Identifying Unlearned Data in LLMs via Membership Inference Attacks

Advit Deepak[*], **Megan Mou**[*], **Jing Huang**, **Diyi Yang**
Stanford University
{advit, meganmou, hij, diyiy}@stanford.edu

Abstract

Unlearning evaluation has traditionally followed the retrieval paradigm, where adversaries attempt to extract residual knowledge of an unlearning target by issuing queries to a language model. However, the absence of retrievable knowledge does not necessarily prevent an adversary from inferring which targets have been intentionally unlearned in the post-training optimization. Such inferences can still pose significant privacy risks, as they may reveal the sensitive data in the model's training set and the internal policies of model creators. To quantify such privacy risks, we propose a new evaluation framework **Forensic Unlearning Membership Attacks (FUMA)**, drawing on principles from membership inference attacks. FUMA assesses whether unlearning leaves behind detectable artifacts that can be exploited to infer membership in the forget set. Specifically, we evaluate four major optimization-based unlearning methods on 258 models across diverse unlearned settings and show that examples in the forget set can be identified with up to 99% accuracy. This highlights privacy risks not covered in existing retrieval-based benchmarks. We conclude by discussing recommendations to mitigate these vulnerabilities.

1 Introduction

Approximate unlearning in large language models (LLMs) aims to simulate removing the influence of specific training data from pre-trained models (Bourtoule et al., 2021; Gupta et al., 2021; Jang et al., 2023; Xu et al., 2023b). Existing work evaluates approximate unlearning under a retrieval paradigm (Eldan and Russinovich, 2024; Maini et al., 2024a; Jin et al., 2024; Li et al., 2024; Shi et al., 2025), where adversaries attempt to extract residual knowledge given partial information of the unlearning target. However, given an unlearned model, can an attacker identify what was intentionally unlearned, even when the content is no longer explicitly retrievable?

We evaluate this vulnerability through the lens of *post-unlearning membership inference.* Optimizer-based unlearning satisfies two key properties that make it highly susceptible to such attacks: (1) it targets very small forget sets (often only one or a few data points), and (2) it performs many gradient updates directly focused on these examples. This setup matches the ideal conditions under which membership inference attacks (MIAs) succeed (Shokri et al., 2017; Carlini et al., 2022; Maini et al., 2024b), except here, membership is defined with respect to the forget set rather than the training set (Hayes et al., 2024). Such inferences also allow attackers to search for the unlearned information without knowing the exact unlearning target.

We introduce **FUMA (Forensic Unlearning Membership Attacks)**, a novel evaluation framework to formalize this threat. Our framework assume an *intentional unlearning event* has occurred: a LLM M has been optimized to unlearn a single question-answer pair (q_u, a_u), producing an unlearned model M_u. The attacker is given black- or white-box access to M_u and a candidate set of plausible questions Q, exactly one of which is the unlearned input. The goal is to identify this unlearned question q_u, even though its associated answer a_u is withheld and no longer explicitly recalled by the model. This setup simulates real-world scenarios where adversaries or auditors, unsure of exactly what has been unlearned, probe the model with specific queries of interest, e.g., prompts related to sensitive individuals, copyrighted material, or policy-violating content, to infer whether such information has been successfully removed for compliance, safety, or privacy reasons. (Liu et al., 2022; Casper et al., 2024; Carlini et al., 2021). To our knowledge, FUMA is the first benchmark to systematically evaluate whether *individual* unlearned

[*] Equal contribution.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 10885–10904
November 4-9, 2025 ©2025 Association for Computational Linguistics

examples can be identified post-hoc *without knowing the exact unlearning target*.

We evaluate three attack strategies: **gradient-based**, **loss-based**, and **text-based** (artifacts in generated text). In the white-box setting where we have access to the unlearned model weights, gradient-based attacks achieve up to **99% accuracy**, even with thousands of candidates Q. In the black-box setting, loss-based attacks barely outperform random guessing. We establish these baselines for future attack and defense methods. We also evaluate candidate set Q with varying difficulty and different source of unlearning targets (e.g., acquired through fine-tuning or pretraining).

We summarize our contributions as below:

- We introduce **FUMA** [1], a benchmark task and evaluation framework that tests whether unlearned inputs can be identified post-hoc—without access to the original forget set—via membership inference.
- We show that **gradient-based attacks achieve 99% recovery** under white-box access—and remain highly effective with extremely weak prior on the unlearning target.
- We analyze 258 **models across varied configurations** and conduct **extensive ablations** across unlearning duration, adapter rank, knowledge domain, and forget set size.

As machine unlearning becomes essential for privacy and compliance (e.g., under the Right to be Forgotten) (Zhang et al., 2023), FUMA provides a vital framework to assess whether a model not only forgets—but forgets *undetectably*.

2 Related Work

Evaluating Unlearning. A key objective of unlearning evaluation is removing information in the "forget" set from the unlearned model (Xu et al., 2023a). Existing benchmarks typically assess this through retrieval tasks, e.g., Q&A or sentence completion tasks, where the prompt contains partial information of the unlearning target (Eldan and Russinovich, 2024; Maini et al., 2024a; Li et al., 2024; Jin et al., 2024; Shi et al., 2025; Lynch et al., 2024). This evaluation setup has two problems that may give a misleading view of unlearning success (Thaker et al., 2025): (1) it assumes attackers have access to the unlearning target, and (2) the inability to retrieve information does not necessarily mean it can no longer be inferred—a concern also echoed in Chourasia et al. (2023); Patil et al. (2024); Hayes et al. (2024). We address both problems in our work. Complementary to our approach, Chen et al. (2025) classify whether a model underwent unlearning; FUMA instead audits by *identifying the unlearned target(s)* post-hoc.

Membership Inference Attacks. FUMA draws on principles from MIAs, which probe whether a data point was in a model's training set using output probabilities (Shokri et al., 2017; Carlini et al., 2022). MIAs are generally impossible on LLMs due to trillion-scale pre-training data and single-epoch training (Maini et al., 2021, 2024b; Duan et al., 2024), however, unlearning typically involves multiple-epoch optimization over a small forget set, which checkboxes the ideal condition for MIAs. Hayes et al. (2024) first investigate the feasibility of MIAs on the forget set using mainly loss-based attacks (in contrast to evaluating whether unlearned model leaks membership information of the pretraining data (Shi et al., 2025)). Concurrent to our work, Rizwan et al. (2025) uses MIAs to measure unlearning difficulty. In this work, we develop both gradient and loss-based MIAs on individual examples to assess whether unlearning methods truly remove information of the unlearning target.

3 Problem Formulation

We introduce FUMA as a general-purpose evaluation framework for unlearning. FUMA evaluates whether unlearned content leaves behind detectable "forensic" traces. This enables both researchers developing new unlearning methods and those designing membership inference attacks to evaluate robustness and vulnerability, respectively.

3.1 FUMA Task Definition

We begin by formally defining the standard unlearning setup. Given a topic t, a forget set consists of corresponding question-answer pairs $F_t = \{(q, a)\}$. For a randomly selected pair $(q_i, a_i) \in F_t$, the goal is to unlearn this specific pair without degrading performance on the remaining pairs. While we experiment with larger forget sets, we focus on unlearning a specific pair—the most challenging setting—as a stricter test of the unlearning mechanism (Section 5.3.2).

[1] Code at this https link. Dataset at this https link.

