WHEN UNLEARNING IS NOT THE END: CONTROLLED ORACLE RECOVERY AGAINST FEDERATED UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated Unlearning (FU) aims to remove the influence of client data on a shared large language model upon request. However, when the target model remains trainable, subsequent updates may restore the behavior intended to be unlearned. This creates an integrity threat: a malicious participating client may counter another client's deletion request by steering the shared model back toward its pre-unlearning behavior. Existing evaluations largely focus on the immediate post-unlearning state and largely overlook the possibility of such a recovery case. We introduce Controlled Oracle Recovery, an anti-unlearning audit that evaluates whether a deletion event reveals its target records and whether small target-guided updates can reverse forgetting more effectively than compute-matched control updates. Deletion target inference reaches at least 0.999 ROC AUC in 10 of 24 evaluations. After 100 local update steps, recovery using the deletion targets exceeds the strongest matched control in all 48 comparisons. In multiple settings, recovery also restores forgotten-answer probability to its pre-unlearning level while retaining at least 95% of the utility measured before unlearning. The results reveal a broad but heterogeneous anti-unlearning susceptibility and establish a necessary local condition for a FU rollback attack on LLMs.

1 INTRODUCTION

Machine unlearning aims to remove the influence of designated training records without retraining the model from scratch. However, if the unlearned model remains trainable, subsequent optimizations can restore the behavior intended to be removed, undermining the unlearning process's guarantees. This creates a particular security concern in federated unlearning (FU): a participating client opposing another client's deletion request can receive the unlearned model, optimize it locally, and return a contribution that attempts to restore the removed behavior. This creates the challenge of post-deletion recovery attacks, as evaluating a model immediately after the request cannot capture its susceptibility to such attacks.

Existing unlearning evaluations compare forgotten behavior and retained utility with a retrained reference (Bourtoule et al., 2021; Guo et al., 2020; Maini et al., 2024). Recent studies further show that benign relearning or downstream fine-tuning can revive behavior suppressed by approximate unlearning (Hu et al., 2025; Fan et al., 2025a; Wang et al., 2025a). However, recovery alone does not provide record-specific anti-unlearning vulnerability. Exact deleted examples may simply relearn the removed records/behavior, while arbitrary in-distribution updates may alter the same endpoint through generic adaptation. Likewise, a large pre/post loss change may reveal which known records were targeted, but it is not automatically a conventional membership signal. Without matched updates on other data, recovery cannot be distinguished from ordinary continued training. And without comparing deleted records with retained training records, a loss change cannot be attributed specifically to the deletion request.

Our central insight is that post-unlearning vulnerability must be identified through matched state transitions. We introduce Controlled Oracle Recovery, which combines a deletion-target inference with a controlled recovery intervention. The first stage measures each candidate's loss change between pre-unlearning and post-unlearning states. The second initializes independent copies of the same

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.

post-unlearning state and alters only the data used in recovery: exact target records, retained records, unseen holdout records, or fluent synthetic text. We measure how far each update restores forgotten behavior relative to the states before and after unlearning. We then compare updates on deleted records with equally budgeted updates on other data and check whether retained model utility remains above the stated threshold. Finally, a one-step smoothness result shows that the target-control contrast is positive when the exact-target direction has sufficient first-order alignment with forgotten-behavior restoration.

Our contributions are:

- We test whether the change between model states before and after unlearning identifies deletion targets among known training records.
- We introduce Controlled Oracle Recovery as a separate test of whether exact deleted records restore forgotten behavior more than equally budgeted updates on other data.
- We derive a sufficient condition for stronger recovery using target records and assess whether recovery preserves retained utility.
- We evaluate six unlearning methods across two model sizes and two deletion ratios, showing how susceptibility varies across settings.

2 Related Work

Machine unlearning. Machine unlearning seeks to remove a training subset's influence without the cost of complete retraining (Cao & Yang, 2015; Ginart et al., 2019). Exact or certified approaches obtain efficiency through data partitioning, stability, or distributional guarantees (Bourtoule et al., 2021; Guo et al., 2020; Neel et al., 2021; Ullah et al., 2021; Sekhari et al., 2021). Approximate approaches instead use influence approximations, parameter scrubbing, distillation, or salient updates (Izzo et al., 2021; Golatkar et al., 2020; Kurmanji et al., 2023; Fan et al., 2024). These methods define deletion relative to retraining or a learning distribution. We study a complementary question: how an approximate post-unlearning state responds to subsequent targeted optimization.

Language-model unlearning. Language-model methods use negative gradients, alternative targets, preference objectives, logit adjustment, or representation interventions (Yao et al., 2023; Jang et al., 2023; Chen & Yang, 2023; Eldan & Russinovich, 2023; Zhang et al., 2024; Fan et al., 2025b; Dong et al., 2024). Other objectives add positive in-domain feedback or reweight examples (Mekala et al., 2025; Yang et al., 2025; Wang et al., 2025b). Benchmark suites measure forgotten behavior, privacy indicators, hazardous capabilities, and retained utility (Maini et al., 2024; Shi et al., 2024b; Li et al., 2024). These evaluations primarily characterize the state produced by unlearning. Our audit instead measures whether later exact-target updates produce excess recovery over matched alternative updates.

Membership, memorization, and deletion leakage. Membership inference exploits confidence or loss differences (Shokri et al., 2017; Yeom et al., 2018). Language-model memorization and extraction depend on duplication, scale, and prompt context (Carlini et al., 2021; 2023; Nasr et al., 2023); reference-free scores extend candidate detection, but remain sensitive to nonmember construction and distribution shift (Shi et al., 2024a; Mattern et al., 2023; Duan et al., 2024). Fine-tuning can generate stronger loss leakage (Zhang et al., 2025), and model updates can reveal deletion status (Chen et al., 2021). Our paired score is therefore evaluated first as deletion-target inference between known training members. Conventional single-state membership inference remains a separate task.

Recovery and federated anti-unlearning. Subsequent training can restore behavior suppressed by approximate unlearning (Hu et al., 2025), motivating smoothness-aware and invariance-based defenses (Fan et al., 2025a; Wang et al., 2025a). Federated unlearning introduces an additional integrity threat because authenticated clients can continue contributing updates (McMahan et al., 2017; Gao et al., 2022; Fraboni et al., 2024). Sheng et al. (2026) combine unlearning-induced leakage with reconstruction in a retaliatory federated attack. Our oracle setting is different: it grants exact target records to isolate recovery susceptibility and uses matched controls to separate that susceptibility from ordinary learning. We do not claim a new reconstruction mechanism or an evaluated aggregation bypass.

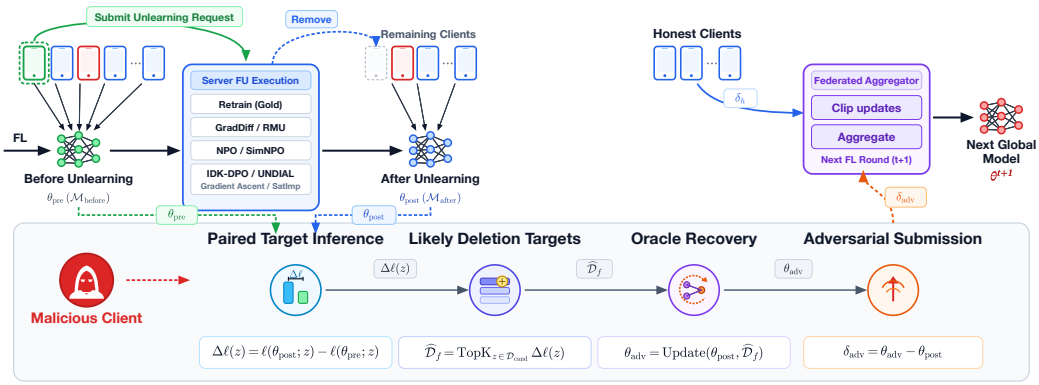

Figure 1: Proposed anti-unlearning workflow. After unlearning, a participating adversary scores candidate records and prepares a target-guided update for a later federated round. Paired target inference and exact-record local recovery are evaluated separately.

3 Methodology

3.1 Problem Formulation

Let $\mathcal{X}$ be the prompt space, $\mathcal{Y}$ the response space, and $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. For $z = (x, y) \in \mathcal{Z}$, let $p_\theta(y \mid x)$ denote the response distribution of an autoregressive language model. Here $\theta \in \Theta \subseteq \mathbb{R}^d$ is the vector of model weights. We define the record loss as

$$\ell(\theta; z) = -\frac{1}{|\mathcal{I}(z)|} \sum_{j \in \mathcal{I}(z)} \log p_\theta(y_j \mid x, y_{<j}), \tag{1}$$

where $\mathcal{I}(z)$ contains the supervised positions in the response y. Let $\mathcal{D}_{\text{ft}} = \mathcal{D}_f \dot{\cup} \mathcal{D}_r$, where $\mathcal{D}_f$ is requested for deletion and $\mathcal{D}_r$ is retained. Let $\mathcal{D}_h$ be disjoint from $\mathcal{D}_{\text{ft}}$, and let $\mathcal{D}_\phi$ contain deterministically generated fluent controls with no target content.

The pre-unlearning and post-unlearning model states are

$$\theta_{\text{pre}} \quad \text{and} \quad \theta_{\text{post}} = \mathcal{U}\left(\theta_{\text{pre}}, \mathcal{D}_f; \xi_u\right), \tag{2}$$

where $\mathcal{U}$ is an arbitrary unlearning procedure and ξ_u collects its randomness. After unlearning, the shared model remains trainable and federated training may resume. A participating client can initialize local training from θ_{post}, produce an adversarial local state θ_{adv}, and submit the update

$$\delta_{\text{adv}} = \theta_{\text{adv}} - \theta_{\text{post}}. \tag{3}$$

Our study asks whether such a client can find a target-aligned recovery direction after the server has completed unlearning.

