but on the $K{=}125$ pool only from $+0.35$ to $+0.16$–0.23, plateauing ($p{\leq}0.007$ at every decoy count) and tracking the residual aligned magnitude as a weak GA does. Second, decoys weaken deletion under the top-5 criterion ($1.00{\to}0.92/0.82/0.74/0.67$), and more steps do not fix this: with 4/8 decoys, top-5 forget stays at 0.71–$0.74/0.62$–0.67 from 8 to 48 steps, while 48 steps re-concentrate the aligned write ($5.8{\to}9.9$; $4.6{\to}7.7$), return o to the lens (rank $35{\to}3$; $80{\to}11$), and degrade the model (utility $0.84{\to}0.60$, perplexity $\times 1.2{\to} \times\, 2.7$). Lens silence is bought with incomplete deletion. Decoy randomization, the activation-space analogue of KSTER's weight-space subspace camouflage (Sun et al., 2026) and akin to ERUF's (Mahmood et al., 2026) subject-signature suppression, is therefore a defense against first-order recovery methods only.

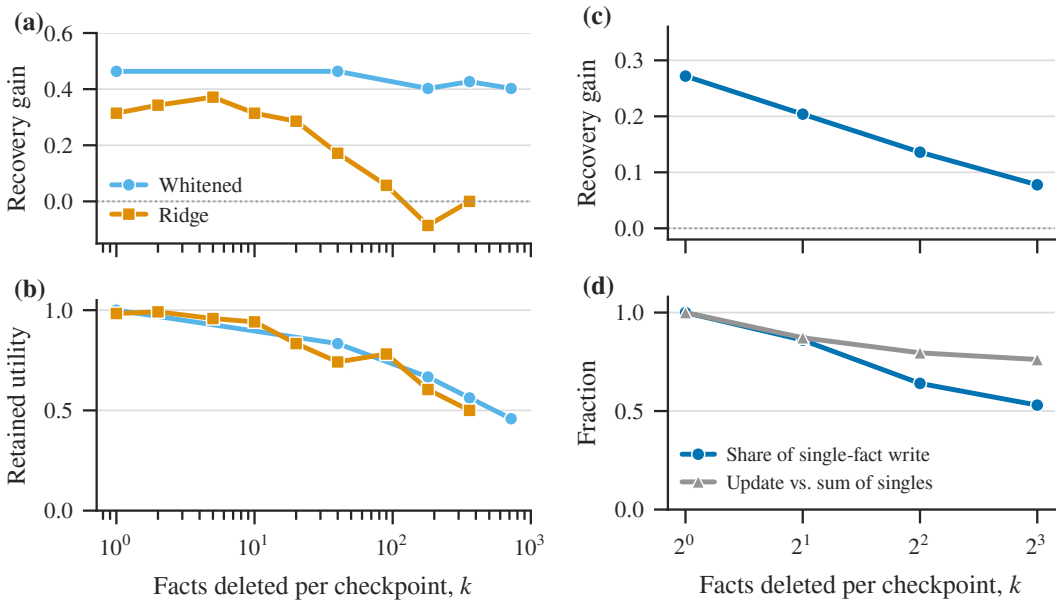


Figure 5: **Batching by mechanism.** **(a)** The whitened editor keeps every target's imprint up to k=720; the ridge variant degrades. **(b)** Utility of the same checkpoints. **(c)** Under jointly batched GA, per-target recovery falls with k. **(d)** The batched update stays close to the sum of single-fact updates, so each target retains a shrinking share of its aligned write.

E.3 Gradient projection and other defenses

We run GA with every step's weight update projected, in the residual output space of the edited down-projections, onto the orthogonal complement of $u \propto W_U[o] \odot \gamma_{\ln_f}$ (gradient-level and exact post-step projection; a random unit direction as capacity control; K=125 pool). (i) Deletion is unaffected: forget 1.00, utility 0.82, perplexity $\times$1.19 (GA: 1.00/0.81/$\times$1.21). (ii) The projection works where applied: just above the edit band the raw shift along u falls from -27.1 to -6.7 and the normed aligned shift from -9.8 to -2.7 (80 of 80 facts, sign test $p<10^{-23}$). (iii) At the read layers the imprint re-forms: aligned magnitude 12.3 vs. 15.3, the differential lens still ranks o first (top-10 0.99), and the adapted gain is unchanged ($+0.33$ vs. $+0.35$; McNemar 2:0, p=0.5). The random-direction control changes nothing (aligned 15.3, band shift -27.1, gain $+0.33$). The direct $W_U[o]$ term determines *where* the write lands, not *whether* it exists: a deletion that succeeds by lowering the target's own logit must move the late representation against $W_U[o]$, and the intervening layers reconstruct that component. The only silent deletions we find are those that do not delete through o's logit: replacement (ROME), token-agnostic scrambling (RMU), or incomplete dilution (decoys).