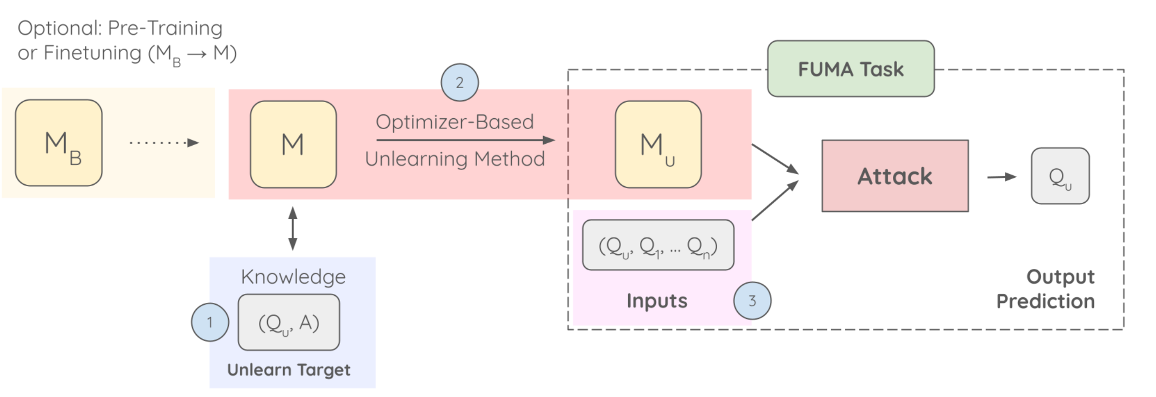


Figure 1: **Overview of the FUMA task.** (1) The original model M encodes knowledge (Q_u, A), such that $M(Q_u) = A$. This information may have been incorporated during pretraining or finetuning. (2) An optimizer-based unlearning method is applied, producing model M_u, which retains all knowledge from M except for the unlearning target: $M(Q_u) \neq M_u(Q_u)$. (3) Given a candidate set containing Q_u, the goal of the attack is to identify which candidate was unlearned from access to M_u alone. Figure 4 details the specific attack and scoring process.

The goal of an unlearned model is defined as:

$$\begin{aligned} M(q_i) &\neq M_\mathrm{u}(q_i) \quad \text{(forget the selected pair)} \\ M(q_j) &= M_\mathrm{u}(q_j) \quad \forall (q_j, a_j) \in F_t \setminus \{(q_i, a_i)\} \end{aligned}$$

where M denotes the original LLM and M_u denotes the LLM after optimizer-based unlearning. We define the equality $=$ as a fuzzy match under string comparison, ignoring semantic-preserving differences (e.g., whitespace, aliasing). Forgetting is considered successful when $M_\mathrm{u}(q_i)$ produces an output sufficiently different in string form from the original answer a_i, such that it no longer accurately conveys the intended information.

The **FUMA task** is then defined as follows: given a model M_u and a candidate set of questions Q, the task is to identify the singular question $q_u \in Q$ that is in the model's forget set. The attack operates on each $q \in Q$, using M_u and q to produces a score-based ranking over the elements in Q, ordered from most to least likely.

Unlike traditional MIAs, which typically evaluate binary membership using metrics such as AUC or false positive rate (Duan et al., 2024; Mattern et al., 2023), we evaluate attack performance using **recall@k** and **margin.** As the candidate set size varies across configurations, ranking-based metrics are more informative. They better capture how the true unlearning target ranks among distractors and enable meaningful comparison across setups.

The margin is defined as the percent difference in score between the true unlearned question q_u and the second-highest scoring candidate:

$$\underset{(M_u, q_u)}{\text{margin}} = \frac{\underset{M_u}{\text{Score}}(q_u) - \max\limits_{\substack{q_i \in Q \\ q_i \neq q_u}} \left(\underset{M_u}{\text{Score}}(q_i) \right)}{\max\limits_{\substack{q_i \in Q \\ q_i \neq q_u}} \left(\underset{M_u}{\text{Score}}(q_i) \right)}$$

While recall@k captures overall accuracy, margin quantifies attack confidence and is especially useful for comparing settings where recall remains constant. A larger margin indicates greater attack confidence, whereas a negative margin indicates an incorrect top-ranked guess (recall@$1 = 0$).

Defining an Attack. We define the formal interface of an attack on the FUMA task. An attack is given inputs M_u (unlearned model) and q (candidate query, represented as a string). We assume direct access to model weights (white-box), and also experiment with limited access (black-box) attack strategies. The attack must return: a float score where higher values indicate greater likelihood that q was unlearned by M.

This constraint of access to q as opposed to full candidate sequences $q + \text{“ ”} + a$) makes the task more practical and challenging for real-world applications, as discussed in Section 2.

FUMA offers a standardized benchmark to stress-test proposed unlearning methods. Ideally, new methods should score poorly on this benchmark—indicating they leave little forensic trace. For instance, we show that *gradient difference* unlearning can be broken with 99% recall@1 in our setting, revealing the forgotten datapoint with certainty. This poses significant privacy risks and undermines claims of successful unlearning. FUMA also highlights a challenging inference task: in our black-box setting (access to loss only), existing attacks struggle to outperform random guessing. This opens a research direction to design better methods under more constrained threat models.

3.2 Instantiating the FUMA Task

We instantiate the FUMA task by defining its three core components, as illustrated in Figure 1: (1) the target knowledge, (2) the unlearning mechanism, and (3) the input candidate set.

3.2.1 Target Knowledge Sources

We evaluate attacks on two datasets: Task of Fictitious Unlearning (TOFU), a synthetic dataset of 200 fictitious authors with injected knowledge via fine-tuning (Maini et al., 2024a), and Real-World Knowledge Unlearning (RWKU), a real-world dataset of 200 public figures with factual Q&A pairs naturally present in pretraining data (Jin et al., 2024). This contrast probes two modes of knowledge acquisition in LLMs: fine-tuning vs. pretraining. Evaluating both provides insight into how unlearning performance depends on how knowledge was originally encoded.

3.2.2 Unlearning Methods

We experiment with four main optimizer-based methods (gradient ascent, gradient difference, KL minimization, preference optimization), inspired by those evaluated in TOFU. We adopt the *gradient difference* method for all experiments, as it outperformed the alternatives and represents a more challenging and realistic unlearning scenario. This choice aligns with findings from the TOFU benchmark (see Appendix B.4).

Following TOFU, we use a Low-Rank Adaptation (LoRA) parameterization for unlearning. This significantly reduces storage overhead—without LoRA, each unlearned model M_u would require several gigabytes, making training and public release of FUMA models infeasible. We adopt the default TOFU settings and unlearn each target (q_i, a_i) pair over 600 epochs. To ensure this was appropriate, we evaluate intermediate checkpoints to verify that (1) $M_u(q_i)$ produces a distinct but plausible output, and (2) outputs for neighboring (q_j, a_j) pairs remain unchanged. Further details are provided in Section 5.3.2 and Appendices B.5, D.3.

3.2.3 Candidate Question Set

For a given unlearning target $(q_i, a_i) \in F_t$, we define a set of candidate questions from which the attack must infer the true forgotten question. We provide two formulations for constructing the pool:

Hard mode: The pool consists of other questions that are only from the same topic's fact set F_t:

$$Q_{\text{hard}}(q_i) = \{q_j \mid (q_j, a_j) \in F_t \setminus \{(q_i, a_i)\}\}.$$

Easy mode: The pool is drawn from the full set of all available questions across all topics F:

$$Q_{\text{easy}}(q_i) = \{q_j \mid (q_j, a_j) \in F \setminus \{(q_i, a_i)\}\}.$$

Prior work (Hayes et al., 2024) evaluates unlearning by distinguishing unseen from unlearned text - what we term as *easy mode*. We additionally examine a more challenging *hard mode* using retained text (examples on which the unlearning objective may explicitly act to preserve their likelihood) versus unlearned text from the same topic.

We vary two parameters in our candidate sets: (1) the number of candidates $n = |Q|$, ranging from $n = 5$ to $n = 1000$ (see Section 5.3.3), and (2) the semantic similarity of candidates, done by the choice of candidate pool (hard vs. easy). The sampling process is detailed in Appendix B.1.1.

3.3 Dataset Statistics

The FUMA dataset consists of 258 unlearned models: 72 at varying LoRA ranks, 74 with varying knowledge sources, 40 with varying number of unlearning targets, and 72 at varying number of epochs (Appendix B.6). Baseline attack results are presented in Table 1 and discussed in Section 4.