3.2 Threat Model

We consider a federated learning system in which a requesting client asks the server to remove the influence of $\mathcal{D}_f$ from a shared model. The server applies $\mathcal{U}$, releases θ_{post}, and later resumes ordinary federated training. A different authenticated participant, denoted by A, remains eligible for subsequent rounds but opposes the deletion. Thus, our attacker is an authenticated but malicious participating client. The attacker neither issues the request nor controls the unlearning procedure. Figure 1 shows the proposed attack sequence from the request through the adversary's later local update.

Adversarial goal. The attack is a targeted integrity violation of the unlearned model. Given a recovery set $\mathcal{D}_A$, the adversary computes θ_{adv} and submits the corresponding client update δ_{adv}. Its goal is to restore a target capability functional M_f toward its pre-unlearning value without causing an obvious collapse of retained capability. We write the local objective as

$$\max_{\theta} \; M_f(\theta) \quad \text{s.t.} \quad V(\theta) \geq (1 - \kappa) V(\theta_{\text{pre}}), \qquad C(\theta; \theta_{\text{post}}) \leq B, \tag{4}$$

where V measures retained utility, $\kappa \in [0, 1)$ is the admissible fractional utility loss relative to the pre-unlearning state, and C represents the local training or update budget.

Knowledge and access. The adversary is a participating client with white-box access to the shared model states: it retains θ_{pre} and receives θ_{post} after unlearning. For deletion-target inference, it has a candidate pool $\mathcal{P}$ but does not know which records were targeted. In the separate oracle recovery setting, it also has the exact target records $\mathcal{D}_f$, independently of that inference. The inference result is not assumed to supply the records used for recovery.

Capabilities, timing, and budget. The adversary may inspect the model states legitimately distributed to participating clients, choose its local examples and objective, run ordinary local optimization, and return one authenticated client update in a later training round. Its resources are bounded by the number of accessible state evaluations, examples presented, optimizer steps, and local computation. It cannot alter the request, $\mathcal{U}$, other clients' data or gradients, the server's aggregation code, hidden evaluation records, or evaluation labels. All controlled recovery arms begin from the identical θ_{post}; the attacker has no privileged control over their initialization.

3.3 Controlled Oracle Recovery

Controlled Oracle Recovery is a two-part audit, summarized in Algorithm 1. The first part asks whether the unlearning transition distinguishes deletion targets from records that remain in training. The second asks whether training on the exact deletion set restores the forgotten behavior more strongly than training on other data. These parts are evaluated separately: recovery uses the exact deletion set, not the inferred candidate set. Before either part, a lineage check verifies that θ_{pre} and θ_{post} share compatible architecture, tokenizer, data partition, and evaluator definitions.

3.3.1 Paired deletion-target inference

For a candidate record $z \in \mathcal{P}$, the adversary computes the loss change induced by unlearning,

$$\Delta(z) = \ell(\theta_{\text{post}}; z) - \ell(\theta_{\text{pre}}; z). \tag{5}$$

The primary inference task distinguishes $z \sim D_f$ from $z \sim D_r$: both groups contributed to fine-tuning, but only D_f was targeted by unlearning. Comparisons with a disjoint holdout distribution P_h diagnose distribution shift and unrelated model drift. Given a threshold τ, the adversary returns

$$\widehat{\mathcal{D}}_f(\tau) = \{z \in \mathcal{P} : \Delta(z) \geq \tau\}. \tag{6}$$

This paired task uses the realized state transition and is distinct from single-state membership inference attacks in (Shokri et al., 2017; Yeom et al., 2018; Duan et al., 2024).

3.3.2 Exact-target recovery intervention

The preceding test ranks likely deletion targets; its predictions are not used for recovery. In this separate test, the adversary has the deletion set $\mathcal{D}_f$. It trains the post-unlearning model on these records to reduce their average supervised loss:

$$L_f(\theta) = \frac{1}{|\mathcal{D}_f|} \sum_{z \in \mathcal{D}_f} \ell(\theta; z). \tag{7}$$

Why comparisons are necessary. To determine whether restoration depends on the deletion records, we repeat the update using retained records $\mathcal{D}_r$, holdout records $\mathcal{D}_h$, or fluent text $\mathcal{D}_\phi$. Each comparison begins from θ_{post} and uses the same optimizer and number of steps; the training data are what change. We also evaluate θ_{post} without an update. A larger recovery from $\mathcal{D}_f$ than from these alternatives identifies the contribution of the deletion records rather than training alone.

3.3.3 Recovery and utility measures

Let M be a behavioral endpoint and choose $q_M \in \{-1, +1\}$ so that $\widetilde{M}(\theta) = q_M M(\theta)$ increases with target capability. We normalize the target transition as

$$R^M_{a,s}(b) = \frac{\widetilde{M}(\theta_{a,b,s}) - \widetilde{M}(\theta_{\text{post}})}{\widetilde{M}(\theta_{\text{pre}}) - \widetilde{M}(\theta_{\text{post}})}. \tag{8}$$

Algorithm 1 Paired Deletion-Target Inference and Controlled Oracle Recovery

Require: Paired input: states $\theta_{\mathrm{pre}}, \theta_{\mathrm{post}}$ and unlabeled candidate pool $\mathcal{P}$
Require: Recovery input: oracle set $\mathcal{D}_f$ and control sets $\mathcal{D}_r, \mathcal{D}_h, \mathcal{D}_\phi$
Require: Evaluator-only group labels; budgets $\mathcal{B}$; seeds $\mathcal{S}$; endpoints $\mathcal{M}$
Verify model, tokenizer, data, and evaluator lineage
for all $z \in \mathcal{P}$ **do**
    Compute $\Delta(z)$
end for
Rank candidates by $\Delta(z)$ and form $\widehat{\mathcal{D}}_f(\tau)$
Evaluate the ranking using group labels withheld from the adversary
Independently provide $\mathcal{D}_f$ to the oracle recovery arm
for all $a \in \{f, r, h, \phi\}$, $b \in \mathcal{B}$, $s \in \mathcal{S}$ **do**
    Reset model and optimizer to θ_{post}
    Train on $\mathcal{D}_a$ for exactly b optimizer steps
    Evaluate $M(\theta_{a,b,s})$ for every $M \in \mathcal{M}$
    Compute $R^M_{a,s}(b)$
end for
Evaluate the no-update, retain-only, and base-state references available
Compute every prespecified $\Delta R^M_c(b)$ and retained utility

Here, $\theta_{a,b,s}$ denotes the model after b updates on $\mathcal{D}_a$ in repeated run s, where $a \in \{f, r, h, \phi\}$. The no-update state is $\theta_{0,b,s} = \theta_{\mathrm{post}}$. $R = 0$ is the post-unlearning level and $R = 1$ is the pre-unlearning level; overshoot and deterioration are not clipped. For each prespecified control $c \in \mathcal{C} = \{r, h, \phi\}$, control-adjusted recovery is

$$\Delta R^M_c(b) = \overline{R}^M_f(b) - \overline{R}^M_c(b), \tag{9}$$

where both means use the same recovery-seed set. Let V denote retained or general utility. A selective recovery event must satisfy both a positive target–control contrast and

$$\overline{V}_f(b) \geq (1 - \kappa) V(\theta_{\mathrm{pre}}), \tag{10}$$

for a stated fractional tolerance κ.

3.4 Theoretical Analysis

The paired score has an exact path interpretation. For any differentiable path $\vartheta : [0, 1] \to \Theta$ with $\vartheta(0) = \theta_{\mathrm{pre}}$ and $\vartheta(1) = \theta_{\mathrm{post}}$,

$$\Delta(z) = \int_0^1 \langle \nabla_\theta \ell(\vartheta(t); z), \vartheta'(t) \rangle \, \mathrm{d}t. \tag{11}$$

Thus the score measures record-wise alignment with the realized state transition, not a static membership signal.

For recovery, let $F(\theta) = L_f(\theta)$ denote the forget set loss and assume the initial forget gap $G_F = F(\theta_{\mathrm{post}}) - F(\theta_{\mathrm{pre}}) > 0$. At $\theta = \theta_{\mathrm{post}}$, consider matched updates $\theta^+_f = \theta - \eta d_f$ and $\theta^+_c = \theta - \eta d_c$.

Proposition 1 (One-step target-control separation)**.** *Assume that F is differentiable and β_F-smooth on a convex neighborhood containing θ, θ^+_f, and θ^+_c. Then*

$$\begin{aligned} R^F_f - R^F_c = \frac{F(\theta^+_c) - F(\theta^+_f)}{G_F} \geq \frac{\eta}{G_F} \langle \nabla F(\theta), d_f - d_c \rangle \\ - \frac{\beta_F \eta^2}{2 G_F} \left(\|d_f\|_2^2 + \|d_c\|_2^2 \right), \end{aligned} \tag{12}$$

where R^F orients lower forget loss as greater recovery.

Proof sketch. Smoothness upper-bounds the exact-target step and lower-bounds the control step by their first-order terms plus opposite quadratic remainders. Subtracting the two bounds and dividing

by $G_F > 0$ gives Equation 12. Appendix A provides the complete derivation and states the boundary of the result. The proposition gives a sufficient condition for a positive recovery contrast. If the exact target direction aligns with the forget loss gradient more strongly than the control direction by an amount exceeding the curvature term, it produces greater normalized recovery of the forget loss. This guarantee applies to one update step under the stated smoothness assumption. It does not guarantee multi-step optimizer behavior, text-generation overlap, global federated influence, certified deletion failure, or persistence of a specific latent representation.

