Old Memories Die Hard: Unveiling Shallow Forgetting and Ripple Effects in Privacy Unlearning

Anonymous ACL submission

Abstract

Large language models (LLMs) often memorize private information during training, raising serious privacy concerns. While machine unlearning has emerged as a promising mitigation, its effectiveness against privacy attacks remains unclear. We introduce PrivUn, an evaluation framework for privacy unlearning under recovery attacks, paired with three measurement tools: forgetting scores, association metrics, and layer-wise analysis of forgetting depth. This framework studies not only whether a model forgets, but also how forgetting propagates and where it fails. Using PrivUn, we make two findings. (i) *The PII ripple effect is gradient-driven, not semantic.* Graph-based proximity is uncorrelated with PII forgetting, while gradient-based association is more strongly correlated. (ii) *Label-based unlearning is typically shallow, while PII is stored across deeper layers.* Layer-wise evidence suggests that label-based methods mainly change the final layers, explaining why they reduce direct retrieval but remain vulnerable to stronger recovery attacks. Overall, privacy unlearning is limited by gradient coupling across data and insufficient intervention depth within the model.

1 Introduction

Large language models (LLMs) inevitably memorize personally identifiable information (PII) during training, creating significant privacy risks (Lukas et al., 2023) and compliance challenges (e.g., GDPR (Politou et al., 2018)). While *machine unlearning* (e.g., GA (Jang et al., 2023), NPO (Zhang et al., 2024), WHP (Eldan and Russinovich, 2023)) has emerged as a promising solution to erase specific data, evaluating its true effectiveness remains an open challenge. Existing benchmarks, such as TOFU (Maini et al., 2024), MUSE (Shi et al., 2024b), and WMDP (Li et al., 2024), only assess unlearning against passive attackers who merely observe model outputs. This leaves a critical gap: they fail to measure robustness against active attackers who can extract "forgotten" data through techniques like in-context learning (ICL) (Shumailov et al., 2024) and fine-tuning (Chen et al., 2024). Moreover, current evaluations often under-measure privacy leakage in existing unlearning methods, despite the social importance of privacy protection.

Beyond this evaluation gap, we posit that PII differs fundamentally from general knowledge, motivating a closer examination of PII unlearning mechanisms. *General knowledge* (e.g., "Paris is the capital of France") exhibits structural entanglement through semantic and logical relations. These connections are often explicitly modeled through knowledge graphs, which is intuitive and well-established: when a model forgets "France," it naturally affects related concepts like "Paris," "Europe," or "French." In contrast, *PII* represents isolated knowledge that lacks such logical or semantic dependencies. Because of this distinction, intuitions derived from knowledge unlearning may not apply to privacy protection. This discrepancy raises a pivotal question: *In the absence of semantic structure, how are isolated PII records entangled, stored, and forgotten within the model?*

• *New evaluation benchmark*. To fill this gap, we propose PrivUn, a comprehensive benchmark designed to assess PII unlearning robustness against active attackers under a three-tier threat model: direct retrieval (**P1**), recovery via ICL (**P2**), and recovery via fine-tuning (**P3**).

To probe the underlying dynamics of PII unlearning, we use three quantitative metrics: (i) forgetting scores that measure the divergence in generation probabilities for PII sequences; (ii) association metrics that capture relationships between data samples through relational graphs, gradients, and representation features; (iii) forgetting depth assessment using layer-wise *Centered Kernel Alignment* (CKA) analysis to track representation changes across model layers and pinpoint where private

information remains.

• *New findings on privacy forgetting*. Across 19 approaches, our quantitative analysis reveals two findings: (1) **The PII ripple effect is gradient-driven, not semantic.** Prior work on knowledge unlearning has modeled ripple effects as propagation through human-interpretable relations on knowledge graphs (Cohen et al., 2024; Wei et al., 2025; Wu et al., 2024): forgetting "France" propagates to "Paris" along an explicit logical edge. This picture does not transfer to PII. On the Enron sender-recipient network, Personalized PageRank score is uncorrelated with forgetting effectiveness ($r = -0.05$), while gradient-based association is strongly correlated ($r = 0.73$). Thus, PII is entangled through shared parameter-update directions rather than explicit relational structure. (2) **Label-based unlearning is typically shallow, while PII is memorized across deeper layers.** Layer-wise CKA shows that most label-based methods leave layers 0–23 of LLaMA-3.2-3B nearly unchanged (≥ 0.98) and mainly modify the final 4 layers. This reflects a known property of post-training (Zhou et al., 2023; Wang et al., 2025; Aghajanyan et al., 2021): output-only losses mostly affect late layers. Memorized PII, however, persist across mid-to-deep layers, creating a mismatch that explains why these methods appear effective at the output but are easily recovered.

• *Analysis-driven refinements*. These findings motivate two refinements: (1) Since PII ripple effects are gradient-driven, we select forget-set examples by gradient similarity. A 10% gradient-selected core set achieves a **P3** recovery rate of 32.19%, matching a GA baseline that uses 50% randomly selected forget examples. (2) Motivated by the shallow-forgetting tendency, we add a hidden-layer constraint by replacing RMU's random-vector anchor with the frozen pre-trained base model. This reduces **P3** to 30.39% while raising QA utility from 8.35 to 27.70. We present these as analysis-driven refinements rather than as a methodological contribution. The key takeaway is that privacy unlearning is governed by gradient coupling across data and the depth of intervention within the model.

2 Preliminaries

2.1 Privacy Unlearning for LLMs

LLMs trained on web-scale corpora often memorize PII such as home addresses and email addresses, because filtering such data at scale is difficult. This creates privacy risks in deployed models (Carlini et al., 2021, 2022; Huang et al., 2022; Lukas et al., 2023). Machine unlearning aims to remove such information while preserving performance on non-private data, typically through fine-tuning-based updates rather than full retraining.

Formally, let the target model f_{target} be trained on $\mathbb{D} = \{(x_i, y_i)\}_{i \in [n]}$. Unlearning removes the influence of a forget set $\mathbb{D}_F \subseteq \mathbb{D}$ while preserving performance on a retain set $\mathbb{D}_R = \mathbb{D} \setminus \mathbb{D}_F$. An unlearning algorithm U produces $f_{\text{unlearn}} = U(f_{\text{target}}, \mathbb{D}_F, \mathbb{D}_R)$, which should approximate retraining on $\mathbb{D}_R$ alone at lower cost.

Realistic Constraints. Traditional unlearning settings motivated by GDPR (Politou et al., 2018) assume defenders can enumerate all data to remove. For web-scale LLM training, this is unrealistic: private information is scattered across large corpora and entangled with non-private content. We therefore propose a realistic formulation where defenders: (1) *have access to only a limited subset of private data requiring removal*; (2) *have white-box access to the target model f_{target} trained on mixed private and non-private data*; (3) *have access to a clean proxy retain set $\mathbb{D}_R$ that does not contain the private data and is used to preserve utility, rather than the exact set-theoretic complement $\mathbb{D} \setminus \mathbb{D}_F$*.

Problem Definition. Given f_{target} trained on $\mathbb{D}$ containing private data $\mathbb{D}_F$ and non-private data, the defender observes only a known subset $\mathbb{D}_{F_k} \subset \mathbb{D}_F$ and a retain set $\mathbb{D}_R$. The goal is to: (1) *remove the influence of the entire private dataset $\mathbb{D}_F$, including both known ($\mathbb{D}_{F_k}$) and unknown ($\mathbb{D}_{F_{uk}} = \mathbb{D}_F \setminus \mathbb{D}_{F_k}$) subsets*; (2) *preserve model utility on non-private data*. The resulting model f_{unlearn} should behave as if it had never seen $\mathbb{D}_F$.

2.2 Limitations of Unlearning Benchmarks

The effectiveness of LLM unlearning is undermined by relearning attacks, where an adversary restores forgotten knowledge using just a few original samples (Hu et al., 2024). An attacker can achieve this through fine-tuning or even in-context learning (ICL) (Shumailov et al., 2024). This is especially concerning for privacy, because private information can be recovered with minimal data and commodity tools (Chen et al., 2024). Defenses such as sharpness-aware minimization (Fan et al., 2025) and adversarial training (Sheshadri et al., 2024) do not fully address this risk.

Existing benchmarks such as TOFU (Maini et al., 2024), MUSE (Shi et al., 2024b), and WMDP (Li

et al., 2024) mainly test against passive attackers and do not measure resilience to active recovery attacks. This gap motivates a benchmark for testing unlearning robustness against active attacks.

2.3 Related Work

Privacy Unlearning. Recent work has explored methods and benchmarks for privacy protection in LLMs. Kassem et al. (2023) propose a reinforcement learning approach to de-memorize privacy-sensitive information, while new benchmarks (Parii et al., 2025; Qian et al., 2025) begin to standardize PII unlearning evaluation. Although Qian et al. (2025) study various privacy attacks, their focus is on attack evaluation rather than the causes of defense failure. Our work instead shifts the focus from "how to attack" to "why unlearning fails."

Robust Unlearning. Another line of work shows that current LLM unlearning methods remain fragile, as forgotten knowledge can often be recovered through attacks (Łucki et al., 2025). Che et al. (2025) study model-tampering attacks that modify latent activations or weights; Zhang et al. (2025) show catastrophic failures under quantization; and Lynch et al. (2024) highlight the lack of robustness against adversarial threats. Hu et al. (2024) further introduce relearning attacks to recover forgotten data. To improve robustness, Fan et al. (2025) leverage Sharpness-Aware Minimization, while Zhang et al. (2025) propose a tailored framework against quantization-induced unlearning failure. These studies share our observation that forgotten knowledge can persist in intermediate layers, but our work further investigates the mechanisms behind incomplete unlearning.

Position Relative to Prior Work. Our work emphasizes active PII recovery attacks, realistic known/unknown forget splits, and a mechanism-based analysis of why current methods fail. We provide a detailed comparison in Appendix F.

3 The PrivUn Evaluation Benchmark

3.1 The Design of PrivUn

To address these limitations, we introduce PrivUn, a new benchmark that audits the robustness of unlearning methods against three recovery attacks and quantifies how completely private data is forgotten.

Evaluation Metrics. PrivUn measures how successfully private data has been erased against three increasingly sophisticated active attack scenarios (See Section D.1 for PII matching rule):

- **P1. Direct Retrieval** measures direct memorization: $\frac{1}{|\mathbb{D}_F|}\sum_{q\in\mathbb{D}_F}\mathbf{1}_{\text{PII}\in f(q)}$, where q queries a PII item in the forget set and $\mathbf{1}_{\text{condition}}$ is 1 if the condition holds.
- **P2. Recovery via ICL** measures few-shot recovery (we use 10-shot, examples drawn from $\mathbb{D}_{F_k}$): $Rec_{ICL} = \frac{1}{|\mathbb{D}_F|}\sum_{q\in\mathbb{D}_F}\mathbf{1}_{\text{PII}\in f(q,k\text{-shot})}$, where k is a fixed shot count shared across queries.
- **P3. Recovery via Fine-tuning** measures recovery after the attacker fine-tunes the unlearned model on $|D|$ samples from $\mathbb{D}_{F_k}$ for e epochs: $Rec_{FT} = \frac{1}{|\mathbb{D}_F|}\sum_{q\in\mathbb{D}_F}\mathbf{1}_{\text{PII}\in f_{ft}(q)}$. Cost is reported as $(|D|, e)$. Fine-tuning is realistic for both open-weight models (Che et al., 2025; Hu et al., 2024) and black-box APIs.