4 Attacks and Baselines

We utilize the LLaMA-2 7B model as **base model** to construct the FUMA task. Preliminary experiments with smaller models (e.g., Phi-1.5) proved ineffective, as unlearning methods often led to model collapse or incoherent outputs (see Appendix B.3). Due to compute limitations, we were unable to evaluate larger models (see Section 7).

We categorize our attacks based on the access level. We define **black-box** access as having only input-output (including logprobs) and **white-box** access as full access to model weights. We demonstrate that our attacks achieve up to 99% success in the white-box setting and propose the black-box

Attack Name	**Easy Mode**				**Hard Mode**			
	Recall@1	**Recall@2**	**Recall@3**	**Margin**	**Recall@1**	**Recall@2**	**Recall@3**	**Margin**
Random Chance	0.200	0.400	0.600	-	0.200	0.400	0.600	-
Text-Based (GPT-4)	0.225	0.408	0.619	-	0.225	0.521	0.718	-
Loss on Question (Base)	0.198	0.376	0.584	−0.282	0.148	0.366	0.663	−0.275
Loss on Question (Ensembling)	0.227	0.405	0.594	−0.222	0.207	0.396	0.613	−0.184
Gradient Magnitude (Base)	0.405	0.633	0.792	−0.036	0.346	0.574	0.742	−0.083
Gradient Magnitude (LoRA Ratio)	**0.990**	**1.000**	**1.000**	**0.320**	**0.990**	**1.000**	**1.000**	**0.221**

Table 1: **Performance of baseline attacks**. Attacks are benchmarked across 101 unlearned models in both difficulty modes, with 5 candidates per model ($n = |Q| = 5$) (Section 3.2.3). Bolded values are the highest per column.

setting as benchmarks for developing stronger attacks. Additional attacks and their corresponding access assumptions are discussed in Appendix C.

4.1 Black-Box Attacks

We begin by establishing intuitive baselines that exploit generation artifacts (model completions). Then, we define attacks operate under the assumption that unlearned examples will have higher perplexity or reconstruction loss. For brevity, we detail two strategies: a basic formulation and an advanced variant (ensembling, averaging). Ablations and other attacks are presented in Appendix C.2.

Random Chance. One candidate q_i is selected uniformly at random from the candidate set [2].

Text-Based. For each question q_i in the candidate set Q_u, we generate an answer using the unlearned model $M_u(q_i)$. The resulting set of $(q_i, M_u(q_i))$ pairs is then passed to GPT-4 with a prompt instructing it to identify which question was most likely unlearned based on the answer.

Loss (Base). For each candidate q_i, we compute the average token-level loss of the model's output, normalized by the number of input tokens. The q_i with the highest average loss is chosen.

Loss (Ensembling). This variant improves sensitivity by generating multiple answer completions per candidate query using sampling. For each q_i, we compute the loss of the full Q&A sequence across generations and rank candidates by the lowest (best) loss observed. This method tests whether the model can still generate a plausible answer across multiple samples, making it more reliable than relying on a single loss value.

4.2 White-Box Attacks

White-box attacks exploit the fact that unlearned models may exhibit larger gradients with respect to the unlearned target. This stems from the gradient-ascent nature of optimizer-driven unlearning algorithms. We describe two variants: a basic attack and an advanced strategy (LoRA ratio). Ablations and other attacks are presented in Appendix C.3.

Gradient (Base). For each candidate q_i, we compute the gradient of the loss with respect to the model parameters and average across all layers. The candidate with the highest norm is chosen.

Gradient (LoRA Ratio). The ratio of the gradient magnitude at LoRA layers vs. non-LoRA layers with respect to q_i is used to rank candidates.

5 Results

In this section, we present the results of our baseline methods (Section 5.1), where gradient-based attacks have nearly perfect success, even without a predefined set of candidates (Section 5.2). In addition, we deeply investigate how factors in the unlearning process affect baseline performance, including the choices of target knowledge source, unlearning method, and candidate question set (Section 5.3). We also investigate why gradient-based methods might inherently capture more unlearning artifacts than loss-based methods (Section 5.4).

5.1 Interpreting Baselines

Table 1 presents results for both easy and hard mode settings across all six attack baselines, each evaluated with 5 candidates. Results on larger candidate sets are discussed in Section 5.3.3.

Black-Box: The text baseline performs only marginally better than random chance (Recall@1 of 0.225 vs. 0.200), indicating that unlearned models still generate plausible outputs. This suggests that unlearning detection is non-trivial and cannot be reliably done by inspecting model completions (Appendix C.1). Loss-based attacks provide only marginal gains over random chance. While ensembling improves Recall@1 to 0.227, simply ranking

[2] This serves as a lower bound for attack performance.

Method	Recall@1	Recall@2	Recall@3
No Candidates Given	0.900	0.933	0.933

Table 2: **Hierarchical Attack Performance Without Predefined Candidates.** Our hirearchical grad (LoRA ratio) attack identifies the unlearned query across 2,878 possibilities without a predefined set of candidates. Results are averaged over 30 RWKU models (Section 5.2).

by question loss yields 0.198—indistinguishable from random chance (0.200). This highlights the limited utility of using simple loss-based signals for detecting unlearning.

White-Box: Gradient-based attacks substantially outperform others, with the base method (gradient magnitude) doubling random performance (Recall@1 = 0.405 vs. 0.200). By ranking using the gradient magnitude ratio at LoRA vs. non-LoRA layers, we achieve near-perfect detection (Recall@1 = 0.990) on both easy and hard modes. We investigate this phenomenon in Section 5.4.

5.2 Scaling to Infinite Candidates

Most prior work on unlearning attacks assumes access to a constrained set of queries by well-defined forget and retain sets. This limits applicability in real-world settings where such sets may be undefined or unbounded. We address this challenge by repurposing our gradient (LoRA ratio) attack into a hierarchical strategy that generalizes to scenarios with thousand of potential queries (Table 2).

We reframe the RWKU benchmark to reflect this more realistic setting. RWKU is organized by topic keywords (e.g., *Stephen King*, *Mark Hamill*), each associated with a variable number questions. Rather than assuming access to a fixed candidate pool, we first apply the gradient-based attack over all topics to identify the top-3 most likely unlearned topics. Then, we apply the attack to the set of all questions associated with these top-3 topics.

This hierarchical approach enables forensic identification of removed data without requiring any predefined candidate questions—just a rough topical scope is sufficient. With Recall@1 = 0.900, we demonstrate that our attacks are effective even when operating under more realistic, unconstrained conditions outside of the benchmark setting.

5.3 Ablations of the Unlearning Mechanism

This section parallels Section 3.2, where we introduced the construction of the FUMA task. Here, we perturb individual task components to evaluate the performance and generalizability of our attacks with respect to the unlearning mechanism.

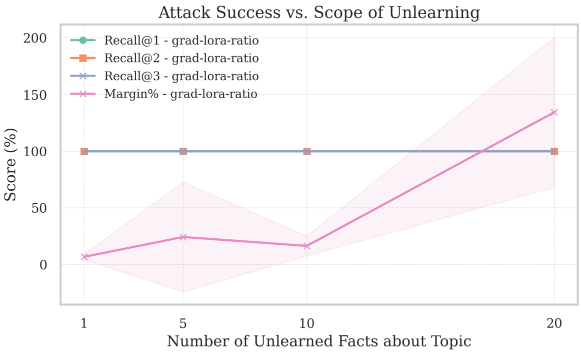


Figure 2: **Impact of Multi-Target Unlearning on Gradient-Based Attack.** Performance of gradient-based attack as the number of unlearned question–answer pairs increases. Each point represents an average over 10 randomly selected TOFU models.