4 EXPERIMENTS

4.1 EXPERIMENTAL QUESTIONS

The evaluation answers four questions: **RQ1**, does the pre/post state transition identify records targeted by unlearning among known training members? **RQ2**, do exact-target updates restore forgotten behavior more strongly than compute-matched alternative updates? **RQ3**, can this recovery occur while retaining general model utility? **RQ4**, how does susceptibility vary across unlearning methods, model scale, and forget ratio?

4.2 EXPERIMENTAL SETUP

Models and datasets. We evaluate Llama-3.2-1B-Instruct and Llama-3.1-8B-Instruct on both TOFU-05 and TOFU-10. TOFU-05 removes 5% of the fictitious author records and retains 95%; TOFU-10 removes 10% and retains 90% (Maini et al., 2024). WMDP Bio paired inference is reported only in the appendix (Li et al., 2024).

Unlearning methods and recovery arms. We test GradDiff (Maini et al., 2024), NPO (Zhang et al., 2024), SimNPO (Fan et al., 2025b), IDK-DPO as implemented in OpenUnlearning (Dorna et al., 2025), UNDIAL (Dong et al., 2024), and RMU (Li et al., 2024). Recovery on the exact forget data is compared with equally budgeted updates on retain, holdout, and fluent text.

Evaluation metrics and uncertainty. *Deletion-target inference (RQ1).* Using the paired score $\Delta(z)$ (Eq. 5) to separate $\mathcal{D}_f$ from $\mathcal{D}_r$, we report ROC-AUC, Average Precision (AP) at positive prevalence $\pi = |\mathcal{D}_f|/(|\mathcal{D}_f| + |\mathcal{D}_r|)$, TPR at 1% FPR ($\mathrm{TPR}_{1\%}$), and membership advantage ($\mathrm{Adv} = \max_\tau(\mathrm{TPR}(\tau) - \mathrm{FPR}(\tau))$). Metrics are test-set means over five stratified calibration/test splits (50/50 partition), with 95% confidence intervals from 1,000 bootstrap resamples.

Controlled recovery (RQ2–RQ4). We measure behavioral restoration on forget-answer probability ($\mathrm{Prob.}$) and ROUGE-L using normalized recovery R (Eq. 8). To confirm record-specific susceptibility, we evaluate the control-adjusted advantage $\Delta R = \overline{R}_{\mathrm{oracle}} - \max_{c \in \mathcal{C}} \overline{R}_c$ (Eq. 9) relative to compute-matched retain, holdout, and fluent controls $\mathcal{C}$. Recovery entries report the mean $\pm$ standard deviation of paired differences across three recovery seeds.

4.3 RQ1: DELETION-TARGET INFERENCE

Deletion targets are visible when unlearning changes their loss differently from the loss of retained training records. Table 1 compares these two groups across all four settings. Because both appeared in the original training data, the task is to identify deletion targets, not to distinguish training members from nonmembers.

The paired adversary identifies deletion targets with high accuracy for several unlearning transitions. Table 1 gives ROC-AUC of at least 0.999 and TPR at 1% FPR of at least 0.981 for NPO and SimNPO in every setting. At this low false positive rate, the adversary can form a highly concentrated candidate target set from known training records. The score succeeds when target losses change consistently relative to retained records, rather than merely reflecting that both groups were once used for training. GradDiff has lower ROC-AUC, from 0.809 to 0.883, and IDK-DPO has ROC-AUC from 0.643 to 0.736. IDK-DPO detects at most 15.7% of targets at 1% FPR. The pattern is consistent with greater overlap between forget and retain loss changes for these methods. Aggregate ranking metrics do not identify the underlying parameter changes.

Table 1: Deletion target inference on TOFU across model sizes and forget splits. AP denotes average precision (PR-AUC), TPR is measured at 1% FPR, and Adv. denotes membership advantage. Values are means over five calibration and test splits.

	Llama-3.2-1B (TOFU-05)				**Llama-3.2-1B (TOFU-10)**				**Llama-3.1-8B (TOFU-05)**				**Llama-3.1-8B (TOFU-10)**			
Method	ROC	AP	TPR	Adv.	ROC	AP	TPR	Adv.	ROC	AP	TPR	Adv.	ROC	AP	TPR	Adv.
Gold retrain	.999	.998	.986	.986	1.000	1.000	.992	.989	.998	.997	.991	.988	1.000	1.000	1.000	.996
GradDiff	.866	.809	.323	.553	.883	.890	.346	.589	.809	.753	.356	.470	.858	.889	.488	.601
IDK-DPO	.733	.622	.157	.350	.736	.697	.059	.370	.643	.421	.000	.257	.645	.598	.019	.252
NPO	1.000	1.000	1.000	1.000	1.000	1.000	1.000	.998	1.000	1.000	1.000	1.000	.999	.999	.981	.984
RMU	.985	.947	.369	.944	.453	.468	.008	.005	1.000	1.000	1.000	1.000	1.000	1.000	1.000	.999
SimNPO	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
UNDIAL	.425	.280	.000	.076	.512	.461	.000	.211	.187	.214	.000	.008	.260	.362	.000	.028

Table 2: Control-adjusted exact-data recovery advantage (ΔR, Eq. equation 9) after 100 recovery steps on TOFU. Entries report the mean $\pm$ one standard deviation of paired differences across three recovery seeds. Positive values ($\Delta R > 0$) establish that exact-target updates restore forgotten behavior strictly more effectively than any compute-matched control.

	TOFU-05		TOFU-10	
Method	$\Delta R_{\text{Prob.}}$	ΔR_{ROUGE}	$\Delta R_{\text{Prob.}}$	ΔR_{ROUGE}
Llama-3.2-1B-Instruct				
GradDiff	2.160 ± 0.040	2.381 ± 0.059	1.308 ± 0.015	1.637 ± 0.021
NPO	1.166 ± 0.005	1.356 ± 0.005	1.044 ± 0.002	1.186 ± 0.011
SimNPO	1.014 ± 0.003	1.015 ± 0.009	0.664 ± 0.003	0.327 ± 0.004
IDK-DPO	2.315 ± 0.017	2.372 ± 0.065	2.838 ± 0.008	1.852 ± 0.032
UNDIAL	1.100 ± 0.002	1.240 ± 0.011	0.647 ± 0.005	0.400 ± 0.015
RMU	0.844 ± 0.009	0.556 ± 0.024	0.182 ± 0.001	0.444 ± 0.007
Llama-3.1-8B-Instruct				
GradDiff	0.549 ± 0.027	0.816 ± 0.022	0.468 ± 0.052	0.652 ± 0.051
NPO	0.997 ± 0.008	0.984 ± 0.007	0.981 ± 0.014	0.955 ± 0.014
SimNPO	1.007 ± 0.002	0.998 ± 0.014	0.973 ± 0.003	0.919 ± 0.003
IDK-DPO	0.871 ± 0.016	0.794 ± 0.042	1.144 ± 0.108	0.958 ± 0.095
UNDIAL	0.869 ± 0.003	0.951 ± 0.008	0.902 ± 0.002	0.931 ± 0.012
RMU	1.000 ± 0.007	0.993 ± 0.027	0.993 ± 0.003	0.943 ± 0.019

The score direction and the evaluated state pair also matter. RMU yields ROC-AUC of at least 0.985 in three settings but 0.453 on 1B TOFU-10. UNDIAL falls below chance in three settings under the prescribed loss-change direction, indicating an opposing ranking rather than necessarily an absence of target information. Gold retraining is highly separable as well, so target visibility alone does not establish ineffective deletion. The holdout diagnostics in Appendix C further show why the comparison population matters: on 1B TOFU-05, GradDiff attains ROC-AUC 0.866 against retain records but 0.183 against unseen holdout records (Table 3). The primary forget versus retain task holds original training membership constant; the holdout task does not.

4.4 RQ2–RQ4: Recovery Specificity, Utility, and Scale

Across four experiments, updates on the exact deletion records restore forgotten behavior more than updates on alternative data trained for the same number of steps. The size and timing of this advantage vary across unlearning methods, as does the utility of the recovered model. Table 2 reports the final comparisons, and Figure 2 shows how recovery develops over the update steps.

Controlled Oracle Recovery demonstrates a consistent advantage for exact target data over matched alternative data. At 100 steps, all 48 probability and ROUGE contrasts in Table 2 are positive. Each comparison starts from the same post-unlearning state and matches the optimizer budget and recovery seeds. Because the alternative updates receive the same number of training steps but recover less, the additional improvement is tied to the deletion records rather than to continued training alone. The exact records supply direct supervision for those answers, whereas the controls do not. The trajectories in Figure 2 show that the advantage can emerge at different budgets. On 8B TOFU-05,

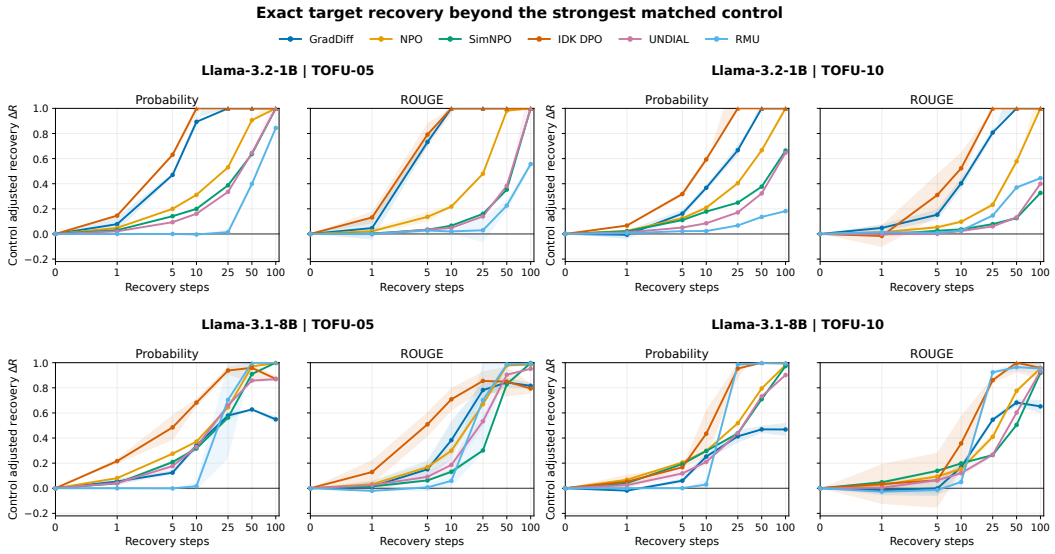

Figure 2: Control adjusted recovery across the four evaluated settings. Each curve subtracts the strongest matched control from exact target recovery. Positive values favor the deletion targets. Shading shows one standard deviation over three recovery seeds.