Other defenses. Erasing the rank-20 object-class subspace from the read layers with a LEACE-style (Belrose et al., 2023b) hook barely reduces recovery ($+0.53 \rightarrow +0.44$) while collapsing utility ($1.00 \rightarrow 0.27$). Concurrent PGA (Rupa & Andy, 2026) drives a cross-sequence memorization probe below chance and holds against re-fit probes, but targets memorization detection rather than loss-target decoding; whether it survives a decoding-adapted probe is untested. Only a token-agnostic objective such as RMU—safe against this imprint though vulnerable to fine-tuning (Łucki et al., 2025)—silences the deletion to every recovery method we field.

F Audit setting and protocol

F.1 Audit setting

A recovery method maps access to the post-deletion model and the subject s to a ranked guess for o; it needs only s, not (s, r), because the imprint is subject-keyed (Appendix B.6). Recovery

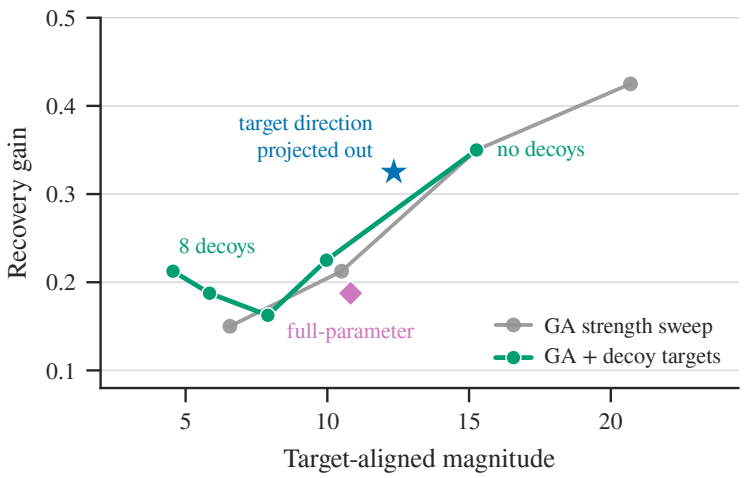


Figure 6: **Attempts to suppress the imprint move along the same curve rather than off it.** Decoys lower aligned magnitude and recovery together but leave deletion incomplete; full-parameter GA lowers both; gradient projection (star) leaves recovery unchanged. Pool K=125, n=80.

Table 7: **Recovery methods and prior deletion forensics.** Access levels as defined above (A_0 outputs, A_1 lens, A_2 activations, A_3 weights; ΔW the weight delta). *Base?*: whether the read-out needs the pre-deletion model. *Prov.*: whether the method separates a deletion-written imprint from surviving memory. *In our audit*: result on a criterion-passing GA checkpoint (— where not run).

Method	access	superv.	base?	mechanisms / decodes	prov.	in our audit
Output top-k	A_0	S_0	no	any	×	no (certifies "deleted")
Logit lens (Patil et al., 2024)	A_1	S_0	no	edits / object (∼38%)	×	no (GA drives it to ≈ 0)
Transferred probe	A_2	S_1	no	any	×	false negative
Foreign probe	A_2	S_0	no	unlearning	×	fails on RMU
Youssef et al. (2026)	A_3	S_0	no	rank-one edits / object	×	rank-one only
Holmov et al. (2026)	A_3	S_0	yes	edits (reversal)	×	rank-one only
Sun et al. (2026)	ΔW	S_0	yes	rank-one edits / subject, prompt	×	—
Wu et al. (2025)	O/W	—	yes	unlearning / content	×	—
Deeb & Roger (2024)	W (FT)	—	no	unlearning / held-out facts	×	—
Chen et al. (2026)	A, O	—	no	RMU/NPO / unlearned or not	×	—
Differential lens (ours)	A_1	S_0	yes	GA/NPO/edits / object token	via canary	yes (rank 1)
Adapted probe (ours)	A_2	S_2/S_3	**no**	**GA/NPO + edits / object token**	**via canary**	**yes (rank 1–2)**