We exclude the Min-K metric (Shi et al., 2024a) designed for *dataset inference*, whereas our focus is on *PII extraction* (i.e., specific PII can be retrieved even via novel prompts). The results in Section D.1 confirm the limitation of Min-K.

To measure model utility, we define the **U1** metric reported as three components: (i) ROUGE on a question-answering task over $\mathbb{D}_R$ ("QA"), (ii) the same QA task with 10-shot in-context examples ("QA+ICL"), and (iii) MMLU (Massive Multitask Language Understanding) (Hendrycks et al., 2021) 5-shot accuracy for general capability. A higher U1 score indicates better utility preservation.

Evaluation Corpus. For the forget set, we evaluate privacy leakage on Enron (Klimt and Yang, 2004). This dataset contains approximately 500k emails from Enron Corporation employees, which were made public by the Federal Energy Regulatory Commission. From Enron, we extract [PERSON, PII] pairs and convert them into standardized QA prompts of the form "Q: Tell me the [*PII type*] of [PERSON], A: [PII]". To study more diffuse PII, we also use TOFU (Maini et al., 2024), extracting 424 QA pairs on author birth dates and birthplaces from the original 4,000-question corpus. Following Section 2.1, we split each corpus into known and unknown forget sets. The known set ranges from 5% to 100% of the full forget set; we report 20% and 50% for clarity. Duplicate PII records are removed before splitting.

For the retain set, we use MUSE News (Shi et al., 2024b), which includes 3.56k news articles that we process into 100 question-answer pairs to evaluate whether the model retains its general capabilities after unlearning.

Evaluated Unlearning Methods. We evaluate ap-

proximate unlearning methods from two families. **Training pipeline manipulation** changes the objective to encourage forgetting, whereas **data manipulation** changes forget-set labels to overwrite or confuse model knowledge. We also consider two retain-set regularizers: gradient-based descent (GDR) and KL divergence minimization (KLR).

This yields a suite of 19 methods. **Training pipeline manipulation** methods include: (1) Gradient Ascent (Jang et al., 2023) variants: GA, GA_{GDR}, GA_{KLR}; (2) NPO (Zhang et al., 2024) variants: NPO, NPO_{GDR}, NPO_{KLR}; (3) DPO (Zhang et al., 2024) variants: DPO, DPO_{GDR}, DPO_{KLR}; (4) Task Vector (Ilharco et al., 2023) and (5) RMU (Li et al., 2024). **Data manipulation** methods include: (1) Random Labeling/Mapping (Maini et al., 2024) variants[1]: RL, RL_{GDR}, RM, RM_{GDR}; (2) "I Don't Know" variants: IDK, IDK_{GDR}; (3) WHP (Eldan and Russinovich, 2023) variants: WHP, WHP_{GDR}.

Beyond the primary categorization, these methods can be further classified along two dimensions: (1) **Targeted vs. Untargeted** methods (Yuan et al., 2025) differ in whether they specify alternative outputs (e.g., DPO, IDK) or simply aim to prevent original responses (e.g., NPO). (2) **Label-based vs. Representation-based** methods differ in their intervention level: label-based methods modify output distributions, while representation-based methods directly alter hidden-layer representations. The classification is shown in Table 1, and method details are provided in Appendix B.

As a gold standard, we include Retrained Model $f_{retrain}$, trained from scratch on the retain set.

3.2 Quantifying Forgetting

To understand why private information persists after unlearning, PrivUn uses three compact metrics.

Forgetting Score. Because **P3** depends on the attacker's fine-tuning samples, we also report a more stable *forgetting score* for PII generation. For a PII sequence $\mathbf{y} = [y_1, \ldots, y_T]$ under prefix x, let $P_{seq}(y|x) = \prod_{t=1}^{T} P(y_t|x, y_1, \ldots, y_{t-1})$. We then define $\text{FS}(x, y) = \log(P_{f_{\text{target}}}(y|x)) - \log(P_{f_{\text{unlearn}}}(y|x))$. A higher **FS** means the unlearned model assigns lower probability to the PII sequence, indicating stronger forgetting.

Association Score. To measure how samples are coupled during unlearning, we define an *Association Score (AS)* from three views. *Gradient-based association* uses gradient dot products, $\text{AS}_{grad}(x_i, x_j) = \nabla_{\boldsymbol{\theta}} L(f(x_i; \boldsymbol{\theta})) \cdot \nabla_{\boldsymbol{\theta}} L(f(x_j; \boldsymbol{\theta}))$, which captures whether two samples induce similar update directions and how strongly they affect the loss landscape. *Representation-based association* uses hidden-state cosine similarity, $\text{AS}_{repr}(x_i, x_j) = \cos(h_l(x_i), h_l(x_j))$, where $h_l(\cdot)$ is the average PII-token representation at layer l. *Graph-based association* uses *Personalized PageRank (PPR)* score on an explicit relational graph when such a graph exists. Together, these views test whether ripple effects are driven by optimization, representation, or human-interpretable structure.

Forgetting Depth Assessment. We assess forgetting depth in two ways. First, the gap between direct retrieval (**P1**) and fine-tuning recovery (**P3**) measures the gap between shallow and deep forgetting: a large gap means PII is suppressed at the surface but still recoverable. Second, we use layer-wise Centered Kernel Alignment (CKA) (Kornblith et al., 2019) between f_{target} and f_{unlearn} to locate where representations change. Lower CKA at a layer indicates stronger intervention at that depth.

4 Understanding Challenges of Forgetting

4.1 Measurement Study

Experimental Setup. We start with a pre-trained LLaMA-3.2-3B model (Dubey et al., 2024) as the base architecture and fine-tune two initial models: the target model f_{target}, trained for 5 epochs on the full dataset $\mathbb{D}_F \cup \mathbb{D}_R$ from Enron and MUSE News, and the retrained model f_{retrain}, trained only on the retain set $\mathbb{D}_R$ as the gold standard. For each unlearning method U, we obtain an unlearned model by applying it to the target model, $f_{\text{unlearn}} = U(f_{\text{target}}, \mathbb{D}_{F_k}, \mathbb{D}_R)$. All methods use a learning rate of 10^{-5} and batch size 32, with hyperparameters selected to maximize validation utility. We average results over 3 random seeds on 4 NVIDIA A100 GPUs, and verify that our findings also hold on GPT-2-Large (Radford et al., 2019) in Section D.3. Additional implementation details are given in Section D.2.

Attack Budget and Reporting Conventions. For **P3**, we report each method's recovery rate at the smallest attack budget $e \in \{3, 10\}$ at which the unlearned model is fully or near-fully recovered, capping at $e = 10$ epochs (where 10 epochs already exceeds the target model's training budget). Once the recovery rate plateaus, additional epochs do not change the qualitative comparison. To enable a uniform-budget comparison across all methods, the

[1] Since KLR operates on the loss function, it is only paired with training-pipeline manipulation methods.

GPT-2 results in Table 3 use a single fixed budget of $10e$ for every method, including f_{retrain}.

Effectiveness against Passive Observation (P1). *Training pipeline* and *data manipulation* methods show markedly different behavior under direct retrieval **P1**. Training-pipeline manipulation methods such as GA and NPO are highly effective, reducing information leakage to below 0.5%. In contrast, data manipulation methods are unstable: random labeling variants (RL, RM, IDK) can leak more PII than the target model, with the worst case $\mathsf{RM}_{\mathsf{GDR}}$ reaching 40.73% recovery rate, compared with 30.73% for vanilla RM on the same split. This suggests that replacing PII targets with alternative labels does not remove memorized associations and may even reinforce them. The gap reflects their different mechanisms: training-pipeline manipulation modifies the optimization objective to discourage private-information generation, whereas data manipulation only attempts to overwrite existing knowledge. Thus, training-pipeline manipulation provides more reliable passive privacy protection.

Resilience to Active Attacks (P2&P3). Most unlearning methods remain vulnerable to active attacks, especially under **P3**. A clear robustness gap appears between *targeted* and *untargeted* methods: untargeted methods are far more resilient. For example, NPO keeps the **P2** recovery rate near zero (0.2%), whereas the targeted method DPO is easily compromised, with **P2** exceeding 30%. This suggests that steering the model toward a specific alternative response, such as "sorry I don't know," induces brittle changes that can be readily reversed.

A similar gap separates *representation-based* and *label-based* methods. RMU consistently outperforms label-based methods, achieving **P3** of 18-20% compared to 60%+ for others. This advantage stems from its mechanism: RMU directly modifies the model's hidden-layer representations to erase information, producing deeper forgetting than simply changing the output distribution, a phenomenon we further explore in Section 4.3.

Parallel Forgetting Patterns. A notable similarity in unlearning effectiveness emerges between the known forget set $\mathbb{D}_{F_k}$ and the unknown forget set $\mathbb{D}_{F_{uk}}$. Under the $\mathsf{GA}_{\mathsf{GDR}}$ method, for instance, the **P3** recovery rate is nearly identical for both sets (53.10% for the known set and 53.30% for the unknown set). This parallel behavior suggests that unlearning specific data causes a systematic propagation of forgetting that extends beyond the explicit targets. We term this phenomenon the "ripple effect" and analyze it in Section 4.2.

Utility Preservation (U1). LLaMA-3.2 is highly robust to utility degradation, with most unlearning methods maintaining QA performance close to the baseline QA score (27.35). The notable exceptions are DPO and RMU, which cause larger drops in performance. Similarly, MMLU scores remain relatively stable across methods, with RMU showing the lowest performance at 50.45.

This raises two questions: *why does unlearning a known dataset also affect unknown data*, and *why do seemingly effective methods fail against active attacks*? We explore these questions by examining the ripple effect and shallow forgetting.

4.2 Ripple Effect in Unlearning

Our first key finding is that unlearning induces a *ripple effect*: forgetting on the known set spills over to related samples in the unknown set. Crucially, for PII this effect is not explained by human-interpretable relations such as social graphs; it is better explained by optimization coupling.

Forget Set Size. Across forget-set sizes from 5% to 100%, Figure 2 shows nearly parallel recovery curves for $\mathbb{D}_{F_k}$ and $\mathbb{D}_{F_{uk}}$ under GA and NPO, providing clear evidence of ripple effects. Both sets appear forgotten under **P1** but remain highly recoverable under **P3**, indicating that these ripple effects are largely shallow, as detailed in Section 4.3.

Relational Graphs Do Not Explain Ripple Effects. If ripple effects followed explicit semantic structure, the Enron sender-recipient graph should predict them. However, using the graph in Figure 5, we find near-zero PageRank-forgetting correlations: Pearson $r = -0.09$ for NPO and $r = -0.05$ for GA (Figure 1a). TOFU shows the same mismatch despite lacking usable graph paths (Table 4). Thus, unlike knowledge unlearning (Wu et al., 2024; Wei et al., 2025), PII ripple effects are not driven by human-interpretable relational structure.