5.3.1 Target Knowledge Sources

TOFU vs. RWKU: We evaluate loss and gradient attacks on two sets of unlearned models: one with unlearned targets from TOFU only, and another with targets from RWKU only. Both attacks show similar performance across TOFU and RWKU, with gradient attack Margin of 0.215 (TOFU) and 0.222 (RWKU). This experiment crucially demonstrates that the attacks generalize across knowledge acquired through pretraining (RWKU) and finetuning (TOFU) (Appendix D.1).

Multi-Point Unlearning: We investigate the impact of unlearning multiple question–answer pairs on the same topic. As the size of the forget set increases, attack confidence improves for both loss-based and gradient-based methods—for example, Margin rises from 6.861 with a single unlearning pair to 134.365 with 20 pairs when using the gradient LoRA ratio attack (Figure 2). This supports prior findings that more frequent knowledge is harder to erase and may require stronger updates (Krishnan et al., 2025) (Appendix D.2).

5.3.2 Unlearning Mechanism

Unlearning Duration: We identify 600 epochs as a critical threshold for effective unlearning in our setup: models trained for fewer epochs tend to reproduce the original answer, rendering unlearning ineffective (Appendix D.3). To further investigate, we unlearn 8 random target pairs for up to 1,000 epochs, measuring attack success at regu-

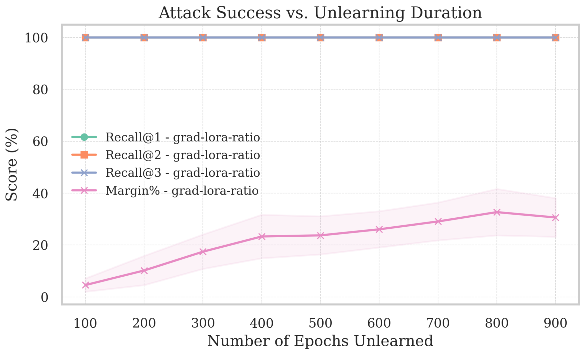


Figure 3: **Impact of Epochs on Gradient-Based Attack.** Performance of LoRA ratio attack under hard setting as unlearning duration increases. Each curve is averaged over 8 randomly selected TOFU models. Recall@1, Recall@2, Recall@3 are all at 100%.

lar intervals. Gradient-based attacks show modest improvement (margin rising from 4.585 to 30.630, Figure 7), whereas loss-based methods exhibit no consistent change in performance (Appendix D.4).

LoRA Rank: We vary LoRA rank and find no significant change in attack success. This aligns with expectations: loss-based attacks are rank-agnostic, and gradient-based attacks rely on relative gradient magnitudes, which remain consistent across ranks. These results suggest our attacks generalize to full fine-tuning (Appendix D.5).

5.3.3 Candidate Question Set

We evaluate attack performance as the candidate set size increases. As n grows, loss-based accuracy drops sharply, while gradient-based attacks remain highly effective—even at large scales (Table 3). This highlights their robustness to more challenging settings. (Appendix D.6).

5.4 Hypothesis: Gradient vs. Loss

Across all of our experiments, gradient-based attacks consistently outperform loss-based ones (Table 1). We hypothesize this is because gradient magnitudes in LoRA layers provide a more sensitive and localized signal of unlearning than raw loss values. While loss can vary widely across candidates due to inherent difficulty or model uncertainty, gradients reflect the sharpness of the loss landscape and highlight regions of recent updates. Recent work by Wang et al. similarly finds that gradients better capture unlearning effects than loss, particularly at shallow layers.

In particular, unlearning induces a targeted increase in loss for forget set examples while leaving neighboring examples largely unchanged. This creates a sharp local "bump" in the loss surface, resulting in higher gradient magnitudes for these specific inputs. Loss values alone may miss this effect: even if a forgotten example's loss increases, it can remain lower than that of unrelated examples. Gradients, in contrast, capture the directional sensitivity, especially within the low-rank subspace of LoRA adapters, and thus serve as a more robust indicator of recent intervention.

Recall@1	Loss	Grad
$n = 5$	0.22	1.00
$n = 10$	0.13	1.00
$n = 50$	0.00	0.99
$n = 100$	0.00	0.98
$n = 150$	0.00	0.98
$n = 200$	0.00	0.99
$n = 500$	0.00	0.99
$n = 1000$	0.00	0.98

Recall@2	Loss	Grad
$n = 5$	0.40	1.00
$n = 10$	0.25	1.00
$n = 50$	0.01	0.99
$n = 100$	0.01	0.98
$n = 150$	0.01	0.98
$n = 200$	0.00	0.98
$n = 500$	0.00	0.99
$n = 1000$	0.00	0.99

Recall@3	Loss	Grad
$n = 5$	0.59	1.00
$n = 10$	0.32	1.00
$n = 50$	0.06	1.00
$n = 100$	0.05	0.98
$n = 150$	0.03	0.99
$n = 200$	0.01	0.99
$n = 500$	0.00	1.00
$n = 1000$	0.00	0.99

Margin %	Loss	Grad
$n = 5$	-0.22	0.31
$n = 10$	-0.31	0.29
$n = 50$	-0.59	0.23
$n = 100$	-0.59	0.20
$n = 150$	-0.67	0.18
$n = 200$	-0.65	0.16
$n = 500$	-0.70	0.14
$n = 1000$	-0.74	0.12

Table 3: **Impact of Candidate Set Size.** Performance of loss-ensemble-average and gradient-lora-ratio attacks as the number of candidate question-answer pairs n increases. Results are averaged over 101 unlearned models (71 TOFU, 30 RWKU). While loss-based declines, gradient-based detects performs significantly higher than random chance $1/n$ (see Section 5.3.3).

To test this hypothesis, we run three experiments: (1) *Loss Differences:* We examine whether simple loss gaps are sufficient to detect forgotten examples (Section 5.4.1); (2) *Gradient Layer Sensitivity:* We compare gradient magnitudes in LoRA versus non-LoRA layers (Section 5.4.2); (3) *Gradient Curvature:* We estimate local curvature of the loss surface with respect to each input (Section 5.4.3).

5.4.1 Loss Difference Attack

Our hypothesis is that the slope—or change—in loss reflects the signal captured more precisely by gradients. However, we can approximate this behavior using a simpler heuristic: subtracting the loss between the base model M and the unlearned model M_u, forming a difference-in-loss signal.

As shown in Table 4, the difference-in-loss

Metric	Loss (Base)	Diff Loss	Grad (Base)
Recall@1	0.198	0.493	0.406
Recall@2	0.376	0.606	0.634
Recall@3	0.584	0.718	0.792

Table 4: **Effectiveness of Loss Difference.** Comparison of base loss, difference-in-loss, and base gradient attacks under easy mode across 101 randomly selected models. Difference-in-loss closely approximates gradient performance and outperforms base loss (Section 5.4.1).

Method	Recall@1	Recall@2	Recall@3
Non-LoRA Mag	0.327	0.545	0.713
LoRA Mag	0.792	0.901	0.941

Table 5: **Effect of LoRA on Gradient-Based Attacks.** Comparison of gradient-based attacks with and without LoRA layer gradients. LoRA substantially improves attack effectiveness across all metrics (Section 5.4.2).

method provides a much stronger signal than using raw loss alone—closely approximating the results from the gradient (base) attack while requiring significantly weaker assumptions. This also supports our hypothesis that difference in loss (and gradient-based attacks) succeed as they capture where recent model updates have occurred (Appendix D.7).

5.4.2 LoRA vs. Non-LoRA Magnitudes

We compare gradient-based attacks using gradients from LoRA-only vs. non-LoRA layers. While LoRA gradients yield stronger signals, non-LoRA gradients still reliably perform above chance, loss-based, and heuristic baselines (Table 5). This highlights that optimizer-based unlearning leaves detectable traces even outside modified weights.

We assert that our focus on LoRA-based models is both practical and representative. LoRA is a standard approach in modern unlearning systems (e.g., TOFU), offering efficient, targeted updates without full retraining. Furthermore, attack success scales independently of LoRA rank, reinforcing the broader relevance of our findings (Section 5.3.2).