IDK-DPO reaches a probability contrast of 0.938 by 25 steps, while RMU remains near zero through 10 steps and reaches 0.704 at 25. The curves are not universally monotone: the 8B TOFU-05 GradDiff contrast decreases from 0.627 at 50 steps to 0.549 at 100. This identifies a stronger intervention at an intermediate budget and shows why an attacker can benefit from choosing when to stop.

The size of ΔR depends on both the additional answer probability gained through recovery and how much probability unlearning removed initially. When the initial drop is small, even a modest raw gain can produce a large normalized value. On 1B TOFU-05, GradDiff loses only 0.081 in forgotten-answer probability after unlearning. Its raw oracle advantage of 0.175 consequently becomes $\Delta R = 2.160$. NPO begins with a larger gap of 0.533, then gains 0.621 over the strongest control in raw probability, yielding $\Delta R = 1.166$. The smaller normalized value thus corresponds to recovery after substantially more initial forgetting. The same denominator effect appears for 8B TOFU-10 IDK-DPO: a raw advantage of 0.031 over a gap of 0.027 yields $\Delta R = 1.144$. Conversely, on 1B TOFU-10, RMU's oracle arm passes $R = 1$, but its contrast is only 0.182 because a matched control also restores much of the score. Reaching the pre-unlearning reference and recovering specifically through target data are distinct outcomes. The 8B contrast profiles are broadly similar across forget ratios, but the tested models belong to different releases and have different initial gaps, so these data do not isolate a causal effect of scale.

The attack can restore forgotten behavior without sacrificing retained utility. The probability frontiers in Appendix E show that two methods meet both $R \geq 1$ and the 95% retained utility threshold on 1B TOFU-10 and 8B TOFU-05, and three do so on 8B TOFU-10 at 100 steps. In particular, 8B TOFU-10 RMU reaches $R = 1.005$ with 101.9% of pre-unlearning utility, whereas 8B TOFU-05 SimNPO reaches $R = 1.008$ with only 74.6%. Exact target optimization directly rewards forgotten answers but does not by itself constrain retained behavior. On 1B TOFU-05, recovery contrasts remain positive, but no final state meets both thresholds. Reporting both outcomes identifies when the local intervention restores the target behavior while maintaining model utility.

5 Discussion and Limitations

What the results establish. The paired inference results in Table 1 show that deletion targets can often be ranked from the pre-to-post loss change, but the signal strength and direction depend on the unlearning method. The recovery results in Table 2 and Figure 2 show a second weakness: exact-target updates restore forgotten behavior more than matched alternative-data updates.

Implications for federated unlearning. Federated unlearning is commonly judged when the server returns the post-unlearning model. Our results show that this state can remain vulnerable to later client updates. A participating adversary can use the same local training interface as an honest client, optimize on identified deletion targets, and produce an update that restores removed behavior while preserving useful performance in several settings. Deletion should therefore be evaluated as a continuing integrity property rather than as a one-time change in model output.

Method and model dependence. Susceptibility varies substantially across unlearning methods and evaluated model states. This variation motivates reporting recovery together with retained utility rather than treating unlearning as a binary success or failure.

Limitations. The recovery audit assumes access to target records. Recovery seeds are conditioned on fixed post-unlearning states, so they do not measure variation across independently trained states. Behavioral recovery also does not by itself identify the internal representation that enabled recovery.

6 Conclusion

Federated unlearning should be evaluated beyond the immediate post-unlearning state, especially when the model remains available for subsequent training. This paper presented Controlled Oracle Recovery, which provides a test by combining deletion-target inference with matched recovery updates. Across the evaluated TOFU settings, the paired loss signal often identifies known deletion targets, and exact-target updates have a positive advantage over matched controls in all 48 step-100 comparisons. Useful recovery remains method- and setting-dependent as retained utility does not necessarily follow forgotten-answer recovery. These results establish a measurable local anti-unlearning vulnerability. Its effect on the shared model after aggregation and later federated rounds remains to be tested.

Reproducibility Statement

The accompanying repository provides the implementation of paired deletion target inference and Controlled Oracle Recovery, together with experiment configurations, evaluation utilities, statistical analysis, plotting scripts, tests, and cluster launchers. Each reported experiment records the model and data revisions, unlearning method, recovery dataset, optimizer settings, update budget, random seeds, metric definitions, and uncertainty estimates.

Ethics Statement

The work studies an attack surface of machine unlearning and could be misused to reverse legitimate deletion requests. Experiments are restricted to public or synthetic benchmark records. We release controls and evaluation code to support defensive auditing, but the methodology does not authorize recovery of personal, copyrighted, or safety-critical data from deployed systems.

AI Use Statement

Generative AI tools were used solely for language editing and mathematical typesetting. The authors verified the experimental artifacts, mathematical statements, citations, and final claims and take responsibility for the submission.

References

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy*, pp. 141–159, 2021. doi: 10.1109/SP40001.2021.00019.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy*, pp. 463–480, 2015. doi: 10.1109/SP.2015.35.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium*, pp. 2633–2650, 2021. URL https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.

Nicholas Carlini, Katherine Lee, Daphne Ippolito, Florian Tramèr, Matthew Jagielski, and Chiyuan Zhang. Quantifying memorization across neural language models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=TatRHT_1cK.

Jiaao Chen and Diyi Yang. Unlearn what you want to forget: Efficient unlearning for LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12041–12052, 2023. doi: 10.18653/v1/2023.emnlp-main.738. URL https://aclanthology.org/2023.emnlp-main.738/.

Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When machine unlearning jeopardizes privacy. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 896–911, 2021.

Yijiang River Dong, Hongzhou Lin, Mikhail Belkin, Ramon Huerta, and Ivan Vulić. UnDIAL: Self-distillation with adjusted logits for robust unlearning in large language models. *arXiv preprint arXiv:2402.10052*, 2024. URL https://arxiv.org/abs/2402.10052.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C. Lipton, J. Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. *arXiv preprint arXiv:2506.12618*, 2025. URL https://arxiv.org/abs/2506.12618.

Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do membership inference attacks work on large language models? In *Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2402.07841.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in LLMs. *arXiv preprint arXiv:2310.02238*, 2023. URL https://arxiv.org/abs/2310.02238.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. SalUn: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/ec4d2e436794d1bf55ca83f5ebb31887-Abstract-Conference.html.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. Towards LLM unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 15762–15778, 2025a. URL https://openreview.net/forum?id=zZjLv6F0Ks.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems*, 2025b. URL https://arxiv.org/abs/2410.07163.

Yann Fraboni, Martin Van Waerebeke, Kevin Scaman, Richard Vidal, Laetitia Kameni, and Marco Lorenzi. SIFU: Sequential informed federated unlearning for efficient and provable client unlearning in federated optimization. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 3457–3465, 2024. URL https://proceedings.mlr.press/v238/fraboni24a.html.

Xiangshan Gao, Xingjun Ma, Jingyi Wang, Youcheng Sun, Bo Li, Shouling Ji, Peng Cheng, and Jiming Chen. VeriFi: Towards verifiable federated unlearning. *arXiv preprint arXiv:2205.12709*, 2022. URL https://arxiv.org/abs/2205.12709.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Zou. Making AI forget you: Data deletion in machine learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/cb79f8fa58b91d3af6c9c991f63962d3-Abstract.html.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9304–9312, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Golatkar_Eternal_Sunshine_of_the_Spotless_Net_Selective_Forgetting_in_Deep_CVPR_2020_paper.html.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3832–3842, 2020. URL https://proceedings.mlr.press/v119/guo20c.html.

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. URL https://arxiv.org/abs/2406.13356.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. Approximate data deletion from machine learning models. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2008–2016, 2021. URL https://proceedings.mlr.press/v130/izzo21a.html.

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*, pp. 14389–14408, 2023. doi: 10.18653/v1/2023.acl-long.805. URL https://aclanthology.org/2023.acl-long.805/.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/062d711fb777322e2152435459e6e9d9-Abstract.html.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 2024. URL https://arxiv.org/abs/2403.03218.

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

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. Membership inference attacks against language models via neighbourhood comparison. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 11330–11343, 2023. doi: 10.18653/v1/2023.findings-acl.719. URL https://aclanthology.org/2023.findings-acl.719/.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017. URL https://proceedings.mlr.press/v54/mcmahan17a.html.

Anmol Mekala, Vineeth Dorna, Shreya Dubey, Abhishek Lalwani, David Koleczek, Mukund Rungta, Sadid Hasan, and Elita Lobo. Alternate preference optimization for unlearning factual knowledge in large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 3732–3752, 2025. URL https://aclanthology.org/2025.coling-main.252/.