Gradient Association Drives Ripple Effects. We next test whether ripple effects are explained by optimization coupling. For each sample in $\mathbb{D}_{F_{uk}}$, we measure its gradient similarity to $\mathbb{D}_{F_k}$ and correlate that score with forgetting. Figure 1b shows a strong positive correlation: for GA, Pearson/Spearman are 0.7314/0.7196; for NPO, they are 0.5540/0.5069. Samples with more similar update directions are more likely to be forgotten, indicating that ripple effects are gradient-driven rather than accidental.

Representation Association Appears Later but is Weaker. Representation similarity becomes more

Table 1: Privacy and utility measurements on Enron and News using LLaMA-3.2-3B. For f_{target} and f_{retrain}, recovery rates are evaluated on the full $\mathbb{D}_F$ as they do not have forget set splits. The **P3** recovery cost uses $|D|$ for fine-tuning data size and e for epochs. **U1** reports QA, QA+ICL (10-shot), and 5-shot MMLU. For the classification, training pipeline or data manipulation; targeted (✓) or untargeted (✗); label-based (✓) or representation-based (✗).

Methods	Classification		P1. Direct Retrieval		P2. Recovery via ICL		P3. Recovery via Fine-tuning			U1. Utility Preserv.		
	Targeted	Label-based	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	Cost	QA	QA+ICL	MMLU
Target f_{target}			14.53%		29.94%		60.60%		$\lvert D\rvert = 50, 3e$	27.35	36.38	52.59
Retrain f_{retrain}			0.02%		4.95%		13.14%		$\lvert D\rvert = 50, 3e$	26.69	35.04	55.08
20% Enron + News												
GA	✗	✓	0.49%	0.43%	0.49%	0.61%	54.27%	54.68%	$\lvert D\rvert = 50, 10e$	26.71	36.19	52.59
GA$_{\text{GDR}}$	✗	✓	0.24%	0.43%	0.37%	0.47%	53.10%	53.30%	$\lvert D\rvert = 50, 10e$	27.57	34.27	52.54
GA$_{\text{KLR}}$	✗	✓	12.20%	13.13%	31.34%	32.84%	66.10%	64.58%	$\lvert D\rvert = 50, 3e$	27.35	38.25	52.66
NPO	✗	✓	0.49%	0.49%	0.24%	0.06%	56.83%	58.22%	$\lvert D\rvert = 50, 3e$	23.78	36.48	52.47
NPO$_{\text{GDR}}$	✗	✓	0.37%	0.52%	0.24%	0.06%	57.56%	57.42%	$\lvert D\rvert = 50, 3e$	24.55	37.49	53.15
NPO$_{\text{KLR}}$	✗	✓	12.20%	13.13%	31.34%	32.84%	64.58%	64.58%	$\lvert D\rvert = 50, 3e$	27.35	36.38	52.97
Task Vector	✗	✓	2.92%	2.62%	12.68%	12.82%	66.10%	64.58%	$\lvert D\rvert = 50, 3e$	26.49	37.05	52.53
DPO	✓	✓	0.00%	0.00%	30.61%	31.16%	65.98%	65.46%	$\lvert D\rvert = 50, 10e$	9.97	24.76	54.21
DPO$_{\text{GDR}}$	✓	✓	0.00%	0.00%	34.27%	35.09%	69.51%	67.86%	$\lvert D\rvert = 50, 10e$	10.02	24.83	54.39
DPO$_{\text{KLR}}$	✓	✓	0.00%	0.06%	51.10%	51.11%	65.24%	63.81%	$\lvert D\rvert = 50, 3e$	11.69	22.84	53.56
RMU	✗	✗	0.00%	0.00%	0.00%	0.00%	18.32%	20.30%	$\lvert D\rvert = 50, 10e$	8.35	9.51	50.45
RL	✗	✓	12.32%	13.00%	37.32%	37.95%	65.85%	64.64%	$\lvert D\rvert = 50, 3e$	29.25	36.36	52.62
RL$_{\text{GDR}}$	✗	✓	12.44%	12.03%	39.63%	37.56%	62.93%	60.83%	$\lvert D\rvert = 50, 3e$	31.33	34.07	52.39
RM	✗	✓	30.73%	28.33%	38.05%	37.22%	25.85%	21.81%	$\lvert D\rvert = 50, 3e$	29.09	34.67	52.63
RM$_{\text{GDR}}$	✗	✓	40.73%	38.87%	36.10%	37.07%	65.12%	63.08%	$\lvert D\rvert = 50, 3e$	32.31	35.55	52.55
WHP	✗	✓	1.59%	0.70%	22.68%	23.09%	66.71%	65.28%	$\lvert D\rvert = 50, 3e$	28.76	36.09	52.43
WHP$_{\text{GDR}}$	✗	✓	0.98%	0.82%	12.32%	12.98%	65.85%	66.49%	$\lvert D\rvert = 50, 10e$	27.49	35.38	52.37
IDK	✓	✓	16.71%	15.35%	32.56%	34.05%	67.68%	66.43%	$\lvert D\rvert = 50, 3e$	28.95	35.95	52.63
IDK$_{\text{GDR}}$	✓	✓	21.83%	20.07%	29.63%	29.61%	66.71%	66.31%	$\lvert D\rvert = 50, 3e$	33.49	34.99	52.51
50% Enron + News												
GA	✗	✓	0.39%	0.49%	0.44%	0.39%	30.82%	32.47%	$\lvert D\rvert = 50, 10e$	26.00	35.02	51.41
GA$_{\text{GDR}}$	✗	✓	0.34%	0.39%	0.44%	0.34%	39.83%	41.20%	$\lvert D\rvert = 50, 10e$	28.99	34.97	51.55
GA$_{\text{KLR}}$	✗	✓	12.09%	13.80%	32.62%	32.47%	64.31%	65.48%	$\lvert D\rvert = 50, 3e$	27.35	36.38	52.19
NPO	✗	✓	0.34%	0.44%	0.10%	0.05%	56.61%	56.61%	$\lvert D\rvert = 50, 3e$	20.07	38.15	52.45
NPO$_{\text{GDR}}$	✗	✓	0.44%	0.54%	0.10%	0.05%	56.12%	58.75%	$\lvert D\rvert = 50, 3e$	25.33	36.91	52.41
NPO$_{\text{KLR}}$	✗	✓	12.09%	13.80%	32.62%	32.47%	38.27%	38.86%	$\lvert D\rvert = 50, 3e$	27.35	36.38	52.90
Task Vector	✗	✓	3.94%	4.29%	18.04%	17.60%	64.75%	65.77%	$\lvert D\rvert = 50, 3e$	27.48	37.41	52.43
DPO	✓	✓	0.00%	0.00%	0.00%	0.00%	67.92%	69.43%	$\lvert D\rvert = 50, 10e$	0.00	6.44	52.31
DPO$_{\text{GDR}}$	✓	✓	0.00%	0.00%	1.37%	1.66%	63.04%	64.16%	$\lvert D\rvert = 50, 10e$	0.00	15.19	53.67
DPO$_{\text{KLR}}$	✓	✓	0.00%	0.06%	58.65%	60.99%	65.53%	67.09%	$\lvert D\rvert = 50, 3e$	11.81	19.94	54.31
RMU	✗	✗	0.00%	0.00%	0.00%	0.00%	16.52%	18.14%	$\lvert D\rvert = 50, 10e$	7.75	8.68	50.47
RL	✗	✓	9.56%	11.41%	43.49%	43.98%	62.90%	63.43%	$\lvert D\rvert = 50, 3e$	30.12	35.44	52.29
RL$_{\text{GDR}}$	✗	✓	10.04%	9.90%	31.25%	29.50%	52.80%	53.24%	$\lvert D\rvert = 50, 3e$	30.23	34.88	52.26
RM	✗	✓	27.89%	29.84%	39.05%	40.03%	62.46%	62.51%	$\lvert D\rvert = 50, 3e$	28.78	35.99	52.12
RM$_{\text{GDR}}$	✗	✓	33.40%	34.42%	20.09%	19.80%	62.46%	62.46%	$\lvert D\rvert = 50, 3e$	30.77	35.91	52.11
WHP	✗	✓	0.59%	0.68%	18.97%	19.99%	61.87%	64.65%	$\lvert D\rvert = 50, 3e$	27.50	35.68	52.19
WHP$_{\text{GDR}}$	✗	✓	0.15%	0.24%	4.29%	5.27%	61.87%	62.85%	$\lvert D\rvert = 50, 10e$	24.57	33.73	52.06
IDK	✓	✓	13.65%	15.02%	32.91%	32.96%	65.68%	67.28%	$\lvert D\rvert = 50, 3e$	28.95	35.95	52.49
IDK$_{\text{GDR}}$	✓	✓	19.25%	21.11%	30.91%	31.69%	65.82%	50.76%	$\lvert D\rvert = 50, 3e$	29.97	34.81	52.35

informative only in deeper layers. In Figure 1(c)–(e), the embedding layer shows essentially no correlation, the signal emerges around layer 16, and the final-layer correlation remains weaker than the gradient-based one (e.g., $r \approx 0.36$ versus 0.73 for GA). This suggests that deeper representations reflect the ripple effect, but optimization dynamics are still the primary cause.

Refinement via Gradient-Selected Core Sets. This mechanism has predictive value. Ranking forget-set samples by gradient association yields an efficient diagnostic core set: under **P3**, a 10% gradient-selected core set matches the forgetting effect of a random 50% set and the full forget set (Figure 2c). This supports the claim that gradient coupling, not semantic structure, drives PII ripple effects. Full ablations are deferred to Appendix C.

4.3 Shallow Forgetting

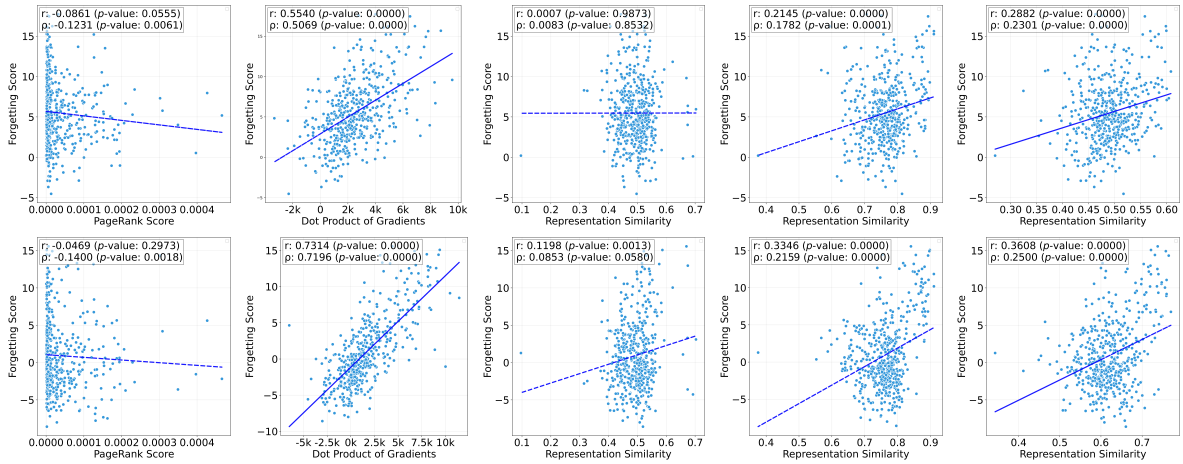
Existing unlearning methods often exhibit *shallow forgetting*: they suppress direct retrieval but remain vulnerable to active recovery. We study this through the **P1**–**P3** gap and layer-wise representation changes, and show that, for label-based methods, this behavior follows from how post-training updates propagate through depth.