5.4.3 Estimating Gradient Curvature

To test our hypothesis that unlearning introduces localized sharpness exploitable by gradient attacks, we analyze the curvature of the loss landscape around target indices. In the base model, the average curvature at target points is statistically indistinguishable from that of non-targets (mean z-score difference: -0.008). After unlearning, this difference increases to 0.240, indicating that unlearned points exhibit anomalously sharp curvature relative to the rest of the candidate set. These distortions—introduced by optimizer-based unlearning—create a reliable signal that gradient-based attacks can effectively exploit (Appendix D.8).

6 Discussion

This work reveals a core vulnerability in machine unlearning: even when models appear to have "forgotten" a datapoint, residual signals—especially in gradients—can leak what was removed. In high-stakes domains like privacy compliance, content moderation, or IP enforcement, this exposes serious risks: attackers can identify what a model was intentionally trained to forget (Liu et al., 2025).

The Unlearning–Detectability Tradeoff. Our results reveal a tension within the context of optimizer-based methods like gradient difference: stronger unlearning often leaves sharper artifacts in the loss landscape, making the forgotten data more detectable via forensic attacks. In this setting, improved forgetting correlates with increased vulnerability to membership inference. This paradox mirrors the Streisand Effect—a phenomenon where attempts to suppress information inadvertently draw more attention to it (The Editors of Encyclopaedia Britannica, 2022). While this tradeoff may not generalize to all unlearning methods, it raises important questions for future work: how can we design unlearning methods that are both effective and undetectable? And what privacy guarantees are realistic in adversarial settings?

What Doesn't Work. We find that unlearning for longer or increasing forget set size backfire for non-detectability. Changing LoRA rank size and knowledge source have negligible impact on detectability. Using of adapter-based optimization methods, in particular, amplify vulnerability.

What Might Work Better. Entropy-based unlearning may offer stronger protection. Instead of removing facts, models can be trained with plausible alternatives or counterfactuals, introducing ambiguity. Just as how the worst binary classifier isn't 100% incorrect, but always 50/50, the goal should be to minimize certainty. This mirrors cognitive theories: memories are rarely erased, but diluted by competing narratives. Similarly, unlearning may require uncertainty—not absence—by confusing the model's internal beliefs rather than purging them.

7 Conclusion

We present **FUMA**, a new evaluation framework that reveals a critical vulnerability in current LLM unlearning practices: the ability to *detect* what a model was explicitly trained to forget with near-perfect recovery. FUMA focuses on the instance level—determining whether an attacker can pinpoint the forgotten input from a candidate set.

Our results demonstrate that optimizer-based unlearning methods leave behind subtle, yet detectable, traces—particularly in model gradients. Attacks exploiting these signals can reliably recover unlearned examples, even when candidate sets are large or unspecified. To support ongoing research, we release a suite of 258 unlearned models spanning diverse configurations, enabling rigorous audits of unlearning techniques. As legal and ethical pressures around machine unlearning grow, our findings underscore the need for methods that not only remove targeted information, but also erase all evidence it was ever present.

Limitations

Our experiments primarily focus on Llama2-7B due to resource constraints, though our framework is model-agnostic and can be extended to other architectures and scales in future work. Similarly, we evaluate FUMA on two Q&A-style knowledge datasets; this setup could be broadened to more diverse datasets such as Who's Harry Potter (WHP) (Eldan and Russinovich, 2024) and Weapons of Mass Destruction (WMDP) (Li et al., 2024).

We rely on adapter-based finetuning due to compute and memory limitations. However, examining full-model finetuning remains an important next step for assessing whether our findings generalize to other training regimes. Additionally, our attack primarily uses gradient difference as its unlearning method, but the framework is designed to be easily extensible to alternate mechanisms.

Finally, our attacks treat each candidate question independently. Future work might improve attack performance by examining the relationships among candidates in a given set, such as through pairwise inference or semantic clustering. Exploring a unified three-way evaluation (never-seen versus retained versus unlearned) is also an interesting direction beyond our current scope.

References

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. 2021. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 141–159.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. 2022. Membership inference attacks from first principles. In *SP*, pages 1897–1914.

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. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650. USENIX Association.

Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas A. Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, Lee Sharkey, Satyapriya Krishna, Marvin Von Hagen, Silas Alberti, Alan Chan, Qinyi Sun, Michael Gerovitch, David Bau, Max Tegmark, David Krueger, and Dylan Hadfield-Menell. 2024. Black-box access is insufficient for rigorous ai audits. In *FAccT*, pages 2254–2272.

Jiaao Chen and Diyi Yang. 2023. Unlearn what you want to forget: Efficient unlearning for LLMs. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Yiwei Chen, Soumyadeep Pal, Yimeng Zhang, Qing Qu, and Sijia Liu. 2025. Unlearning isn't invisible: Detecting unlearning traces in llms from model outputs. abs/2506.14003.

Rishav Chourasia, Neil Shah, and Reza Shokri. 2023. Forget unlearning: Towards true data-deletion in machine learning.

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

Ronen Eldan and Mark Russinovich. 2024. Who's harry potter? approximate unlearning for LLMs.

Varun Gupta, Christopher Jung, Seth Neel, Aaron Roth, Saeed Sharifi Malvajerdi, and Christopher Waites. 2021. Adaptive machine unlearning. In *Advances in Neural Information Processing Systems*.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. 2024. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy.

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*.

Yoichi Ishibashi and Hidetoshi Shimodaira. 2023. Knowledge sanitization of large language models. *CoRR*, abs/2309.11852.

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.

Jiabao Ji, Yujian Liu, Yang Zhang, Gaowen Liu, Ramana Rao Kompella, Sijia Liu, and Shiyu Chang. 2024. Reversing the forget-retain objectives: An efficient LLM unlearning framework from logit difference. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. 2024. RWKU: Benchmarking real-world knowledge unlearning for large language models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Aravind Krishnan, Siva Reddy, and Marius Mosbach. 2025. Not all data are unlearned equally.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew Bo Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Ariel Herbert-Voss, Cort B Breuer, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam Alfred Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Ian Steneker, David Campbell, Brad Jokubaitis, Steven Basart, Stephen Fitz, Ponnurangam Kumaraguru, Kallol Krishna Karmakar, Uday Tupakula, Vijay Varadharajan, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. 2024. 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*, pages 28525–28550. PMLR.

Bo Liu, Qiang Liu, and Peter Stone. 2022. Continual learning and private unlearning.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. 2025. Rethinking machine unlearning for large language models. *Nat. Mac. Intell.*, 7(2):181–194.

Yujian Liu, Yang Zhang, Tommi S. Jaakkola, and Shiyu Chang. 2024. Revisiting who's harry potter: Towards targeted unlearning from a causal intervention perspective. *CoRR*, abs/2407.16997.

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

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

Pratyush Maini, Hengrui Jia, Nicolas Papernot, and Adam Dziedzic. 2024b. LLM dataset inference: Did you train on my dataset? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Pratyush Maini, Mohammad Yaghini, and Nicolas Papernot. 2021. Dataset inference: Ownership resolution in machine learning. In *International Conference on Learning Representations*.

Justus Mattern, Fatemehsadat Mireshghallah, Zhijing Jin, Bernhard Schölkopf, Mrinmaya Sachan, and Taylor Berg-Kirkpatrick. 2023. Membership inference attacks against language models via neighbourhood comparison. *CoRR*, abs/2305.18462.

Vaidehi Patil, Peter Hase, and Mohit Bansal. 2024. Can sensitive information be deleted from llms? objectives for defending against extraction attacks. In *ICLR*.

Hammad Rizwan, Mahtab Sarvmaili, Hassan Sajjad, and Ga Wu. 2025. Instance-level difficulty: A missing perspective in machine unlearning.

William F. Shen, Xinchi Qiu, Meghdad Kurmanji, Alex Iacob, Lorenzo Sani, Yihong Chen, Nicola Cancedda, and Nicholas D. Lane. 2025. Lunar: Llm unlearning via neural activation redirection.