Milad Nasr, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Eric Wallace, Florian Tramèr, and Katherine Lee. Scalable extraction of training data from (production) language models. *arXiv preprint arXiv:2311.17035*, 2023. URL https://arxiv.org/abs/2311.17035.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. Descent-to-delete: Gradient-based methods for machine unlearning. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory*, volume 132 of *Proceedings of Machine Learning Research*, pp. 931–962, 2021. URL https://proceedings.mlr.press/v132/neel21a.html.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/9627c45df543c816a3ddf2d8ea686a99-Abstract.html.

Xinyi Sheng, Wei Bao, Hequn Wang, Yuqin Liu, and Sen Fu. Retaliatory attacks against federated unlearning via data leakage. In *The Fortieth AAAI Conference on Artificial Intelligence*, pp. 25321–25329, 2026.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *International Conference on Learning Representations*, 2024a. URL https://arxiv.org/abs/2310.16789.

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. *arXiv preprint arXiv:2407.06460*, 2024b. URL https://arxiv.org/abs/2407.06460.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy*, pp. 3–18, 2017.

Enayat Ullah, Tung Mai, Anup Rao, Ryan A. Rossi, and Raman Arora. Machine unlearning via algorithmic stability. In *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 4126–4142, 2021. URL https://proceedings.mlr.press/v134/ullah21a.html.

Changsheng Wang, Yihua Zhang, Jinghan Jia, Parikshit Ram, Dennis Wei, Yuguang Yao, Soumyadeep Pal, Nathalie Baracaldo, and Sijia Liu. Invariance makes LLM unlearning resilient even to unanticipated downstream fine-tuning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 65464–65479. PMLR, 2025a. URL https://proceedings.mlr.press/v267/wang25en.html.

Qizhou Wang, Jin Peng Zhou, Zhanke Zhou, Saebyeol Shin, Bo Han, and Kilian Q. Weinberger. Rethinking LLM unlearning objectives: A gradient perspective and go beyond. In *International Conference on Learning Representations*, 2025b. URL https://arxiv.org/abs/2502.19301.

Puning Yang, Qizhou Wang, Zhuo Huang, Tongliang Liu, Chengqi Zhang, and Bo Han. Exploring criteria of loss reweighting to enhance LLM unlearning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, 2025. URL https://openreview.net/forum?id=mGOugCZlAq.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. *arXiv preprint arXiv:2310.10683*, 2023. URL https://arxiv.org/abs/2310.10683.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium*, pp. 268–282, 2018.

Kaiyuan Zhang, Siyuan Cheng, Hanxi Guo, Yuetian Chen, Zian Su, Shengwei An, Yuntao Du, Charles Fleming, Ashish Kundu, Xiangyu Zhang, and Ninghui Li. SOFT: Selective data obfuscation for protecting LLM fine-tuning against membership inference attacks. In *34th USENIX Security Symposium*, pp. 8135–8154, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/zhang-kaiyuan.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.05868.

A Proof of Proposition 1

Proof. For a differentiable β_F-smooth function, the two-sided descent inequality gives

$$F(\theta_f^+) \leq F(\theta) - \eta\langle \nabla F(\theta), d_f\rangle + \frac{\beta_F \eta^2}{2}\|d_f\|_2^2, \tag{13}$$

$$F(\theta_c^+) \geq F(\theta) - \eta\langle \nabla F(\theta), d_c\rangle - \frac{\beta_F \eta^2}{2}\|d_c\|_2^2. \tag{14}$$

Subtracting Eq. equation 13 from Eq. equation 14 yields

$$F(\theta_c^+) - F(\theta_f^+) \geq \eta\langle \nabla F(\theta), d_f - d_c\rangle - \frac{\beta_F \eta^2}{2}\left(\|d_f\|_2^2 + \|d_c\|_2^2\right).$$

Division by $G_F > 0$ proves Eq. equation 12. The argument is conditional on the realized directions d_f, d_c and applies within the local smoothness neighborhood. It supplies neither a multi-step convergence guarantee nor a causal-memory guarantee. □

B Additional Experimental Details

Paired deletion-target inference. Each report uses five stratified calibration and test partitions that reuse examples and therefore measure split sensitivity for one fixed pair of model states. The original Llama-3.2-1B-Instruct TOFU-10 report contains 400 forget, 3,600 retain, and 400 holdout examples. Its split level intervals use 1,000 stratified bootstrap resamples distributed over five bootstrap random seeds. The empirical score is the response loss difference in Eq. equation 5; single state loss attacks remain separate membership baselines.

Controlled recovery. The default step grid is $\{1, 5, 10, 25, 50, 100\}$, with recovery seeds $\{0, 1, 2\}$, microbatch size four, gradient accumulation four, constant learning rate 10^{-5}, and zero weight decay. Consequently, one optimizer step presents 16 examples, although supervised-token exposure varies with response length. Every budget and arm restarts independently from the same post-unlearning model state with fresh optimizer state. Exact normalized-string hashes audit question, answer, and pair overlap between the target data and controls; this check does not detect semantic paraphrases.

Uncertainty boundary. The implemented recovery bootstrap resamples aligned evaluation identifiers and returns per-arm percentile intervals for R. The current runner reports ΔR_c as a difference of seed means but does not yet construct paired or simultaneous confidence intervals for those contrasts. Confirmatory inference should resample the target and control arms jointly and nest recovery seeds within independent unlearning seeds.

C Additional Paired State Inference Diagnostics

The main paper uses forget versus retain as the deletion target task. The following tables report comparisons with holdout examples. These values are diagnostics for distribution shift and should not be interpreted as repeated estimates of the same attack. AP denotes average precision and TPR is measured at 1% FPR. Each entry is a mean over five stratified calibration and test partitions of one fixed pair of model states. The TOFU diagnostics are given in Tables 3 and 4; the WMDP Bio diagnostic is in Table **??**.

D Supplementary Recovery Visualizations

These figures provide additional views of the recovery results. The upper panels show normalized recovery, and the lower panels show target-minus-control differences after 100 steps. Bands and error bars denote standard deviations over recovery seeds. The four model and dataset settings are shown in Figures 3, 4, 5, and 6.

Table 3: TOFU diagnostics for Llama-3.2-1B-Instruct. The first group treats forget examples as positive and holdout examples as negative. The second treats retain examples as positive and holdout examples as negative.

		Forget versus holdout				Retain versus holdout			
Split	Method	ROC-AUC	AP	TPR@1%	Adv.	ROC-AUC	AP	TPR@1%	Adv.
TOFU-05	Gold retrain	.998	.998	.986	.986	.534	.647	.003	.195
	GradDiff	.183	.387	.046	.044	.019	.457	.000	.000
	IDK-DPO	.419	.494	.049	.147	.235	.508	.000	.036
	NPO	.999	.999	.969	.966	.112	.475	.000	.000
	RMU	1.000	1.000	1.000	.996	.916	.952	.371	.707
	SimNPO	1.000	1.000	1.000	1.000	.962	.976	.606	.794
	UNDIAL	.998	.998	.960	.960	.990	.995	.893	.899
TOFU-10	Gold retrain	.999	.999	.989	.983	.466	.864	.002	.126
	GradDiff	.187	.377	.024	.024	.008	.746	.000	.000
	IDK-DPO	.151	.343	.011	.017	.057	.756	.000	.000
	NPO	.972	.972	.629	.824	.060	.756	.000	.003
	RMU	.884	.907	.426	.655	.872	.984	.378	.617
	SimNPO	1.000	1.000	1.000	1.000	.866	.980	.197	.572
	UNDIAL	.994	.994	.871	.924	.962	.995	.611	.782

Table 4: TOFU diagnostics for Llama-3.1-8B-Instruct. Columns follow Table 3.

		Forget versus holdout				Retain versus holdout			
Split	Method	ROC-AUC	AP	TPR@1%	Adv.	ROC-AUC	AP	TPR@1%	Adv.
TOFU-05	Gold retrain	.991	.994	.980	.974	.357	.547	.000	.281
	GradDiff	.192	.423	.083	.079	.002	.452	.000	.000
	IDK-DPO	.869	.769	.021	.796	.851	.852	.039	.776
	NPO	.980	.975	.569	.886	.326	.535	.000	.217
	RMU	1.000	1.000	1.000	1.000	.676	.690	.006	.621
	SimNPO	1.000	1.000	1.000	1.000	.887	.892	.042	.761
	UNDIAL	.983	.979	.579	.913	.994	.997	.862	.916
TOFU-10	Gold retrain	.996	.997	.983	.981	.514	.457	.000	.456
	GradDiff	.175	.401	.061	.056	.001	.308	.000	.000
	IDK-DPO	.612	.516	.008	.516	.582	.491	.000	.469
	NPO	.980	.979	.678	.861	.361	.393	.000	.277
	RMU	1.000	1.000	1.000	1.000	.652	.531	.004	.605
	SimNPO	1.000	1.000	1.000	1.000	.942	.912	.218	.836
	UNDIAL	.978	.959	.215	.896	.986	.984	.659	.886

E Recovery and Utility Frontiers

The following figures show normalized recovery against retained utility over the recovery trajectory. Point labels denote recovery steps. The four frontiers are shown in Figures 7, 8, 9, and 10.

Table 5: Supplementary paired state inference on WMDP Bio with **Llama-3.1-8B-Instruct**. Forget versus retain is the primary deletion target comparison. The two holdout comparisons diagnose distribution shift. AP denotes average precision and TPR is measured at 1% FPR.