methods differ along two axes: *access* (A_0 outputs, A_1 logit lens, A_2 activation probes, A_3 weights) and *supervision* (S_0 none, S_1 a probe transferred from a different mechanism, S_2 same-mechanism shadow edits on self-chosen facts, S_3 the labeled deletion set). The adapted probe is (A_2, S_3) for the first-party auditor, or (A_2, S_2) when trained on self-chosen facts; Table 7 places every method we compare. Output, logit-lens and transferred methods all read through a decoder not fit to the audited mechanism (even a tuned lens (Belrose et al., 2023a) is fit to the base model), so a deletion that reshapes the representation can defeat all three while leaving o decodable. A failed recovery therefore speaks only for its own cell, and a certificate of deletion must quantify over a family of methods at a stated access level.

Our main test is *conditional recoverability*: B is a probe on edit-invariant features (the subject token at early layers), $B{+}L$ additionally reads the post-deletion late layers, both are trained on the audited mechanism under a subject-disjoint split, and we test the paired gain $\mathrm{acc}(B{+}L) - \mathrm{acc}(B)$ one-sidedly. A positive gain establishes recoverability; a non-positive gain is inconclusive for the tested probe family, not proof of absence. B is edit-invariant only when the edit lies above the early read layers $\{2, 4, 6\}$; every localized mechanism satisfies this (Appendix C), the early features are verified bit-identical across the IN/OUT arms of the DiD (Appendix A.2), and the full-parameter case is tested separately (Appendix C.3). Recovery is well-defined for a subject carrying a single deleted fact; for gradient unlearning we claim single-fact recovery only (Appendix E).

Definition 1 (ε-trace-freeness)**.** Under the status-quo criterion we critique, a deletion is deemed certified if no recovery method in the auditor's set recovers o. We instead call θ' *ε-trace-free at access*

level A over a family $\mathcal{F}$ of recovery methods if every $f \in \mathcal{F}$ operating under A recovers o at no more than ε above its matched baseline, i.e., $\max_{f \in \mathcal{F}} \left[\mathrm{acc}(f) - \mathrm{acc}(\text{baseline}_f)\right] \leq \varepsilon$. A recovery method whose conditional gain has a 95% confidence interval excluding ε falsifies ε-trace-freeness; we report gains against ε=0. LEACE (Belrose et al., 2023b) is a constructive route to trace-freeness against linear probes at access level A_2.

F.2 Protocol

Probe. A read-only non-lens probe (own head, no unembedding) decodes o from residual activations at the subject token, where factual associations are enriched and read out (Geva et al., 2023). B uses early layers $\{2, 4, 6\}$ (upstream of every localized edit); $B{+}L$ adds late layers $\{28, 32, 36\}$. Each is a linear+MLP $\times$3-seed ensemble. Its supervision (same-mechanism examples with known o) is available to the first-party auditor, who holds the deletion set and can run the mechanism on self-chosen facts.

Experimental unit. Each fact is deleted *separately*: the operation is applied, activations are read, and the base model is restored, so each probe example comes from that fact's own single-deletion checkpoint. The probe is trained across many single-deletion checkpoints of the audited mechanism and tested on held-out ones. "100% forget" is per fact. Multi-target checkpoints are treated separately (Appendix E).

Splits and statistics. All splits are subject-disjoint. The candidate set is the K=94 objects appearing ≥ 2 times in the training split (a rule that biases it toward frequent objects), and recovery is scored only for facts whose o lies in it. We report top-5 recovery (chance $5/K$), Wilson intervals, a paired McNemar test of $B{+}L$ vs. B, discordant counts, a paired-bootstrap gain CI, and recovery on the B-missed subset. Per-experiment K and n are in Table 8. We report uncorrected p-values; the multiplicity policy is in Appendix A.2.

Controls. (1) A *shuffled-label* probe bounds classifier-capacity and frequency artifacts. (2) A *frozen-transfer* probe (trained on ROME, applied unchanged) is the standard leakage-safe audit. (3) The *random-target canary* samples, per record, $o_{\mathrm{rand}} \neq o$ uniformly from the K candidates, applies the same mechanism targeting o_{rand}, and trains the adapted probe to recover o_{rand}; it re-runs the mechanism once per record per target, so it is a diagnostic, not a scalable attack. (4) The *no-op* control runs that probe on the unmodified model for the same random targets. (5) The *loss-label* test applies ROME toward o_{new} and probes o_{new}, and separately applies ROME toward o_{new}, a random object token, and a fixed dummy token (losses that never name o) and probes o. (6) The *wrong-relation* control fills a deterministically chosen different-relation template with the same subject. (7) The *paraphrase* control reads the edit-trained probe at the subject's last token in a paraphrase of the fact.