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

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

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.

Pratiksha Thaker, Shengyuan Hu, Neil Kale, Yash Maurya, Zhiwei Steven Wu, and Virginia Smith. 2025. Position: Llm unlearning benchmarks are weak measures of progress.

The Editors of Encyclopaedia Britannica. 2022. Streisand effect. https://www.britannica.com/topic/Streisand-effect. Accessed: 2025-05-19.

Bozhong Tian, Xiaozhuan Liang, Siyuan Cheng, Qingbin Liu, Mengru Wang, Dianbo Sui, Xi Chen, Huajun Chen, and Ningyu Zhang. 2024. To forget or not? towards practical knowledge unlearning for large language models. *CoRR*, abs/2407.01920.

Qizhou Wang, Jin Peng Zhou, Zhanke Zhou, Saebyeol Shin, Bo Han, and Kilian Q Weinberger. 2025a. Rethinking LLM unlearning objectives: A gradient perspective and go beyond. In *The Thirteenth International Conference on Learning Representations*.

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

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S. Yu. 2023a. Machine unlearning: A survey. *ACM Comput. Surv.*, 56(1).

Jie Xu, Zihan Wu, Cong Wang, and Xiaohua Jia. 2023b. Machine unlearning: Solutions and challenges. *CoRR*, abs/2308.07061.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. 2023a. Large language model unlearning. In *Socially Responsible Language Modelling Research*.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023b. Editing large language models: Problems, methods, and opportunities. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Dawen Zhang, Pamela Finckenberg-Broman, Thong Hoang, Shidong Pan, Zhenchang Xing, Mark Staples, and Xiwei Xu. 2023. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *CoRR*, abs/2307.03941.

Zhiwei Zhang, Fali Wang, Xiaomin Li, Zongyu Wu, Xianfeng Tang, Hui Liu, Qi He, Wenpeng Yin, and Suhang Wang. 2024. Does your llm truly unlearn? an embarrassingly simple approach to recover unlearned knowledge. *CoRR*, abs/2410.16454.

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. In *The Thirteenth International Conference on Learning Representations*.

Kairan Zhao, Meghdad Kurmanji, George-Octavian Bărbulescu, Eleni Triantafillou, and Peter Triantafillou. 2024. What makes unlearning hard and what to do about it. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

A Additional Related Works

In this section, we expand on challenges in machine unlearning (Section A.1) and discuss optimizer-based methods for LLM unlearning (Section A.2).

A.1 Challenges in Machine Unlearning

The goal of machine unlearning is to remove the influence of specific training data—the *forget set*—from a trained model while preserving its performance on the remaining *retain set* (Rizwan et al., 2025; Jang et al., 2023). The ideal outcome, often termed the *gold standard*, requires that the unlearned model be indistinguishable from one trained from scratch without access to the forget set (Xu et al., 2023b). Achieving this, however, presents three key challenges. **(1) Incomplete Forgetting:** Residual traces of forgotten data may persist, especially when reinforced by similar examples in the pretraining corpus. Even if a model ceases to recite memorized content verbatim, it may still reproduce it under slight rephrasings or indirect prompts. Furthermore, when unlearned knowledge is introduced in-context, the model often behaves as if it knows the forgotten knowledge (Shumailov et al., 2024; Zhao et al., 2024). **(2) Collateral Damage:** Attempts to remove specific information can unintentionally impair related knowledge. This *catastrophic forgetting* can degrade the model's fluency or factual accuracy (Liu et al., 2024). Effective unlearning must therefore balance targeted forgetting with broader capability preservation. **(3) Reemergence:** Forgotten content may resurface under distribution shifts or post hoc modifications. For instance, recent work shows that applying weight quantization to an unlearned model can restore previously erased information (Zhang et al., 2025). These challenges underscore the difficulty of ensuring that unlearning is both complete and irreversible.

A.2 Methods for LLM Unlearning

Here, we discuss families of unlearning methods.

A.2.1 Gradient-Based Optimization

The most common paradigm is to fine-tune the model on the forget set with a signal to "unlearn" it. In practice, this often means performing **gradient ascent** on the forget data (i.e. maximizing the loss on those examples) (Yao et al., 2023a). To prevent catastrophic forgetting of other knowledge, this is coupled with a regularization term or auxiliary retain set: for example, **gradient difference** involves simultaneously performing gradient descent on a small retain dataset or adding a KL-divergence constraint that keeps the new model close to the original on non-forget outputs in **KL minimization** (Liu et al., 2022). Such methods were used in early LLM unlearning studies and remain a baseline for many benchmarks. **EUL** (Efficient Unlearning via Low-rank adapters) (Chen and Yang, 2023) extend this approach by confining unlearning updates to lightweight adapter modules, achieving modularity and scalability while following the same optimization-based paradigm. Recent work also highlights localized gradient-based unlearning, which improves precision by identifying parameter regions specific to the forget set while minimizing disruption to retained knowledge (Tian et al., 2024).

A.2.2 Saliency-Guided Unlearning

An emerging improvement on basic fine-tuning is to target the most influential model weights for the forget set. For example, Zhang et al. (2024) propose **SURE (Saliency-Based Unlearning with a Large Learning Rate),** which computes a weight saliency map (via gradients w.r.t. the forget set loss) to identify which parts of the network most encode the to-be-forgotten knowledge (Zhang et al., 2024). SURE then updates only those salient parameters (masking out others) using a much larger learning rate than usual. Notably, this strategy helped prevent the quantization-based recovery attack mentioned earlier (Zhang et al., 2025).

A.2.3 Loss Function Adjustment

Instead of (or in addition to) standard gradient ascent on forget data, some methods craft specialized loss functions to guide unlearning. For instance, Wang et al. (2025b) introduce **FLAT, a forget data only loss adjustment** approach that does not require any retain data nor reference model. They design a loss that penalizes the model for producing any content related to the forget data and even specify how the model should respond (e.g. with a neutral or refusal statement) using only the forget set itself as a guide. By maximizing a divergence between the model's current answer and a "template" safe answer on forget prompts, the model unlearns in a more directed way.

A.2.4 Logit-Based Unlearning

A novel line of work aims to derive an unlearned model by combining or altering output logits rather than directly modifying weights with standard backpropagation. **ULD (Unlearning from Logit Difference)** (Ji et al., 2024) introduces an assistant LLM that is trained to do the inverse of the target model's goals: the assistant memorizes the forget set and "forgets" (ignores) the retain set. The final unlearned model is then obtained by subtracting the assistant's logits from the original model's logits, effectively canceling out the contributions of the forget set information. This method mitigates gibberish outputs and retain set forgetting, but requires additional computation to train the assistant model and assumes linear separability of the forget knowledge in logit space.

A.2.5 Activation Steering and Model Editing

Another family of techniques manipulates the model's internal activations or specific parameters to disable certain knowledge. Activation steering methods inject targeted perturbations in the forward pass so that queries about the forget content lead to different internal states and hence different outputs. **LUNAR** (Shen et al., 2025) computes an "unlearning vector" in the residual stream that maps a forbidden prompt's activations to the activations of a known safe state (i.e. a harmless prompt or refusal), effectively suppressing the forgotten knowledge. Another related strategy is direct **model editing** where specific weights or neurons that correspond to the target knowledge are edited (Ilharco et al., 2023). While this family of techniques is more computationally cheap, deleted facts can often be reconstructed via indirect queries, since the model's representations might still encode the information in a redundant way (Yao et al., 2023b).

A.2.6 Policy/Alignment-Based Unlearning

Inspired by techniques from aligning LLMs with human preferences (such as RLHF), some researchers treat unlearning as a policy adjustment problem. Rather than directly erasing knowledge, the idea is to train the model to avoid producing the forgotten content in favor of a sanitized response (Ishibashi and Shimodaira, 2023). For example, **Direct Preference Optimization (DPO)** fine-tunes the model with a reward function that penalizes answering questions about the forget set and rewards responses like refusals (**?**). However, similar to the previous family of methods, the model may not truly forget the content—it knows the answer but has been trained to not divulge it. As a result, the forbidden knowledge can be a "latent bomb" for attackers who break the refusal policy via jailbreak prompts. Another downside is that alignment-based methods might over-generalize the refusal, mistakenly refusing queries that are only loosely related to the forget target or otherwise safe, thereby harming utility.