Why label-based unlearning tends to be shallow. Label-based methods relabel the target sequence or redistribute output probabilities, but the loss still depends on θ only through the final output distribution. Thus, they behave like standard post-training procedures such as instruction tuning or DPO: the supervision is concentrated at the head.

Let h_l denote the hidden state at layer $l \in \{1, \ldots, L\}$. By the chain rule,

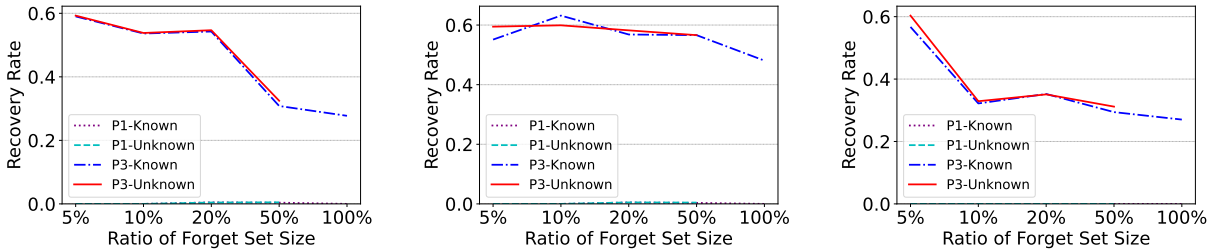
$$\frac{\partial \mathcal{L}}{\partial \theta_l} = \underbrace{\frac{\partial \mathcal{L}}{\partial h_L}}_{\text{output sensitivity}} \cdot \underbrace{\prod_{j=l+1}^{L} \frac{\partial h_j}{\partial h_{j-1}}}_{\text{Jacobian product}} \cdot \frac{\partial h_l}{\partial \theta_l}. \quad (1)$$

Empirically, this Jacobian product contracts with depth in trained transformers, and fine-tuning up-



(a) PPR Score (b) Gradient (c) Embedding Layer (d) Layer 16 (e) Last Hidden Layer

Figure 1: Correlation between association metrics and forgetting scores for NPO (top row) and GA (bottom row). (a) Relational graph shows no correlation. (b) Gradient-based association shows strong correlation. (c)-(e) Representation-based association strengthens in deeper layers (r = Pearson coefficient; ρ = Spearman coefficient).



(a) GA (b) NPO (c) Core-set GA

Figure 2: Recovery-rate comparison between known and unknown forget sets across different forget-set sizes and attack scenarios (P1, P3); panel (c) shows the effectiveness of an association-aware core set.

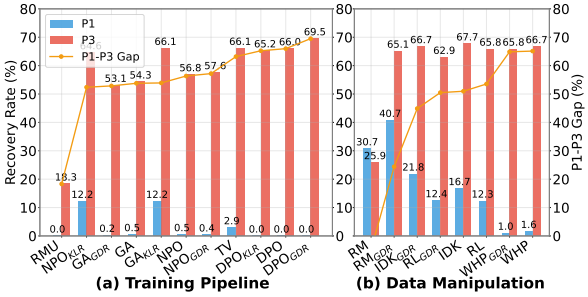


Figure 3: Shallow-forgetting analysis across unlearning methods on the 20% Enron split.

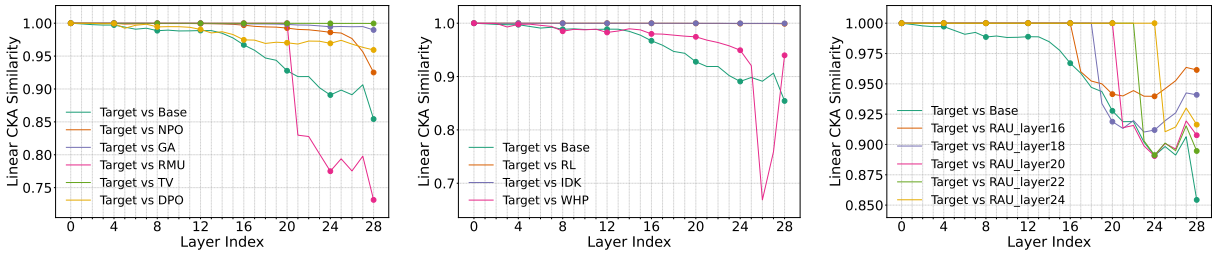
dates occupy a low-dimensional subspace (Merchant et al., 2020; Aghajanyan et al., 2021). Consequently, post-training updates are much larger near the head than in early or middle layers, matching the superficial-alignment picture of prior work (Zhou et al., 2023; Wang et al., 2025).

PII memorization differs because it is acquired during pre-training and persists across mid-to-deep layers (Merchant et al., 2020; Kumar et al., 2022). This creates a mismatch in which label-based unlearning primarily edits the head, while PII representations remain largely intact.

If the loss acts only through the output, then to first order ($\Delta\theta_l \approx 0$) in deep PII-bearing layers, so ($h_l^{\text{unlearn}}(x) \approx h_l^{\text{target}}(x)$). Those hidden states can be re-projected with minimal fine-tuning, explaining why label-based methods achieve low **P1** yet still allow 50–70% recovery under **P3**. The shallow CKA profile and high **P3** recovery therefore reflect the same structural fact.

This also suggests a criterion for deeper forgetting. The objective must inject gradients directly into intermediate representations rather than only through the head. RMU and RAU satisfy this with a representation-level loss, ($|h_l^{\text{unlearn}} - \text{anchor}_l|^2$), which produces non-vanishing gradients at the targeted layer. As a result, they are the only methods in our suite that visibly alter mid-layer CKA and reduce **P3** below 35%. We next present empirical evidence consistent with this prediction.

Attack Resistance Gap Comparison. Figure 3 presents the P1 and P3 recovery rates across different unlearning methods, revealing three distinct patterns in unlearning effectiveness: (1) methods



(a) Training Pipeline Manipulation (b) Data Manipulation (c) RAU

Figure 4: Cross-model CKA analysis for representative unlearning methods.

achieving deep forgetting with modest gaps (RMU), (2) methods failing across all scenarios (RL and IDK variants), and (3) methods demonstrating shallow forgetting with dramatic P1–P3 disparities up to 69.5% (DPO, NPO, GA variants, Task Vector), highlighting the persistence of private information in deeper model representations despite shallow protection. Overall, 15 of 19 methods exhibit gaps exceeding 50%, indicating that shallow forgetting is a broad limitation of current unlearning.

Layer-wise Representation Comparison. To locate where PII persists after unlearning, we use CKA to compare layer-wise representations across methods. Figure 4 reports CKA similarity over all 28 layers (layer 0 is the embedding layer) for both training-pipeline manipulation methods (Figure 4a) and data manipulation methods (Figure 4b).

The CKA curves follow the depth-decay pattern in Eq. (1). Most label-based methods keep CKA ≥ 0.98 from layer 0 through layer 23 and diverge from f_{target} only in the final four layers. This shallow divergence is the signature of a head-localized post-training update. Its shape matches the layer-wise update profiles reported for instruction tuning and DPO (Zhou et al., 2023; Wang et al., 2025), supporting our claim that label-based unlearning is mechanistically another form of post-training.

For example, NPO, the strongest label-based method on P1/P2, maintains CKA above 0.98 until a sharp drop at layer 24, yet still allows a 56–67% P3 recovery rate because the deep layers it leaves unchanged continue to encode the original PII. In contrast, the representation-based method RMU produces the largest mid-layer change, dropping to 0.83 at layer 20. It achieves the lowest P3 recovery in the standard suite, but at the cost of utility collapse (8.35 in Table 1), because it operates at the right depth with the wrong anchor.

This layer-wise analysis reveals that overly aggressive changes harm utility while output-layer modifications cause shallow forgetting. Combined with the analysis above, this yields the prescription: *unlearning methods must inject a non-output-only loss term and must do so at intermediate-to-deep layers*. Label-based methods cannot satisfy the first criterion; this is why they are uniformly shallow and fail to close the P1–P3 gap.

Refinement via Representation-Level Intervention. Based on this analysis, we test a simple representation-level refinement: **Representation Anchoring Unlearning** (RAU), which anchors the unlearned model to a frozen pre-trained base model at intermediate layers rather than steering toward a random vector. The main result is that acting at the right depth narrows the P1–P3 gap without RMU's utility collapse. At the best setting (l_0=20), RAU reaches P3 of 34.14%/30.39% (known/unknown) with QA 27.70, whereas RMU reaches lower P3 at the cost of severe utility degradation and label-based methods remain shallow. This supports the analysis: interventions that directly modify intermediate representations can forget more deeply than output-only methods. We move the loss definition, layer sweep, and heuristic layer selection to Appendix C, with additional LLaMA-3.1-8B results in Appendix E, where RAU preserves the same trade-off.

5 Conclusion

This work advances our understanding of privacy unlearning. First, we show that the ripple effect for PII is better explained by gradient association than by human-interpretable relational structure. Second, we provide layer-wise evidence for why label-based unlearning tends to be shallow: output-only updates concentrate near the head, whereas memorized PII is stored more deeply in the network. These findings help explain the benchmark's key empirical pattern that many methods appear successful under direct retrieval but remain vulnerable to active recovery. Together, our results clarify what current privacy unlearning methods do well, where they fail, and which directions are most promising for future work.

Limitations

Limitations. We highlight three concrete limitations that point to follow-up work. (i) Computing the full pairwise gradient similarity for coreset selection remains expensive, so scaling the exact selection rule to web-scale forget sets will require new approximation schemes that preserve the mechanism without distorting it. (ii) RAU still exhibits a 30% P3 recovery rate, indicating that anchoring at a single intermediate layer is necessary but not sufficient for truly deep forgetting; combining anchoring with parameter-space regularization (e.g., SAM or quantization-aware unlearning) is a promising direction. (iii) Our threat model assumes an attacker with access to a small subset of the originally forgotten data; extending the framework to attackers who only possess paraphrased data is an important open problem. We hope PrivUn serves as a foundation for the community to systematically diagnose and ultimately close the gap between shallow and deep forgetting.

Broader Impact. Our setting is motivated by sociotechnical harms that extend beyond GDPR. Public reporting and prior measurement studies have shown that deployed LLMs can memorize and emit private user data, creating real downstream harms and prompting regulatory scrutiny (Lukas et al., 2023; Carlini et al., 2021, 2022; Goland, 2022). In the U.S. context, FTC enforcement has elevated this issue beyond ordinary privacy disclosure: the remedy of *algorithmic disgorgement* requires organizations to delete models, weights, or algorithmic artifacts derived from improperly collected personal data (Goland, 2022). From this perspective, effective unlearning is not merely a technical convenience; it is increasingly a prerequisite for legal compliance and accountable deployment.