Full-vocabulary and object-held-out probes. The full-vocabulary probe fits a closed-form (dual) ridge map from the post-deletion features to o's unit unembedding direction (λ chosen on a validation split by MRR) and ranks all $|V|$ tokens by cosine. The object-held-out variant partitions candidate objects into disjoint train/val/test sets (51/14/29 objects, n_{test}=158), so no test object token is ever a training label, and reports a bootstrap CI clustered over test objects.

Preregistered difference-in-differences. Its endpoint, statistic, split and λ-selection rule are emitted to the run log before any test-set number is computed, and released with the code. Each mechanism is rerun twice with identical band, key, and hyperparameters, targeting o (IN) or a per-record decoy object token whose clean prompt-probability is matched to o's (OUT), and o is probed under both on the object-held-out full-vocabulary endpoint: $\mathrm{DiD} = [(B{+}L){-}B]_{\mathrm{IN}} - [(B{+}L){-}B]_{\mathrm{OUT}}$, with object- and relation-clustered bootstrap inference. Manipulation checks confirm 100% target deletion in each arm.

Other architectures. A generic adapter auto-detects the text decoder, places the edit layer at $\mathrm{depth}/2$ with an erasure band spanning it, uses the last three layers as late layers and layers below the band as early layers. For the Pythia clean GA checkpoint we grid-search $(\text{steps}, \text{lr}) \in \{2, 3, 5, 8\} \times \{5, 10, 20, 40\}{\times}10^{-5}$ and select the utility-maximizing schedule that forgets $\geq 90\%$ (5 steps, $5{\times}10^{-5}$). On Gemma-4-E2B, whose upper 20 of 35 layers reuse earlier key–value caches, a subject-position edit at or above layer 14 never reaches the prediction, so the ROME edit that trains the transferred probe is placed at layer 13, where it installs the new object as top prediction for 0.74 of facts (at $\mathrm{depth}/2$ it installs none).

Table 8: **Per-experiment scale.** K candidate object classes, n test facts, chance top-5 $= 5/K$ (GPT-2 XL / CounterFact unless noted). All splits are subject-disjoint; the object-held-out endpoint additionally disjoins object tokens. Full-vocabulary endpoints rank over $|V|$ (chance median $|V|/2$).

experiment	appendix	K	n_{test}	chance
Main pool: Table 1, canary, regime grid, adapted-probe audit	A.4, B.1	94	103	0.05
Loss-label test, o_{new} row	A.2	84	57	0.06
RMU/NPO dissociation (second pool)	A.3	87	80/85	0.06
Alignment sweep, decoys, gradient projection	A.5, E.2	125	80	0.04
Object-held-out full-vocab	B.2	94	158	$\lvert V\rvert^{-1}$
DiD (Eras++/GA, object-held-out)	A.2	37/43 obj	87/207	$\lvert V\rvert^{-1}$
MEMIT strength-matched (k=1)	A.6	94	103	0.05
Whitened MEMIT k-sweep (k=1–720)	E.1	74	82	0.07
Ridge MEMIT k-sweep (k=1–360)	E.1	40	35	0.13
Jointly batched GA	E.1	94	103	0.05
Full-parameter GA	C.3	71	60	0.07
Decoy-randomization (first pool)	E.2	73	71	0.07
Cross-arch canary (Pythia-2.8B)	D	75	74	0.07
Cross-arch audit (Pythia-2.8B, clean GA)	D	78	82	0.06
Cross-arch audit (Gemma-4-E2B)	D	78	91	0.06
Cross-arch audit (Qwen2.5-7B)	D	73	67	0.07
Full-vocab rank (Qwen2.5-7B)	D	71	70	$\lvert V\rvert^{-1}$

G LIMITATIONS

The random-target control shows that a deletion can by itself produce recovery at the observed magnitude, but it does not say how much of a real target's recovery is surviving memory; separating the two quantitatively, for instance against a model retrained without the target, is a natural next step. Our causal tests of the loss-target rule (the difference-in-differences, the loss-label test, and the regime grid) are run on GPT-2 XL, while the recovery results and the canary replicate on larger architectures and a second dataset. The audit targets single-token facts deleted one subject at a time: jointly batched gradient unlearning dilutes the per-fact imprint, and extending the audit to multi-token, free-form forget sets and to subjects carrying several deleted facts is left to future work.