Method	Positive versus negative	Prev.	ROC-AUC	AP	TPR@1%	Adv.
Gradient Ascent	Forget versus retain	.10	.808	.762	.026	.496
	Forget versus holdout	.50	.825	.779	.016	.538
	Retain versus holdout	.90	.513	.524	.009	.067
GradDiff	Forget versus retain	.10	1.000	1.000	1.000	.997
	Forget versus holdout	.50	1.000	1.000	1.000	.997
	Retain versus holdout	.90	.498	.492	.002	.051
NPO	Forget versus retain	.10	1.000	1.000	.994	.993
	Forget versus holdout	.50	1.000	1.000	.994	.993
	Retain versus holdout	.90	.423	.457	.006	.014
RMU	Forget versus retain	.10	.999	.999	.996	.991
	Forget versus holdout	.50	1.000	1.000	.989	.983
	Retain versus holdout	.90	.509	.491	.004	.082
SatImp	Forget versus retain	.10	.999	.999	.983	.981
	Forget versus holdout	.50	.999	.999	.993	.989
	Retain versus holdout	.90	.437	.475	.019	.016
SimNPO	Forget versus retain	.10	.995	.995	.876	.939
	Forget versus holdout	.50	.994	.991	.925	.943
	Retain versus holdout	.90	.449	.474	.020	.016

Table 6: Supplementary paired state inference on WMDP Bio with **Llama-3.2-1B-Instruct**. Forget versus retain is the primary deletion target comparison. The two holdout comparisons diagnose distribution shift. AP denotes average precision and TPR is measured at 1% FPR.

Method	Positive versus negative	Prev.	ROC-AUC	AP	TPR@1%	Adv.
Gradient Ascent	Forget versus retain	.10	.738	.661	.000	.405
	Forget versus holdout	.50	.684	.588	.000	.354
	Retain versus holdout	.90	.457	.465	.006	.006
GradDiff	Forget versus retain	.10	1.000	1.000	.996	.992
	Forget versus holdout	.50	.999	1.000	.995	.994
	Retain versus holdout	.90	.469	.472	.004	.041
NPO	Forget versus retain	.10	.999	.999	.981	.979
	Forget versus holdout	.50	.999	.999	.987	.981
	Retain versus holdout	.90	.405	.448	.012	.008
RMU	Forget versus retain	.10	.999	.999	.989	.985
	Forget versus holdout	.50	.999	.999	.980	.977
	Retain versus holdout	.90	.535	.520	.005	.081
SatImp	Forget versus retain	.10	.998	.998	.979	.974
	Forget versus holdout	.50	.997	.997	.979	.976
	Retain versus holdout	.90	.401	.449	.013	.009
SimNPO	Forget versus retain	.10	.991	.991	.836	.911
	Forget versus holdout	.50	.992	.992	.852	.915
	Retain versus holdout	.90	.414	.464	.011	.015

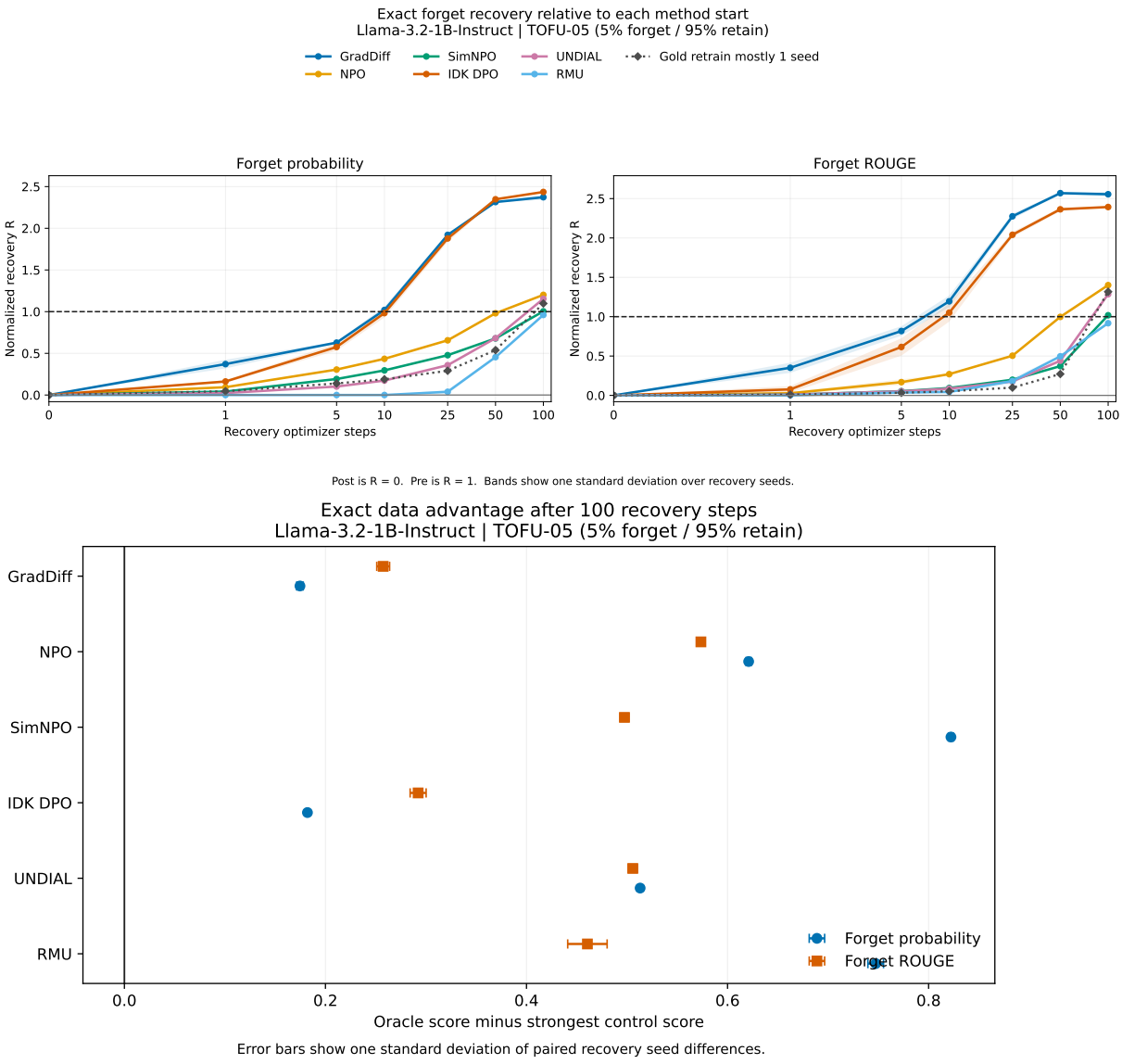

Figure 3: Additional Llama-3.2-1B-Instruct results on TOFU-05. Top: normalized oracle recovery, where $R = 0$ is the post-unlearning state and $R = 1$ is the pre-unlearning reference. Bottom: the paired exact-data advantage at 100 recovery steps. Bands and error bars show one standard deviation over three recovery seeds.

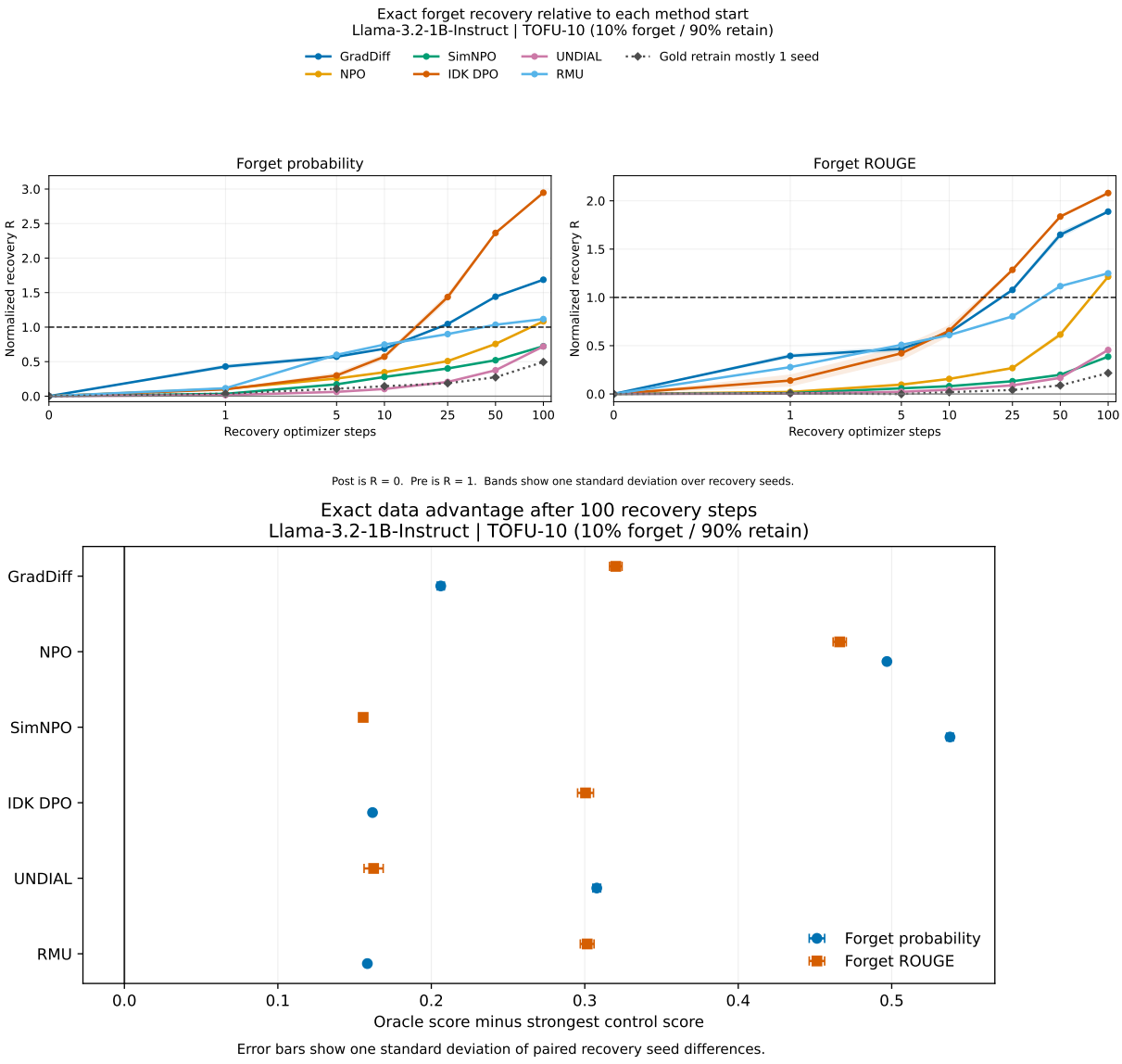

Figure 4: Additional Llama-3.2-1B-Instruct results on TOFU-10. The normalized trajectories (top) measure movement from the post-unlearning state toward the pre-unlearning reference; the endpoint comparison (bottom) subtracts the strongest matched control separately for each method and metric. Uncertainty denotes one standard deviation over three recovery seeds.