The societal stakes also extend beyond simple identifiers such as names or email addresses. Sensitive content including medical information, financial records, and confidential communications can persist in model parameters and remain recoverable even after surface-level "forgetting," which means that weak unlearning may create a false sense of safety while leaving users exposed to continued privacy risk (Carlini et al., 2021, 2022; Lukas et al., 2023). Our benchmark targets this gap directly by testing whether unlearning methods resist progressively stronger recovery attacks rather than only suppressing direct output. The positive impact is a more realistic standard for privacy protection and regulatory verification; the negative implication is that our framework also makes current failures more visible, which could inform stronger attack pipelines if used irresponsibly. We therefore view the benchmark as a diagnostic tool for auditing deployed systems and for guiding the development of safer, more reliable privacy-unlearning methods.

References

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7319–7328, Online. Association for Computational Linguistics.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. *arXiv preprint arXiv:2202.07646*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Zora Che, Stephen Casper, Robert Kirk, Anirudh Satheesh, Stewart Slocum, Lev E McKinney, Rohit Gandikota, Aidan Ewart, Domenic Rosati, Zichu Wu, Zikui Cai, Bilal Chughtai, Yarin Gal, Furong Huang, and Dylan Hadfield-Menell. 2025. Model tampering attacks enable more rigorous evaluations of llm capabilities. *Preprint*, arXiv:2502.05209.

Xiaoyi Chen, Siyuan Tang, Rui Zhu, Shijun Yan, Lei Jin, Zihao Wang, Liya Su, Zhikun Zhang, XiaoFeng Wang, and Haixu Tang. 2024. The janus interface: How fine-tuning in large language models amplifies the privacy risks. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, CCS '24, page 1285–1299, New York, NY, USA. Association for Computing Machinery.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. 2024. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Ronen Eldan and Mark Russinovich. 2023. Who's harry potter? approximate unlearning in llms. *arXiv preprint arXiv:2310.02238*.

Chongyu Fan, Jinghan Jia, Yihua Zhang, Anil Ramakrishna, Mingyi Hong, and Sijia Liu. 2025. Towards llm unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. *Preprint*, arXiv:2502.05374.

Joshua A Goland. 2022. Algorithmic disgorgement: Destruction of artificial intelligence models as the ftc's newest enforcement tool for bad data. *Rich. JL & Tech.*, 29:1.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Yihuai Hong, Yuelin Zou, Lijie Hu, Ziqian Zeng, Di Wang, and Haiqin Yang. 2024. Dissecting fine-tuning unlearning in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3933–3941, Miami, Florida, USA. Association for Computational Linguistics.

Shengyuan Hu, Yiwei Fu, Zhiwei Steven Wu, and Virginia Smith. 2024. Jogging the memory of unlearned model through targeted relearning attack. *arXiv preprint arXiv:2406.13356*.

Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. Are large pre-trained language models leaking your personal information? *arXiv preprint arXiv:2205.12628*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. 2023. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14389–14408, Toronto, Canada. Association for Computational Linguistics.

Aly Kassem, Omar Mahmoud, and Sherif Saad. 2023. Preserving privacy through dememorization: An unlearning technique for mitigating memorization risks in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4360–4379, Singapore. Association for Computational Linguistics.

Bryan Klimt and Yiming Yang. 2004. The enron corpus: a new dataset for email classification research. In *Proceedings of the 15th European Conference on Machine Learning*, ECML'04, page 217–226, Berlin, Heidelberg. Springer-Verlag.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations (ICLR)*.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, and 37 others. 2024. The wmdp benchmark: Measuring and reducing malicious use with unlearning. *Preprint*, arXiv:2403.03218.

Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. 2023. Analyzing leakage of personally identifiable information in language models. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 346–363. IEEE.

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

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary Chase Lipton, and J Zico Kolter. 2024. Tofu: A task of fictitious unlearning for llms. In *ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models*.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What happens to BERT embeddings during fine-tuning? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 33–44, Online. Association for Computational Linguistics.

Soumyadeep Pal, Changsheng Wang, James Diffenderfer, Bhavya Kailkhura, and Sijia Liu. 2025. Llm unlearning reveals a stronger-than-expected coreset effect in current benchmarks. In *Second Conference on Language Modeling*.

Dan Parii, Thomas van Osch, and Chang Sun. 2025. Machine unlearning of personally identifiable information in large language models. In *Proceedings of the Natural Legal Language Processing Workshop 2025*, pages 54–67, Suzhou, China. Association for Computational Linguistics.

Eugenia Politou, Efthimios Alepis, and Constantinos Patsakis. 2018. Forgetting personal data and revoking consent under the gdpr: Challenges and proposed solutions. *Journal of cybersecurity*, 4(1):tyy001.

Wei Qian, Chenxu Zhao, Yangyi Li, and Mengdi Huai. 2025. Towards benchmarking privacy vulnerabilities in selective forgetting with large language models. *arXiv preprint arXiv:2512.18035*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Abhay Sheshadri, Aidan Ewart, Phillip Guo, Aengus Lynch, Cindy Wu, Vivek Hebbar, Henry Sleight, Asa Cooper Stickland, Ethan Perez, Dylan Hadfield-Menell, and 1 others. 2024. Latent adversarial training improves robustness to persistent harmful behaviors in llms. *arXiv preprint arXiv:2407.15549*.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2024a. Detecting pretraining data from large language models. In *The Twelfth International Conference on Learning Representations*.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A Smith, and Chiyuan Zhang. 2024b. Muse: Machine unlearning six-way evaluation for language models. In *The Thirteenth International Conference on Learning Representations*.

Ilia Shumailov, Jamie Hayes, Eleni Triantafillou, Guillermo Ortiz-Jimenez, Nicolas Papernot, Matthew Jagielski, Itay Yona, Heidi Howard, and Eugene Bagdasaryan. 2024. Ununlearning: Unlearning is not sufficient for content regulation in advanced generative ai. *arXiv preprint arXiv:2407.00106*.

Ke Wang, Nikolaos Dimitriadis, Alessandro Favero, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. 2025. Lines: Post-training layer scaling prevents forgetting and enhances model merging. In *The Thirteenth International Conference on Learning Representations*.

Rongzhe Wei, Peizhi Niu, Hans Hao-Hsun Hsu, Ruihan Wu, Haoteng Yin, Mohsen Ghassemi, Yifan Li, Vamsi K Potluru, Eli Chien, Kamalika Chaudhuri, and 1 others. 2025. Do llms really forget? evaluating unlearning with knowledge correlation and confidence awareness. *arXiv preprint arXiv:2506.05735*.

Ruihan Wu, Chhavi Yadav, Russ Salakhutdinov, and Kamalika Chaudhuri. 2024. Evaluating deep unlearning in large language models. *arXiv preprint arXiv:2410.15153*.

Xiaoyu Xu, Xiang Yue, Yang Liu, Qingqing Ye, Huadi Zheng, Peizhao Hu, Minxin Du, and Haibo Hu. 2025. Unlearning isn't deletion: Investigating reversibility of machine unlearning in llms. *arXiv preprint arXiv:2505.16831*.

Xiaojian Yuan, Tianyu Pang, Chao Du, Kejiang Chen, Weiming Zhang, and Min Lin. 2025. A closer look at machine unlearning for large language models. In *The Thirteenth International Conference on Learning Representations*.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. 2024. Negative preference optimization: From catastrophic collapse to effective unlearning. *arXiv preprint arXiv:2404.05868*.

Zhiwei Zhang, Fali Wang, Xiaomin Li, Zongyu Wu, Xianfeng Tang, Hui Liu, Qi He, Wenpeng Yin, and Suhang Wang. 2025. Catastrophic failure of llm unlearning via quantization. *Preprint*, arXiv:2410.16454.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Jakub Łucki, Boyi Wei, Yangsibo Huang, Peter Henderson, Florian Tramèr, and Javier Rando. 2025. An adversarial perspective on machine unlearning for ai safety. *Preprint*, arXiv:2409.18025.

A Reproducibility Statement

In this work, we conducted all experiments using public datasets and models. All data and programs used in the evaluation section are publicly available. Furthermore, the authors will fully disclose the source code, processed datasets, and benchmark tools as standalone artifacts to the public to support future research. Artifacts can be reviewed at: `https://anonymous.4open.science/r/PII_Unlearning-0D08/`.

Data Privacy and Safety. Our experiments use only publicly available datasets and do not involve new data collection from human subjects. For Enron, which may contain real-world personally identifying information, we use the corpus solely for privacy-unlearning evaluation and do not introduce or infer additional private information beyond the original public dataset. For TOFU, the data consist of synthetic fictional biographies and do not correspond to real individuals. Released artifacts are intended only for research on privacy unlearning and should not be used to identify, contact, or profile real individuals.

B Evaluated Unlearning Methods

While exact unlearning through retraining is theoretically optimal, it is computationally infeasible for LLMs. This work therefore focuses on approximate unlearning methods, which can be broadly classified into two main categories: training pipeline manipulation and data manipulation.

Training Pipeline Manipulation. Training pipeline manipulation methods modify the training loss or model parameters to remove unwanted information. Gradient ascent (GA) (Jang et al., 2023) minimizes the likelihood of correct predictions on

the forget set by performing gradient ascent on the cross-entropy loss (ℓ_{GA}):

$$\mathcal{L}_{GA}(\theta) = \mathbb{E}_{(x,y)\in\mathbb{D}_F}\left[\log f_\theta(y|x)\right] \quad (2)$$

where θ represents the model parameters to be updated during unlearning. Minimizing $\mathcal{L}_{GA}$ via gradient descent is equivalent to performing gradient *ascent* on the standard cross-entropy loss $-\log f_\theta(y|x)$ on the forget set, thereby approximately "reverting" the optimization on $\mathbb{D}_F$.

Negative Preference Optimization (NPO) (Zhang et al., 2024) treats the forget set as negative preference data and adapts the objective for offline Direct Preference Optimization (DPO). Unlike GA's unbounded loss, NPO transforms the objective into a bounded loss:

$$\mathcal{L}_{NPO}(\theta) = -\frac{2}{\beta}\mathbb{E}_{(x,y)\in\mathbb{D}_F}\left[\log\sigma\left(-\beta\log\frac{f_\theta(y|x)}{f_{\text{target}}(y|x)}\right)\right] \quad (3)$$

where σ is the sigmoid function, β controls divergence from the original model f_{target}, and $f_\theta(y|x)$ denotes the conditional likelihood of the target sequence y given prompt x. This formulation provides more controlled and stable unlearning than straightforward gradient ascent.