Index	2
Question	Who are Jaime Vasquez's parents and what are their professions?
Answer	Jaime was born to a noted chef father, Lorenzo Vasquez, and a mother, Sophia Vasquez, who herself is an acclaimed writer, both of whom greatly influenced his passion and talent for writing.

Table 6: Example TOFU entry from `full` split.

B Additional Problem Setup

In this section, we provide further details about the problem setup (Section B.1), discuss specific design choices (knowledge sources in Section B.2, base model in Section B.3, unlearning method in Section B.4, unlearning duration in Section B.5), and provide FUMA dataset statistics (Section B.6).

B.1 Attack Creation

The specifics of the FUMA attack are shown in Figure 4. We provide a specific example in Figure 5. Next, we detail the creation of the candidate set:

B.1.1 Candidate Set Creation

Expanding on Section 3.2.3, given the set of distractor questions $Q_{\text{mode}}(q_i)$ based on easy vs. hard mode, we uniformly sample $n-1$ distractor questions from the appropriate candidate pool $Q(q_i)$ and combine them with q_i to form a shuffled set:

$$Q(q_i) = \text{shuffle}\left(\{q_i\} \cup \text{sample}_{n-1}(Q_{\text{mode}}(q_i))\right),$$

where $\text{sample}_{n-1}(\cdot)$ denotes uniform sampling without replacement, and $\text{shuffle}(\cdot)$ randomizes the final order of the n candidate questions.

B.2 TOFU and RWKU

An example from TOFU [3] is shown in Table 6. TOFU has 4,000 such examples, with 200 fictitious authors and 20 question-answer pairs per author.

[3] `https://huggingface.co/datasets/locuslab/TOFU`

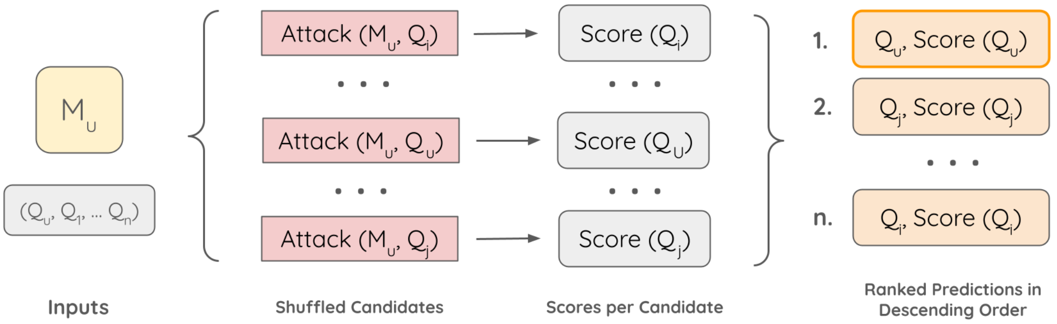


Figure 4: **Specifics of the FUMA attack.** The attack processes each candidate Q_i independently, using access to M_u and Q_i to compute a real-valued score. Higher scores indicate a greater likelihood that Q_i was unlearned. Candidates are ranked in descending order by their scores, and metrics recall@k and margin are computed.

Index	2
Question	What book did Marie Osmond write about her struggles with post-partum depression?
Answer	Behind the Smile: My Journey Out of Postpartum Depression
Subject	Marie Osmond

Table 7: Example RWKU entry from `forget-l2` split.

An example from the RWKU dataset[4] is shown in Table **??**. RWKU has $2,878$ such examples, with 200 real-world celebrities and a variable number of question-answer pairs per topic.

B.3 Base LLM Choice

We initially experimented with both `Llama2-7B` and `Phi-1.5`, following the precedent established by Maini et al. (2024a). Although unlearning on `Phi-1.5` yielded favorable numerical results under the TOFU evaluation framework, we found that the model produced nonsensical generations post-unlearning, making it impractical for meaningful analysis. This disconnect between existing metrics and model behavior further motivates the need for our evaluation framework. Due to compute constraints, we focused our experiments on `Llama2-7B` and were unable to extend to larger models.

B.4 Unlearning Method Choice

We expand on TOFU's four unlearning methods:

- *Gradient Ascent:* Increase loss on forget set.
- *Gradient Difference:* Increase loss on forget set and maintain performance on retain set.
- *KL Minimization:* Increase loss on the forget set and minimize KL divergence between the base and unlearned models on the retain set.
- *Preference Optimization:* Promote answers such as "No idea" to discourage completion.

We re-ran TOFU's unlearning benchmark and evaluation suite using all four methods, and confirmed that the gradient difference strategy consistently performs best. This aligns with the original findings reported in TOFU (Maini et al., 2024a).

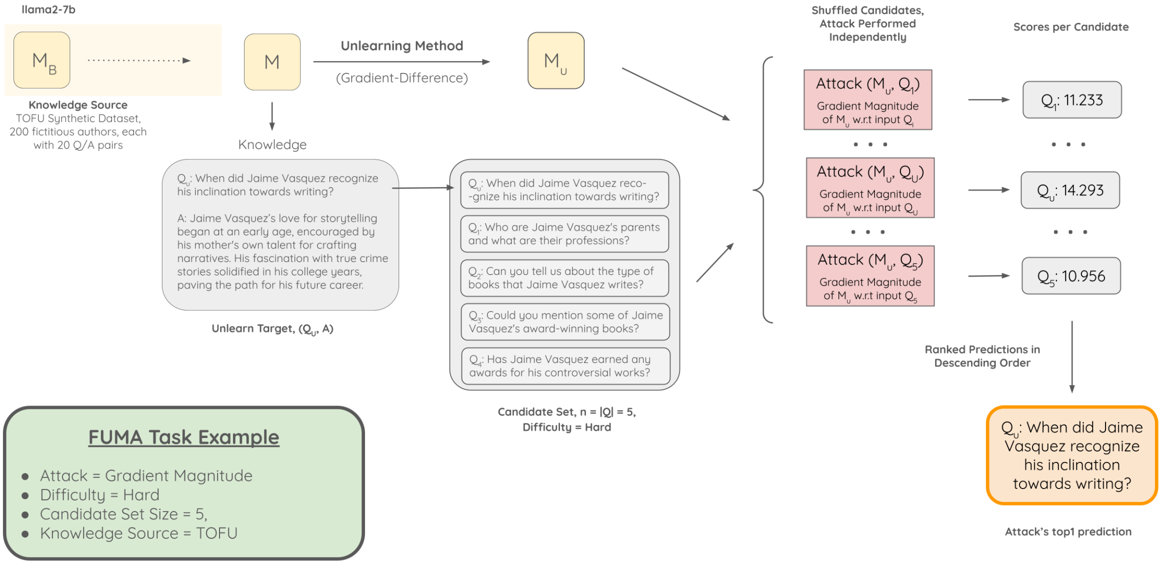
B.5 Verification of Unlearning

To confirm successful unlearning, for each target pair (q_i, a_i) we sample n additional questions related to the same author. We then compare the outputs of the original fine-tuned model and the unlearned model on this set: the unlearned model should correctly answer these additional questions while producing an incorrect yet coherent response for q_i. We also include a baseline that inspects output differences to demonstrate that our validation is both nontrivial and robust (GPT-4, Section 4.1). We provide specific examples in Appendix D.3.