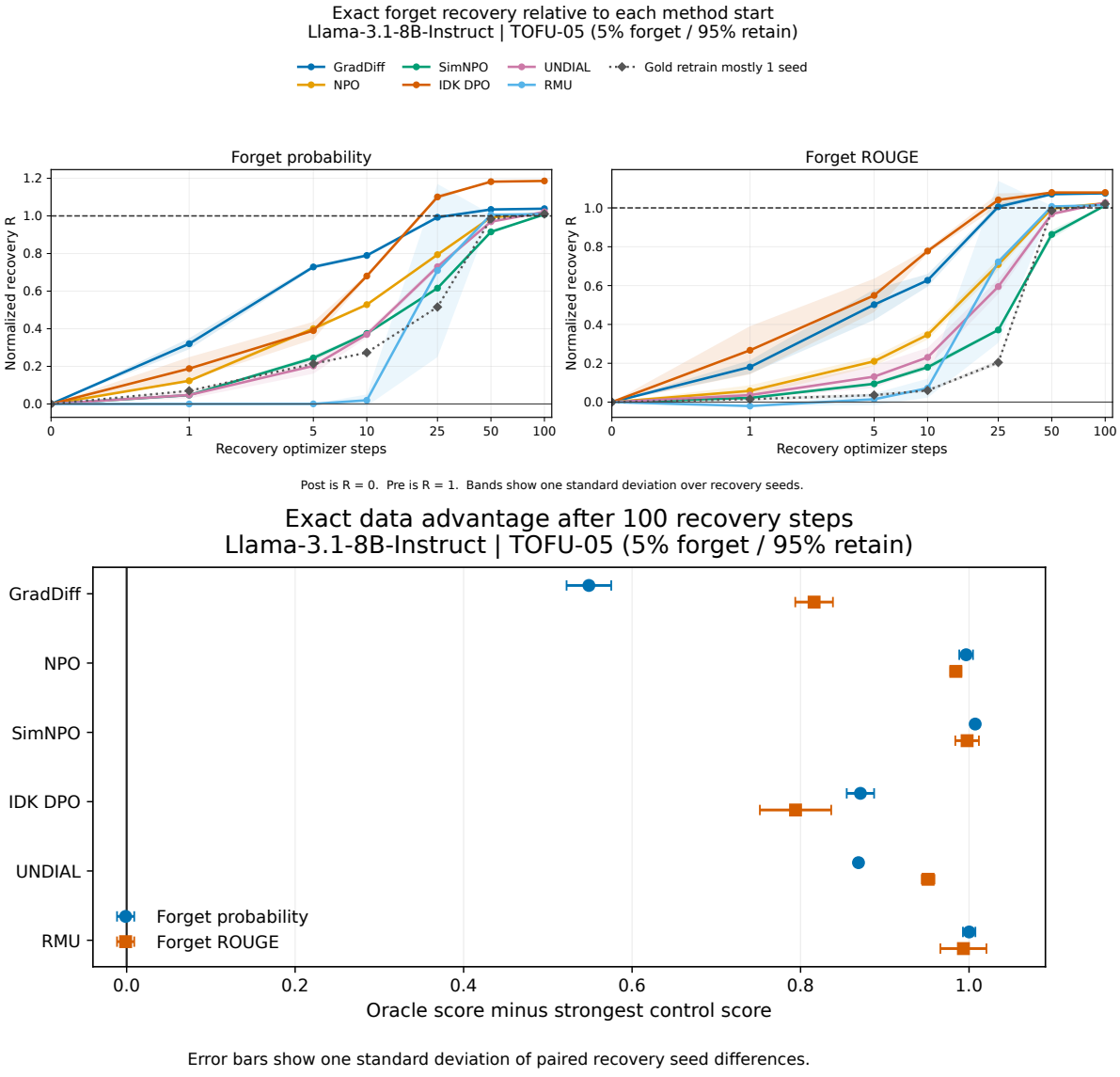

Figure 5: Additional Llama-3.1-8B-Instruct results on TOFU-05. Top: normalized recovery relative to each method's post-unlearning state. Bottom: exact-data advantage after 100 recovery steps, computed by subtracting the strongest compute-matched control. Bands and error bars show one standard deviation over three recovery seeds.

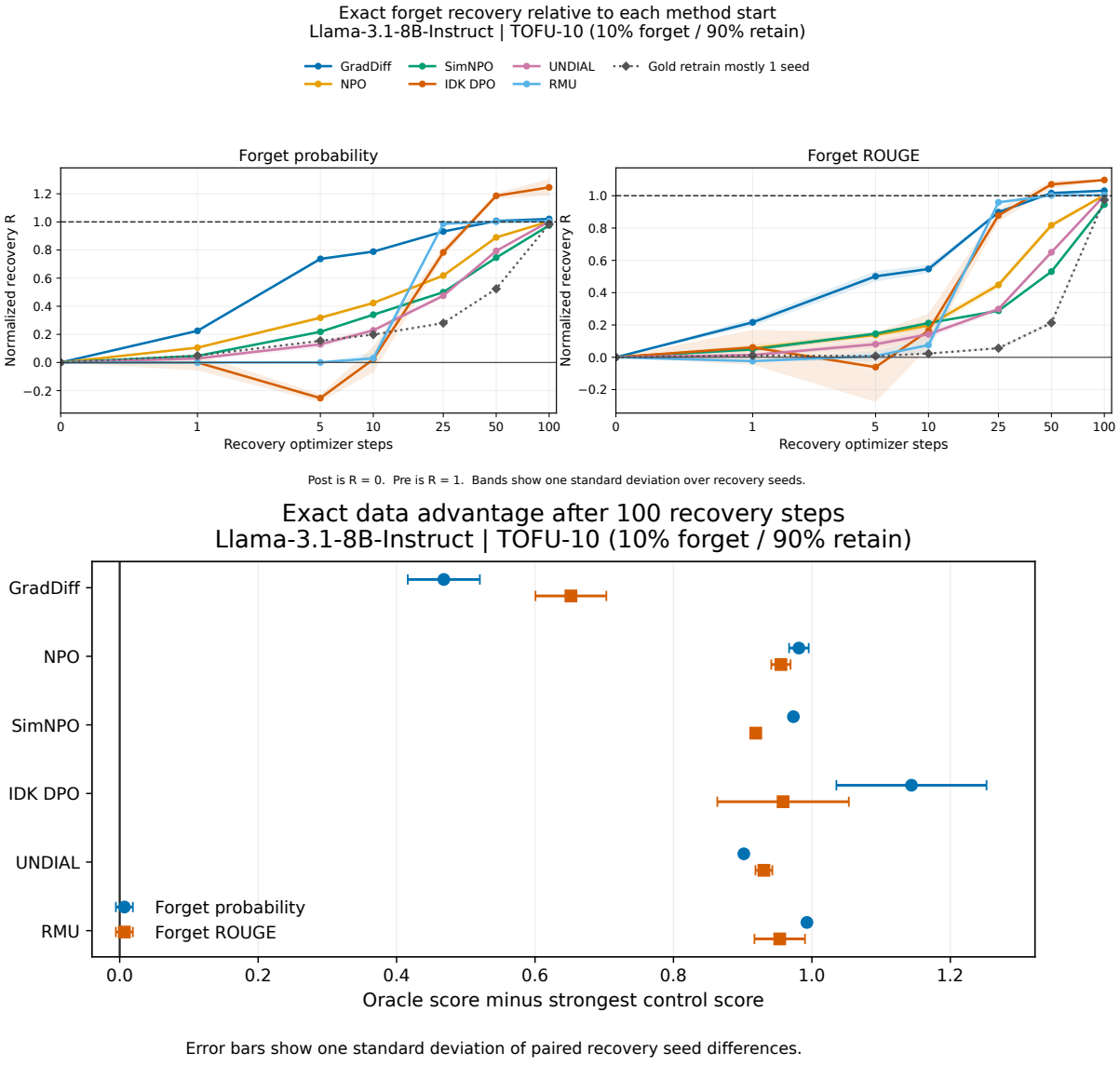

Figure 6: Additional Llama-3.1-8B-Instruct results on TOFU-10. The upper panel reports normalized recovery from the post-unlearning state toward the pre-unlearning reference. The lower panel reports exact data advantage after 100 recovery steps relative to the strongest compute matched control. Bands and error bars show one standard deviation over three recovery seeds.