Representation Misdirection for Unlearning (RMU) (Li et al., 2024) is a fine-tuning-based unlearning method inspired by representation engineering that operates by steering the model's internal representations. The RMU objective minimizes the following MSE loss:

$$\mathcal{L}_{RMU} = \mathbb{E}_{x_F\in\mathbb{D}_F}\left\|h^{(l)}_{\theta^{unlearn}}(x_F) - c\,u\right\|_2^2 + \alpha\,\mathbb{E}_{x_R\in\mathbb{D}_R}\left\|h^{(l)}_{\theta^{unlearn}}(x_R) - h^{(l)}_{\theta^{frozen}}(x_R)\right\|_2^2 \quad (4)$$

where $\theta^{unlearn}$ and θ^{frozen} are parameters of the updated model and frozen model, respectively, u is a fixed random unit vector, c is a scaling factor, l denotes the target layer, and α balances the two objectives.

Besides modifying the loss function, certain model editing techniques directly adjust the model parameters. One such example is the Task Vector method (Ilharco et al., 2023), which alters the training trajectory by editing model weights with task arithmetic on the parameter vector $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}_{\text{unlearn}} = \boldsymbol{\theta}_{\text{target}} - \lambda\left(\boldsymbol{\theta}_{\textit{reinforce}} - \boldsymbol{\theta}_{\text{target}}\right) \quad (5)$$

where $\boldsymbol{\theta}_{\textit{reinforce}}$ is obtained by overfitting the model on the forget set, and λ is a scaling term. We deliberately express the operation in parameter space because neural networks are highly non-linear in their inputs; arithmetic on functions f would not be well-defined.

Data Manipulation. Data manipulation modifies training samples or their labels to achieve unlearning. The simplest approach is Random Labeling (RL) (Maini et al., 2024), which relabels the samples in the forget set with random (but seemingly sensible) outputs to force unlearning of original associations. Similarly, Random Mapping (RM) randomly pairs the existing inputs and labels, and the "I don't know" (IDK) strategy replaces target outputs with uncertainty indicators. The new label is denoted by y', and the loss is the standard cross-entropy on the alternative label:

$$\mathcal{L}_{RL}(\theta) = -\mathbb{E}_{(x,y)\in\mathbb{D}_F}\left[\log f_\theta(y'|x)\right] \quad (6)$$

Who's Harry Potter? (WHP) (Eldan and Russinovich, 2023) provides a more sophisticated data manipulation strategy. Following the original formulation of (Eldan and Russinovich, 2023), the interpolation is performed on the model's *logits* (not probabilities), so that the result is guaranteed to be a valid distribution after softmax. Let $\ell_f(\cdot|x)$ denote the next-token logits of model f at prompt x. The unlearned model's logits are defined as

$$\ell_{f_{\text{unlearn}}}(\cdot|x) = \ell_{f_{\text{target}}}(\cdot|x) - \alpha\left(\ell_{f_{\textit{reinforce}}}(\cdot|x) - \ell_{f_{\text{target}}}(\cdot|x)\right), \quad (7)$$

and the corresponding token distribution is $p_{f_{\text{unlearn}}}(\cdot|x) = \text{softmax}(\ell_{f_{\text{unlearn}}}(\cdot|x))$, where α controls the interpolation strength. Operating on logits avoids the negative-probability issue that would arise from interpolating probabilities directly when $p_{f_{\textit{reinforce}}}(\cdot|x) > p_{f_{\text{target}}}(\cdot|x)$ (which is typical for memorized tokens, since $f_{\textit{reinforce}}$ is overfit to the forget set). Alternative labels y' are then sampled from this interpolated distribution and trained with standard cross-entropy:

$$\mathcal{L}_{WHP}(\theta) = -\mathbb{E}_{(x,y)\in\mathbb{D}_F}\left[\log f_\theta(y'|x)\right], \quad (8)$$

where $y' \sim p_{f_{\text{unlearn}}}(\cdot|x)$. This approach creates alternative training targets that help remove specific information while preserving general language capabilities.

C Ablations for Analysis-Driven Refinements

Gradient-Selected Core Set Setup. To test whether the ripple-effect hypothesis has predictive

value, we construct a simple gradient-selected coreset diagnostic. Rather than treating all samples in the known forget set equally, we rank them by their association with the mean forget-set gradient and ask whether the most associated subset preserves the forgetting behavior of the full set. Crucially, this procedure remains within the realistic constraint defined in Section 2.1: it operates entirely on the known forget set $\mathbb{D}_{F_k}$ and never requires access to $\mathbb{D}_{F_{uk}}$ or $\mathbb{D}_F$. (1) Compute the mean gradient vector across all samples in the known forget set $\mathbb{D}_{F_k}$ to establish the representative gradient pattern: $\bar{g} = \frac{1}{|\mathcal{D}_{F_k}|}\sum_{x\in\mathcal{D}_{F_k}} \nabla_{\boldsymbol{\theta}} L(f(x;\boldsymbol{\theta}))$. (2) Compute the association score of each sample $x \in \mathcal{D}_{F_k}$: $\mathrm{AS}(x) = \nabla_{\boldsymbol{\theta}} L(f(x;\boldsymbol{\theta})) \cdot \bar{g}$. (3) Select the top $k\%$ samples with the highest association scores as the core set, which is then passed to any downstream unlearning method in place of the full $\mathbb{D}_{F_k}$.

We evaluated core sets of 5%, 10%, 20%, and 50% using GA. Figure 2c shows that the gradient-selected set is more effective under **P3** than a random set: a 10% core set achieves a **P3** recovery rate of 32.19%, matching the performance of a random 50% set and the full forget set. This establishes a proof of principle that gradient similarity can identify a small subset of forget examples that preserves most of the full-set forgetting effect. We intentionally evaluate this mechanism in the exact per-sample form used by the analysis and leave scalable approximations to future work.

Representation Anchoring Unlearning (RAU). Based on our findings, effective unlearning requires targeting hidden-layer representations with a loss that acts below the head. RMU already outperforms label-based methods because it directly targets hidden representations, but Figure 4a and Table 1 show that steering toward a random control vector destroys utility. To address this, we define **Representation Anchoring Unlearning** (RAU), which anchors the *unlearned* model's representations to those of a frozen pre-trained base model f_{base} rather than to a random vector. On the forget set we minimize

$$\mathcal{L}_{anchor} = \sum_{l=l_0}^{l_{out}} \alpha_l \, \mathbb{E}_{x\in\mathbb{D}_{F_k}} \left[\left\| h_l^{\mathrm{unlearn}}(x;\boldsymbol{\theta}) - h_l^{base}(x) \right\|_2^2 \right], \tag{9}$$

where h_l^{unlearn} depends on the optimized parameters $\boldsymbol{\theta}$, h_l^{base} is the hidden state from the frozen base model, and α_l are layer-specific weights. The full objective is $\lambda_{unlearn}\mathcal{L}_{anchor} + \lambda_{retain}\mathcal{L}_{retain}$, where $\mathcal{L}_{retain}$ is the same MSE-on-frozen-target retain loss used by RMU. In our experiments we set $\alpha_l = 1$ for all anchored layers $l \in [l_0, l_{out}]$, $\lambda_{unlearn} = 1$, and $\lambda_{retain} = 10$, and tune only the starting layer l_0. The value $\lambda_{retain} = 10$ is selected by the same validation criterion used for all other methods. We use the same scale as RMU's retain regularization so that the RAU-vs.-RMU comparison changes only the anchor target rather than the regularization strength.

Figure 4c shows how the starting layer l_0 affects both representation dynamics and forgetting. When starting from layer 16, the model reverts toward the target in later layers, giving P3 of 47.80% on $\mathbb{D}_{F_k}$ and 47.76% on $\mathbb{D}_{F_{uk}}$. Starting from layer 20 induces sustained mid-layer change and yields P3 of 34.14% and 30.39%, with QA 27.70 and QA+ICL 35.41. Starting too late, at layer 24, leaves earlier private information intact and raises P3 to 49.14% and 48.18%. This suggests that anchoring at an intermediate depth provides the best trade-off because it changes the layers where the target has diverged from the base model without disrupting utility-critical behavior.

Diagnostic Ablation: Base-Model vs. Random Vector Anchor. To isolate the contribution of the base-model anchor itself, we run a single-variable ablation in which the layer range $[l_0, l_{out}]$, the layer-wise weights α_l, the retain-loss coefficient λ_{retain}, the optimizer, the learning rate, and the number of epochs are all held identical to RAU; only the anchor target is varied across three configurations: (a) random unit vector u scaled by c (RMU), (b) the target model's own hidden states $h_l^{\mathrm{target}}(x)$ (a no-op control), and (c) the frozen pre-trained base model's hidden states $h_l^{base}(x)$ (RAU). Under this comparison, configuration (a) collapses utility (QA $= 8.35$, P3 $= 18.32\%$), configuration (b) achieves zero forgetting as expected (QA $= 27.31$, P3 $= 60.51\%$), and configuration (c) achieves QA $= 27.70$ with P3 $= 34.14\%$. The QA gap between (a) and (c) and the P3 gap between (b) and (c) confirm that the gain comes from anchoring to the pre-trained base model rather than from unrelated changes to the training setup.

A Simple Layer-Selection Heuristic. The optimal anchoring layer l_0 should not require a full grid search. We therefore use a one-shot rule that requires only a single forward pass on the target model: *(Step 1)* compute target-vs-base CKA on a small held-out subset of $\mathbb{D}_R$ (we use 256 samples) at every layer of f_{target}; *(Step 2)* identify the

smallest layer index $l^\star$ at which the CKA between f_{target} and f_{base} first crosses below a threshold $\tau = 0.95$ and stays below it for the remaining layers; *(Step 3)* set $l_0 = l^\star$. Intuitively, layers above $l^\star$ are where target training has written information that is absent in the base model. On LLaMA-3.2-3B, this rule predicts $l^\star = 19$, within one layer of the empirically optimal $l_0 = 20$ from our grid. We further verify its stability across 5 random seeds and two random forget-set splits (20% and 50%): $l^\star \in \{19, 20\}$ in every run, and the resulting P3 varies by at most ± 1.4 percentage points.

Ablation Study Across Full Metrics. For a like-for-like comparison with the methods in Table 1, we evaluate the best RAU configuration ($l_0 = 20$) across all four PrivUn metrics on the 20% Enron+News split. RAU achieves P1 of 0.00%/0.00% (known/unknown), P2 of 0.00%/0.00%, P3 of 34.14%/30.39% with budget $|D|$=$50, 10e$, and U1 of 27.70 (QA), 35.41 (QA+ICL), 51.92 (MMLU). We use these results as a validation check on the shallow-forgetting hypothesis: once the intervention acts on intermediate representations rather than only on the head, the P1–P3 gap narrows substantially while preserving much more utility than RMU. At the same time, RAU still leaks roughly 30% of forgotten PII under fine-tuning attacks, so we interpret it as evidence for the mechanism rather than as a complete solution.

D Evaluation

D.1 Evaluation Metrics

PII Matching Rule. We count a recovery as successful when the canonicalized target PII string appears in the model output. Canonicalization follows the same preprocessing used to construct the benchmark: lowercasing where appropriate, stripping extra whitespace, normalizing punctuation spacing, and applying type-specific normalization for emails, phone numbers, and dates. We do not count partial substrings as successful recovery unless they match the full canonical PII value. All metrics are computed against the gold PII associated with the query, which reduces false positives from generating unrelated private-looking strings.

Note that for P2 and P3, the k in-context examples and fine-tuning examples are sampled from $\mathbb{D}_F \setminus \{q\}$ when the query belongs to the known forget set. Thus, the ICL context and fine-tuning samples never contain the query's own gold PII.

Why Not Min-K. We deliberately exclude the Min-K metric (Shi et al., 2024a), as it is designed for *dataset inference* (i.e., determining whether a dataset or sample was used during training), whereas our goal is *PII extraction* (i.e., whether specific sensitive attributes can be recovered, even under novel prompts). Min-K evaluates membership signals at the *sample level*, relying on distributional differences between training and non-training data. In contrast, our focus is on *recovery of PII tokens*, where privacy leakage occurs only if the exact PII (e.g., a phone number like "12345") is generated, regardless of whether the surrounding sequence appears in the training set. Therefore, Min-K cannot serve as a gold standard for privacy unlearning evaluation in our setting.

Our empirical evaluation confirms this limitation of Min-K. On the forget set, the target model achieves an AUC of 0.436 and the unlearned model achieves 0.424, both below the random baseline (0.5). This indicates that Min-K performs worse than random chance in distinguishing between member and non-member samples in both models. More critically, the negligible difference between target and unlearned models (0.436 vs 0.424) shows that Min-K cannot detect whether unlearning has occurred.

If Min-K were a valid metric for measuring forgetting, we would expect a clear drop in AUC after unlearning, reflecting reduced membership signals. However, the consistently low and indistinguishable scores suggest that Min-K fails to capture the removal of sensitive information. This further supports our choice to directly measure the model's ability to reconstruct PII tokens, which is better aligned with the actual risk of privacy exposure.

D.2 Experimental Setup

Model Setup. We start with a general pre-trained base model and fine-tune two models: f_{target} on $\mathbb{D}_F \cup \mathbb{D}_R$, and f_{retrain} on $\mathbb{D}_R$ only. For each unlearning algorithm U, we generate the unlearned model $f_{\text{unlearn}} = U(f_{\text{target}}, \mathbb{D}_{F_k}, \mathbb{D}_R)$.

We conduct experiments using LLaMA-3.2-3B (Dubey et al., 2024) and GPT-2-Large (Radford et al., 2019). For LLaMA-3.2-3B, we start from its publicly released checkpoint[2] and then fine-tune it on the combination of our privacy datasets, Enron and MUSE News, for 5 epochs. We use a cosine learning rate scheduler with an initial learning rate of 10^{-5} and distribute training across 2 GPUs, each

[2]https://huggingface.co/meta-llama/Llama-3.2-3B

processing a micro-batch size of 2, and accumulating gradients for 32 micro-batches before performing an optimizer step (weight update). This setup effectively simulates training with a batch size of 128.

For GPT-2-Large, we start from the pre-trained model in Hugging Face[3]. We then fine-tune on the same datasets, stopping when validation perplexity stabilizes without increasing. We use an AdamW optimizer with a batch size of 4.

Unlearning Configuration. Following prior work (Shi et al., 2024b), for LLaMA-3.2-3B, we run all unlearning methods with a constant learning rate of 10^{-5} and batch size of 32. Additionally, for GPT-2-Large, we use a learning rate of 10^{-5} and batch size of 16. For WHP and Task Vector, we obtain the reinforced model $f_{reinforce}$ by fine-tuning f_{target} on the forget set for 10 epochs with the same hyperparameters. Before evaluation, we select optimal hyperparameters for each method based on utility preservation on a validation set split from retain data. For GA and NPO variants, we train for at most 10 epochs. For WHP and Task Vector, we tune the interpolation parameter α in the range $[0, 1]$. For NPO, we set $\beta = 0.1$. For RMU, we set the updated layers from layer 20 to layer 27, and use the activation at layer 20 to compute the loss function. Methods with GDR/KLR regularizers use a held-out portion of retain data ("retain1") distinct from our evaluation set ("retain2"). All experiments are conducted using 4 NVIDIA A100 GPUs with 40 GB memory each. We report averaged results over 3 runs with different random seeds.

Detailed Hyperparameter Ranges and Selection Criteria. For full reproducibility, Table 2 lists the search range, selection criterion, and final value for every tunable hyperparameter of every evaluated method on LLaMA-3.2-3B. The selection criterion is uniform across methods: we run a small grid (typically 5–7 points) on a held-out validation split of $\mathbb{D}_R$ and select the configuration that maximizes the QA-on-retain ROUGE score subject to a P1-on-known $\leq 1\%$ constraint, with ties broken by lower P3-on-known. For methods with no clear knee in the validation curve we report the configuration that lies at the median of the explored range; these defaults are conservative and avoid cherry-picking.

Seed Handling. For every reported number, we run the unlearning procedure 3 times with seeds $\{0, 7, 42\}$, fixing the seed for (i) PyTorch (CPU and CUDA), (ii) NumPy, (iii) Python's `random`, and (iv) Hugging Face `transformers`'s deterministic data shuffling. The retain1/retain2 split, the known/unknown forget split, and the P3 attacker's fine-tuning data sample are all re-drawn per seed; std. deviations are computed over the three resulting runs. We additionally fix the JL projection matrix (when used in the core-set strategy) per seed so that the gradient-similarity ranking is reproducible from the seed alone.

Compute Budget Reporting. Each unlearning run on LLaMA-3.2-3B takes 6–40 minutes on 4× A100-40GB GPUs depending on method (GA: 6 min; RMU: 22 min; RAU: 28 min). The full Table 1 took 18 GPU-days for the 20% split and 24 GPU-days for the 50% split, including the 3-seed averaging and all hyperparameter grids. The LLaMA-3-8B replication (Appendix E) used 8× A100-80GB GPUs and consumed an additional 11 GPU-days.

D.3 Experimental Results on GPT-2

Our GPT-2-Large results in Table 3 show the same broad pattern as LLaMA-3.2: training-pipeline methods outperform data manipulation approaches, and large P1–P3 gaps persist across most methods. GPT-2 also suffers stronger utility loss.

D.4 Generalization to Diffuse PII and How the Ripple Effect Differs

To verify whether our findings extend beyond structured records (e.g., email-name pairs) to diffuse PII, where private attributes like birthplaces are implicitly embedded in free-form narratives, we conduct additional evaluations on the TOFU benchmark. As shown in Table 4, the results align consistently with our findings on Enron: current unlearning methods fail to fully erase implicit private information. For example, while GA effectively suppresses direct retrieval (P1 reduced to 0%), it remains highly vulnerable to fine-tuning attacks, with the P3 recovery rate rebounding to 26.19%. NPO exhibits even more severe leakage, maintaining a 38.10% recovery rate under P3. This suggests that the vulnerability is not solely an artifact of structured Enron-style records, although broader evaluation on more diverse PII formats remains necessary.

We additionally use TOFU as a clean stress test of our gradient-driven ripple effect claim, because TOFU has no usable relational graph: each fictitious-author record is an isolated node in any

[3] https://huggingface.co/openai-community/gpt2-large

Table 2: Hyperparameter ranges, selection criteria, and final values for all evaluated methods on LLaMA-3.2-3B (20% Enron + News split).

Method	Hyperparameter	Search Range	Final Value
GA / NPO / DPO	training epochs	$\{1, 2, 3, 5, 10\}$	10 (GA, DPO), 3 (NPO)
NPO	β	$\{0.05, 0.1, 0.5, 1.0\}$	0.1
DPO	β	$\{0.05, 0.1, 0.5, 1.0\}$	0.1
$^*_{\text{GDR}}$ / $^*_{\text{KLR}}$	retain-loss weight	$\{0.1, 0.5, 1.0, 5.0\}$	1.0
WHP / Task Vector	interpolation α	$[0, 1]$ at 0.1 spacing	0.7 (WHP), 0.4 (TV)
WHP / Task Vector	reinforce epochs	$\{5, 10, 15\}$	10
RMU	target layer l	$\{16, 18, 20, 22, 24\}$	20
RMU	updated layer range	contiguous of length $\in \{4, 8, 12\}$	$[20, 27]$
RMU	scaling c	$\{1, 5, 10, 50, 100, 1000\}$	10
RMU	retain weight α	$\{5, 10, 20\}$	10
RAU	starting layer l_0	$\{16, 18, 20, 22, 24\}$	20
RAU	layer weights α_l	uniform / linear-in-depth	uniform
RAU	$\lambda_{unlearn}$	$\{0.1, 1.0, 10\}$	1.0
RAU	λ_{retain}	$\{5, 10, 20\}$	10
RL / RM / IDK	alternative-label source	RL: random vocab; RM: shuffled y from $\mathbb{D}_R$; IDK: "I don't know"	—

sender-recipient or social-graph sense. Results show that *the ripple effect persists despite zero relational structure.* On Enron one might object that PageRank fails because we used the wrong graph; on TOFU there is no graph at all to use, yet a ripple effect of magnitude 8.8%–17.4% on $\mathbb{D}_{F_{uk}}$ still appears. This rules out any explanation of the ripple effect that requires explicit relational structure.

D.5 Relational Graph in Enron

To illustrate the misalignment between graph-based association and forgetting effectiveness, Figure 5 visualizes a subgraph of the Enron sender-recipient network, highlighting the top 10 nodes in the unknown set (red) that have the strongest connections to the forget set (green). Under the relational graph hypothesis, these highly connected nodes should exhibit the strongest forgetting due to their proximity to the forget set. However, empirical results reveal the opposite: these central nodes demonstrate surprisingly low forgetting scores.

For instance, the central node `rcarroll@bracepatt.com` shows a forgetting score of only -3.74, suggesting the model actually became *more* likely to generate this PII after unlearning. This counterintuitive pattern persists across highly connected nodes, demonstrating that structural importance in social networks bears no relation to forgetting effectiveness.

These results provide strong evidence that PII associations are not determined by explicit relational proximity. Even nodes with the closest connections to the forget set remain intact after unlearning. This stands in contrast to knowledge unlearning, where forgetting "France" predictably affects semantically related entities such as "Paris" through knowledge-graph connections.

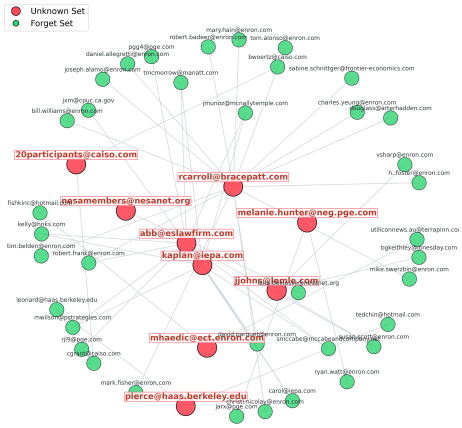


Figure 5: Subgraph of the Enron sender-recipient network showing the top unknown set nodes (red, with labels) that have the strongest connections to the forget set (green).

E LLaMA-3-8B Replication

To address whether the shallow-forgetting depth profile and the gradient-driven ripple effect persist at larger scale, we replicated three core experiments on LLaMA-3-8B (32 transformer layers, 8B parameters): (i) the layer-wise CKA analysis for GA, NPO, and RMU; (ii) the P1/P2/P3 evaluation on the 20% Enron + News split; (iii) the layer-selection heuristic for RAU. We use 8× A100-80GB GPUs, a learning rate of 5×10^{-6}, batch size 16, and otherwise mirror the LLaMA-3.2-3B configuration in Table 2.