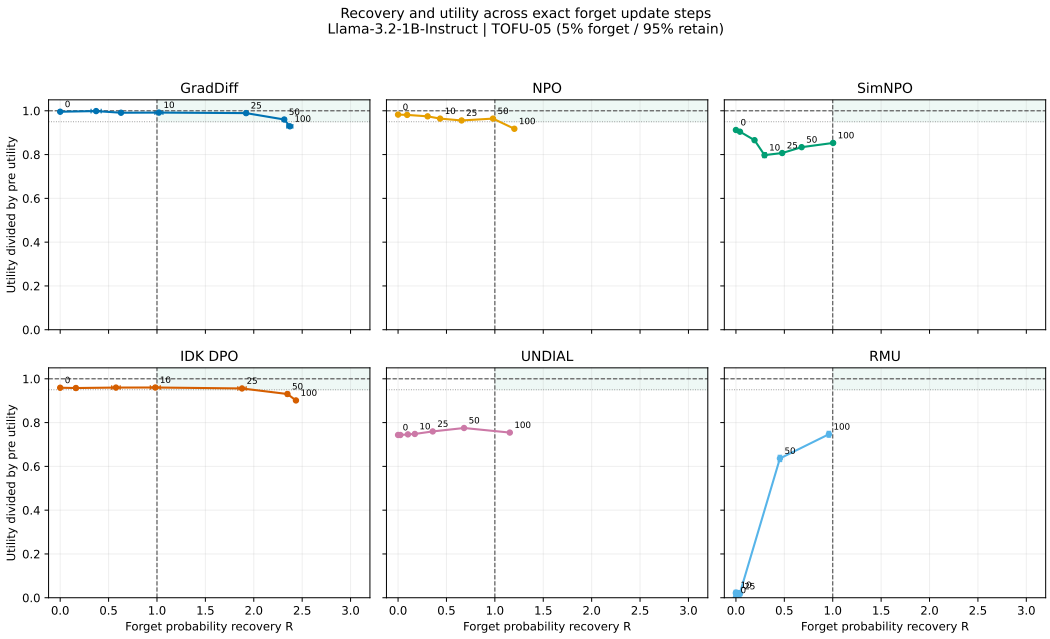

Figure 7: Recovery and retained utility for Llama-3.2-1B-Instruct on TOFU-05.

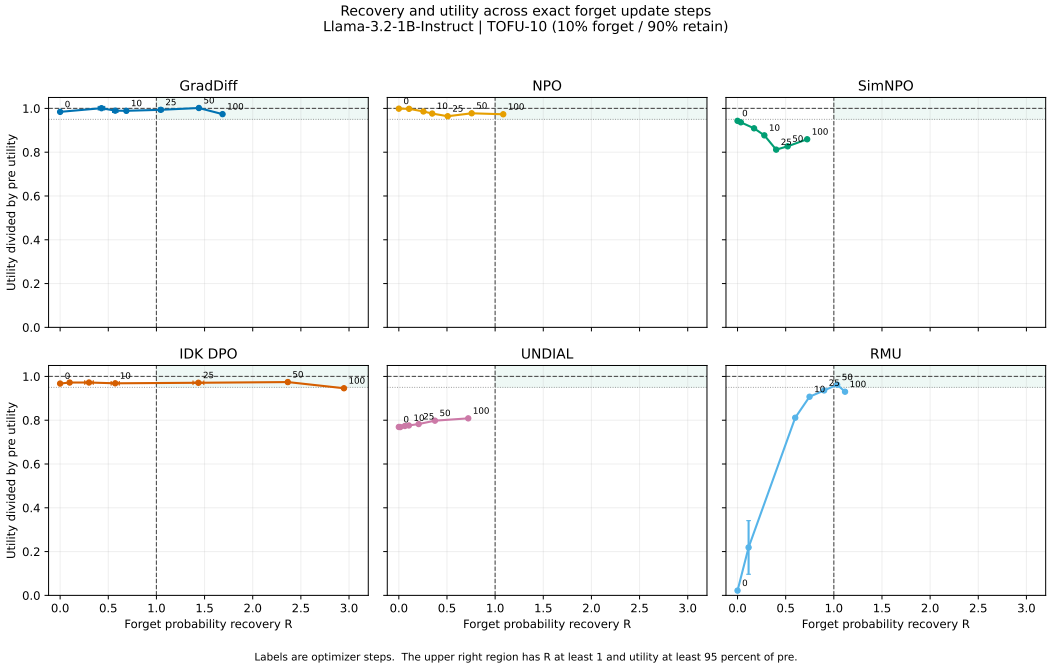

Figure 8: Recovery and retained utility for Llama-3.2-1B-Instruct on TOFU-10.

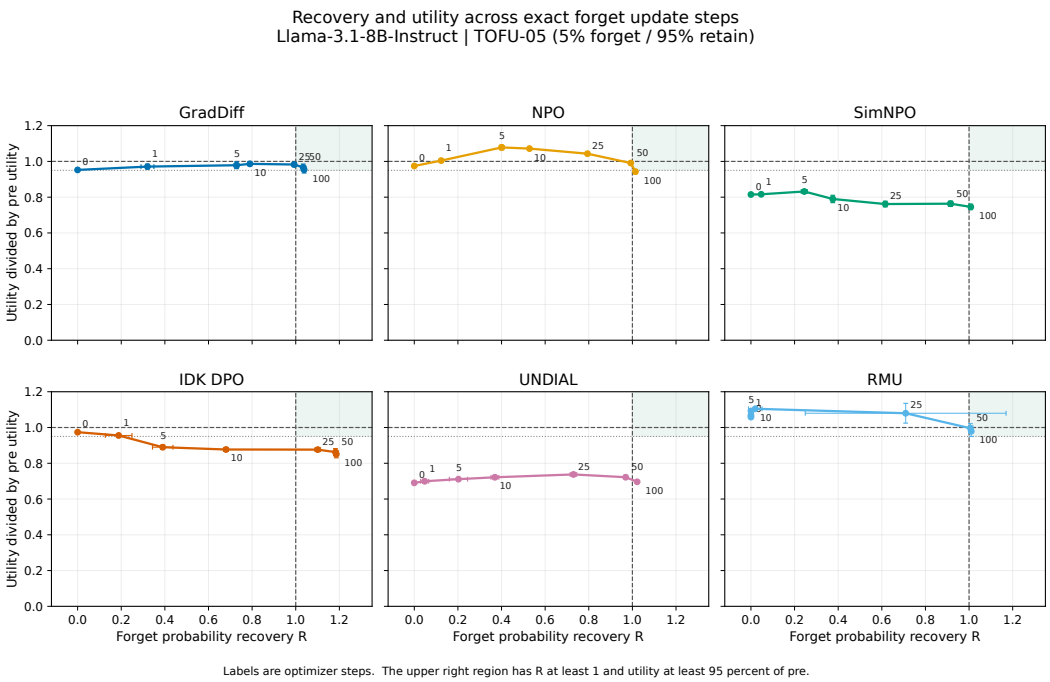

Figure 9: Recovery and retained utility for Llama-3.1-8B-Instruct on TOFU-05.

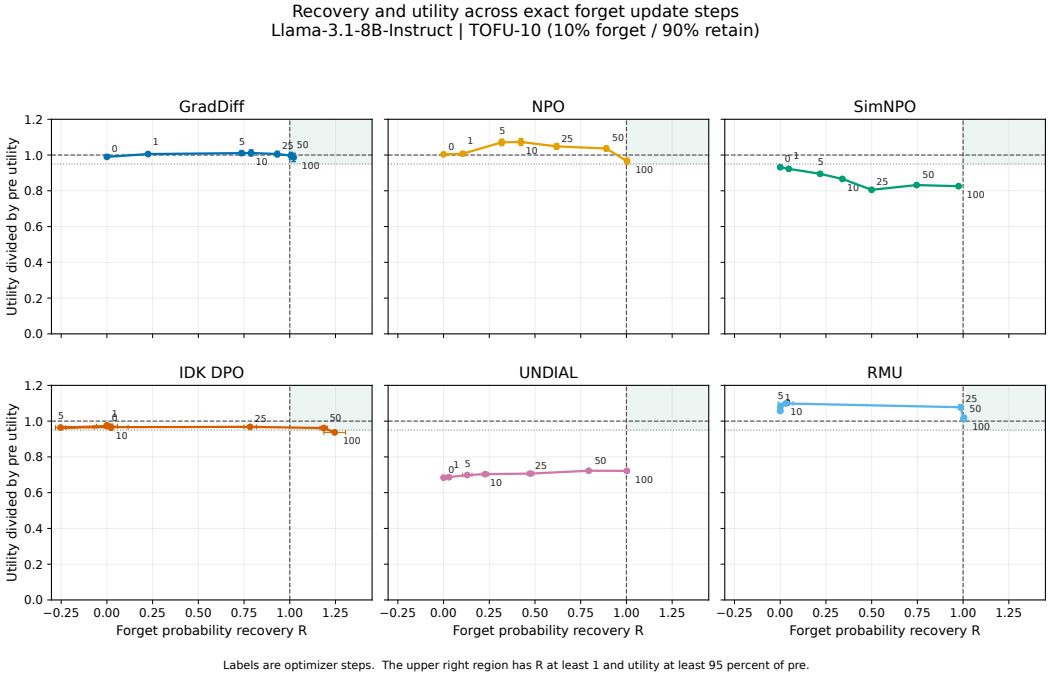

Figure 10: Recovery and retained utility for Llama-3.1-8B-Instruct on TOFU-10.