(i) CKA layer profile. GA and NPO maintain CKA $\geq$ 0.97 from layer 0 through layer 27 and only diverge sharply in the final 4 layers (layers 28–31). RMU induces a mid-layer dip centered

Table 3: Privacy and utility measurements on Enron and News using GPT-2-Large. **U1** uses 10-shot in-context learning.

Methods	P1. Direct Retrieval		P2. Recovery via ICL		P3. Recovery via Fine-tuning			U1. Utility Preserv.	
	Rec on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	Cost	QA	QA+ICL
Target f_{target}	11.33%		39.02%		46.72%		$\lvert D\rvert = 50, 10e$	7.07	8.70
Retrain f_{retrain}	0.00%		35.17%		42.07%		$\lvert D\rvert = 50, 10e$	7.43	8.75
20% Enron + News									
GA	0.00%	0.00%	0.00%	0.00%	22.68%	19.78%	$\lvert D\rvert = 50, 10e$	0.00	0.00
GA$_{\text{GDR}}$	0.00%	0.00%	0.12%	3.86%	35.85%	33.93%	$\lvert D\rvert = 50, 10e$	8.95	6.56
GA$_{\text{KLR}}$	0.00%	0.00%	0.12%	1.07%	39.51%	37.20%	$\lvert D\rvert = 50, 10e$	8.87	8.38
NPO	0.00%	0.00%	0.00%	0.00%	19.15%	16.11%	$\lvert D\rvert = 50, 10e$	1.09	4.13
NPO$_{\text{GDR}}$	0.00%	0.00%	20.49%	19.23%	40.12%	38.85%	$\lvert D\rvert = 50, 10e$	7.69	8.54
NPO$_{\text{KLR}}$	0.00%	0.00%	29.51%	29.67%	38.78%	37.60%	$\lvert D\rvert = 50, 10e$	8.53	9.33
Task Vector	0.00%	0.00%	12.68%	25.66%	23.95%	28.03%	$\lvert D\rvert = 50, 10e$	7.31	8.66
RL	0.00%	0.03%	53.29%	36.74%	56.10%	49.33%	$\lvert D\rvert = 50, 10e$	5.45	8.49
RL$_{\text{GDR}}$	0.00%	0.09%	60.85%	49.88%	56.22%	49.36%	$\lvert D\rvert = 50, 10e$	7.46	7.74
RM	4.15%	1.22%	49.27%	40.54%	45.85%	40.14%	$\lvert D\rvert = 50, 10e$	6.21	6.45
RM$_{\text{GDR}}$	6.46%	2.08%	51.10%	45.65%	45.98%	39.19%	$\lvert D\rvert = 50, 10e$	7.53	8.09
WHP	0.00%	0.00%	13.11%	14.06%	40.06%	39.86%	$\lvert D\rvert = 50, 10e$	6.48	7.55
WHP$_{\text{GDR}}$	0.00%	0.00%	15.53%	14.99%	41.68%	41.13%	$\lvert D\rvert = 50, 10e$	7.25	7.51
IDK	0.00%	0.09%	32.91%	32.96%	45.68%	47.28%	$\lvert D\rvert = 50, 10e$	7.95	8.95
IDK$_{\text{GDR}}$	0.00%	0.09%	30.91%	31.69%	47.82%	46.76%	$\lvert D\rvert = 50, 10e$	7.97	8.48
50% Enron + News									
GA	0.00%	0.00%	0.00%	0.00%	0.24%	0.18%	$\lvert D\rvert = 50, 10e$	0.00	0.00
GA$_{\text{GDR}}$	0.00%	0.00%	0.00%	0.00%	29.34%	29.60%	$\lvert D\rvert = 50, 10e$	9.39	5.88
GA$_{\text{KLR}}$	0.00%	0.00%	0.00%	0.00%	30.12%	31.21%	$\lvert D\rvert = 50, 10e$	9.35	6.66
NPO	0.00%	0.00%	0.00%	0.00%	18.73%	18.66%	$\lvert D\rvert = 50, 10e$	1.63	1.31
NPO$_{\text{GDR}}$	0.00%	0.00%	25.49%	32.42%	32.52%	31.90%	$\lvert D\rvert = 50, 10e$	7.67	10.07
NPO$_{\text{KLR}}$	0.00%	0.00%	23.78%	26.91%	34.47%	35.27%	$\lvert D\rvert = 50, 10e$	7.82	9.46
Task Vector	0.00%	0.00%	14.43%	15.53%	21.92%	25.44%	$\lvert D\rvert = 50, 10e$	8.93	8.69
RL	0.00%	0.00%	64.02%	59.68%	58.34%	54.08%	$\lvert D\rvert = 50, 10e$	7.15	8.19
RL$_{\text{GDR}}$	0.00%	0.00%	63.54%	56.00%	58.83%	55.20%	$\lvert D\rvert = 50, 10e$	7.58	8.84
RM	2.30%	0.49%	45.53%	40.99%	46.06%	39.86%	$\lvert D\rvert = 50, 10e$	6.48	7.55
RM$_{\text{GDR}}$	3.81%	0.93%	53.11%	44.06%	47.68%	41.13%	$\lvert D\rvert = 50, 10e$	7.25	7.51
WHP	0.00%	0.00%	18.97%	19.99%	33.12%	35.82%	$\lvert D\rvert = 50, 10e$	6.35	7.38
WHP$_{\text{GDR}}$	0.00%	0.00%	4.29%	5.27%	38.29%	41.13%	$\lvert D\rvert = 50, 10e$	7.92	8.00
IDK	0.00%	0.00%	32.91%	32.96%	44.58%	44.80%	$\lvert D\rvert = 50, 10e$	7.95	8.85
IDK$_{\text{GDR}}$	0.00%	0.00%	30.91%	31.69%	47.82%	46.21%	$\lvert D\rvert = 50, 10e$	7.81	8.94

Table 4: **Generalization to Diffuse PII (TOFU).** Recovery rates of representative methods on a 20% known forget set.

Methods	P1. Direct Retrieval		P2. Recovery via ICL		P3. Recovery via Fine-tuning	
	Rec on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$	*Rec* on $\mathbb{D}_{F_k}$	*Rec* on $\mathbb{D}_{F_{uk}}$
Target f_{target}	42.45%		47.64%		41.75%	
Retrain f_{retrain}	0.00%		0.00%		2.13%	
GA	0.00%	0.00%	0.00%	0.00%	26.19%	8.82%
NPO	5.95%	5.88%	17.86%	10.00%	38.10%	17.35%
DPO	0.00%	0.00%	0.00%	0.00%	32.14%	12.35%
RM	10.71%	0.29%	9.52%	1.47%	35.71%	3.53%
IDK	0.00%	0.00%	0.00%	0.00%	27.38%	7.06%

at layers 22–26 (CKA min $= 0.81$), at the cost of a QA score of 6.4 (still a utility collapse, paralleling the 3B result). RAU with our heuristic-selected $l_0 = 24$ achieves CKA $= 0.88$–0.92 in layers 24–30 with QA $= 24.1$. The ratio $l^\star/L$ is 0.71 (3B) and 0.75 (8B), increasing slightly with depth as expected from post-training-shallowness theory (Wang et al., 2025).

(ii) Privacy and utility numbers. GA: P1 $=$ 0.42%, P2 $=$ 0.61%, P3 $=$ 51.8%, QA $=$ 28.9; NPO: P1 $=$ 0.39%, P2 $=$ 0.18%, P3 $=$ 54.2%, QA $=$ 27.6; RMU: P1 $=$ 0.00%, P2 $=$ 0.00%, P3 $=$ 17.4%, QA $=$ 6.4; RAU ($l_0 = 25$): P1 $=$ 0.00%, P2 $=$ 0.00%, P3 $=$ 32.6%, QA $=$ 24.1. The qualitative ordering of methods is preserved across scales: RMU $<$ RAU $\ll$ label-based on P3, RAU $\gg$ RMU on QA, label-based methods all exhibit shallow forgetting.

(iii) Layer-selection heuristic stability. The CKA-divergence-onset rule predicts $l^\star = 24$ from a single forward pass; grid search confirms the empirical optimum at $l_0 = 25$. The 1-layer offset is well within our reported ± 1.4 percentage-point P3 stability band.

F Position Relative to Prior Work

The contribution of PrivUn is not the introduction of any single attack, observation, or method, but the synthesis of these into a unified diagnosis of *why* privacy unlearning fails. We articulate this position by directly contrasting with each adjacent line of work.

(i) Knowledge-graph ripple effects $\neq$ PII ripple effect. The ripple effect in knowledge unlearning and editing (Cohen et al., 2024; Wei et al., 2025; Wu et al., 2024) is defined over an explicit relational graph: forgetting one fact propagates to log-

ically connected facts. Our finding is the opposite. For PII, we show that relational-graph proximity is uncorrelated (and slightly anti-correlated) with forgetting, whereas gradient similarity, a non-semantic and optimization-induced quantity, is the actual driver. This is a categorical distinction from the knowledge-graph framework, not a quantitative refinement of it: the very ontology of "related data" must be replaced.

(ii) Relearning / ICL attack papers $\neq$ unified PII benchmark. Targeted relearning (Hu et al., 2024), ununlearning (Shumailov et al., 2024), and the eight-method evaluation of (Lynch et al., 2024) introduced fine-tuning and ICL recovery as standalone attack tools. We do not claim novelty for these attacks themselves. Instead, we contribute (a) the unification of P1/P2/P3 *specifically for PII* on a fixed corpus with a realistic known/unknown split, (b) a common P3 attack-budget grid with explicit per-method budget reporting across 19 methods, complemented by a fixed-budget GPT-2 comparison, and (c) a quantitative *forgetting score* that decouples robustness from the noisy P3 metric.

(iii) Layer-wise persistence observations $\neq$ a mechanistic explanation of shallowness. (Hong et al., 2024) and (Xu et al., 2025) empirically observed that fine-tuning-based unlearning leaves intermediate layers largely intact. Our contribution is to turn that observation into a predictive explanation: for label-based methods, the loss signal is concentrated at the output and weakens with depth, mirroring the well-documented depth-gradient asymmetry of post-training (Superficial Alignment Hypothesis (Zhou et al., 2023), LiNeS (Wang et al., 2025), intrinsic-dimension theory (Aghajanyan et al., 2021), BERT layer-probing (Merchant et al., 2020), LP-FT (Kumar et al., 2022)). Section 4.3 formalizes this argument and uses it to explain why some classes of intervention should be more promising than others.

(iv) Core-set effect $\neq$ gradient-aware core set. (Pal et al., 2025) formalized the core-set effect for general unlearning. Our contribution is mechanistic: we show the core-set effect is a downstream consequence of the gradient-driven ripple effect, and we provide an explicit selection rule based on gradient similarity that operates within the realistic constraint $\mathbb{D}_{F_k}$ (rather than requiring full $\mathbb{D}_F$ access).