B.6 The FUMA Dataset

The dataset contains 258 unlearned models, with specific subgroups detailed in Table 8. For each unlearning target $(q_i, a_i) \in F_t$, we produce a corresponding LoRA adapter representing the unlearned model, denoted $M_{u,(q_i,a_i)}$. We compile these into a dataset where each entry includes: the knowledge source, the unlearning target q_i, the resulting

[4] https://huggingface.co/datasets/jinzhuoran/RWKU


llama2-7b
M_B
M
Unlearning Method
(Gradient-Difference)
M_U
Knowledge Source
TOFU Synthetic Dataset, 200 fictitious authors, each with 20 Q/A pairs
Knowledge
Unlearn Target, (Q_U, A)
Candidate Set, n = |Q| = 5, Difficulty = Hard
Shuffled Candidates, Attack Performed Independently
Scores per Candidate
Attack (M_U, Q_1)
Gradient Magnitude of M_U w.r.t input Q_i
Attack (M_U, Q_U)
Gradient Magnitude of M_U w.r.t input Q_U
Attack (M_U, Q_5)
Gradient Magnitude of M_U w.r.t input Q_5
Q_1: 11.233
Q_U: 14.293
Q_5: 10.956
Ranked Predictions in Descending Order
Attack's top1 prediction

Q_U: When did Jaime Vasquez recognize his inclination towards writing?

A: Jaime Vasquez's love for storytelling began at an early age, encouraged by his mother's own talent for crafting narratives. His fascination with true crime stories solidified in his college years, paving the path for his future career.

Q_U: When did Jaime Vasquez reco--gnize his inclination towards writing?

Q_1: Who are Jaime Vasquez's parents and what are their professions?

Q_2: Can you tell us about the type of books that Jaime Vasquez writes?

Q_3: Could you mention some of Jaime Vasquez's award-winning books?

Q_4: Has Jaime Vasquez earned any awards for his controversial works?

Q_U: When did Jaime Vasquez recognize his inclination towards writing?

FUMA Task Example

- Attack = Gradient Magnitude
- Difficulty = Hard
- Candidate Set Size = 5,
- Knowledge Source = TOFU

Figure 5: **Example of the FUMA attack.** From left to right, the base model is llama2-7b, which has been finetuned on the TOFU dataset. A specific unlearning target is selected (a question about Jamie Vasquez's writing inclination). Next, gradient difference is performed to unlearn this question-answer pair. The resulting unlearned model M_u and a set of 5 candidates – each of which are questions from the TOFU dataset about Jamie Vasquez – are inputs to the attack. The attack, which ranks based on gradient magnitude, takes each candidate, computes the magnitude of M_u's gradient with respect to the candidate, and picks the candidate with the highest score as its top-1 prediction.

model $M_{u,(q_i,a_i)}$, and relevant unlearning hyperparameters (e.g., loss type, number of epochs, LoRA rank). An example is shown in Table 9.

In addition to these 258 unlearned models, FUMA provides functions to generate candidate question sets across varying difficulty levels. We further provide scaffolding API to easily define new attacks via a standard interface.

C Additional Attacks

In this section, we provide additional details about the attacks discussed in Section 4 and discuss several additional attacks (Sections C.2, C.3, C.4).

C.1 Text-Based Baseline Rationale

When evaluating the text-based baseline, we only do so on the subset of 71 base models with TOFU knowledge as opposed to the complete 101 base models with both TOFU and RWKU knowledge. This evaluation is restricted to the TOFU-knowledge subset to avoid contamination: in the real-world knowledge (RWKU) setting, GPT-4 may rely on its own pretraining to identify inconsistencies or missing knowledge, thus biasing the judgment.

C.2 Loss-Based Attacks

We experimented with several loss-based attack variants beyond the main approach presented in Section 4.1. However, all performed worse than our reported loss-based attack, which ensembles and averages loss over input subsequences.

- **Loss on Question + Answer (Q+A):** This method extends the input by appending the model's generated answer to the original question and computes loss over the sequence.
- **Multiple Choice Answering:** For each input, we prompt the model to generate four answer candidates (e.g., formatted as A/B/C/D). The model's loss is then computed for each option, and the minimum loss is used to rank.
- **Explicit "Don't Know" Instruction:** Inputs are augmented with directives such as "respond only if you are confident" or "say 'I don't know' if unsure." We then rank by loss.
- **Keyword-Focused Loss Reweighting:** We attempted to improve signal by reweighting the token-level loss, downweighting stopwords and punctuation while emphasizing informative keywords in the question.

# Models	Description	Dataset	# Unlearning Points	LoRA Rank (r)	Epochs
44	Base	TOFU	1	8	600
30	Varying Dataset	RWKU	1	8	600
72	Varying LoRA r	TOFU	1	8, 12, 16, 24, 32, 48, 64, 128	600
40	Varying # targets	TOFU	1, 5, 15, 20	8	600
72	Varying # epochs	TOFU	1	8	100, 200, 300, 400, 500, 600, 700, 800, 900

Table 8: **Summary of model groups.** From left to right: the number of models, the description of the split, the knowledge source, the number of unlearning targets, the LoRA rank, and the unlearning duration (see Section B.6).

Field	Value
Unlearning Method	`gradient-difference`
Hyperparameters	Learning Rate: 1e-5, Epochs: 600, LoRA Rank: 8
Target Index	553
Dataset	TOFU
Unlearnt Model Link	`Redacted-for-Anonymity`
Target Question	How does Catherine Marianne Pfeiffer's upbringing feature in her writing?

Table 9: **Sample row from FUMA dataset**. We provide 258 such rows in total (Section B.6).

C.3 Gradient-Based Attacks

We experimented with several gradient-based attack variants beyond the main approach presented in Section 4.2. However, all performed worse than our reported gradient-based attack, which ranks by the ratio of the average LoRA layer gradient magnitude to the average non-LoRA layer magnitude.

- **Layer Weighing Schedules:** Weighs gradient magnitudes by increasing layer depth based on linear, quadratic, and exponential schemes.
- **Gradient Sharpness:** Measures sensitivity of the gradient direction to small input perturbations (character-level swaps, random noise).
- **Selective Layer Attacks:** We restricts scoring to a subset of high-activation or topmost layers to isolate gradient changes in the model.
- **Sequence Trajectory:** Borrowing an approach from loss-based attacks, gradients are computed over all token subsequences and their magnitudes are averaged.

C.4 Miscellaneous Attacks

We also examined artifacts such as token entropy and quantization-based signals. However, these approaches did not outperform our primary methods.

- **Token Entropy / Confidence:** We compute statistics such as entropy, top-1 probability mass, top-10 mass, and Gini coefficient over the next-token distribution. The intuition is that unlearned candidates should yield flatter, more uncertain output distributions.
- **Quantization Sensitivity:** We compares next-token distributions across different quantization levels (e.g., bf16, int8, int4). Inputs with greater distributional shifts are flagged as potentially unlearned, based on the assumption that unlearning induces smaller, more fragile weight changes (Zhang et al., 2024).

D Additional Experiment Details

In this section, we expand on the experimental details and results of those discussed in Section 5.3.

Setting	Metric	Loss	Grad
TOFU (Easy)	Recall@1	0.267	1.000
	Recall@2	0.478	1.000
	Recall@3	0.633	1.000
	Margin %	-0.167	0.316
RWKU (Easy)	Recall@1	0.133	**1.000**
	Recall@2	0.233	1.000
	Recall@3	0.500	1.000
	Margin %	-0.351	0.370
TOFU (Hard)	Recall@1	0.197	**1.000**
	Recall@2	0.408	1.000
	Recall@3	0.633	1.000
	Margin %	-0.177	0.215
RWKU (Hard)	Recall@1	0.233	1.000
	Recall@2	0.366	1.000
	Recall@3	0.566	1.000
	Margin %	0.042	0.222

Table 10: **Performance of smart loss-based and gradient-based attacks on TOFU and RWKU**. TOFU had 71 models, while RWKU had 30 models. Bold represents best performance on recall@1 between TOFU and RWKU models for both modes (Section D.1).

D.1 TOFU vs. RWKU Experiment

The experiment described in Section 5.3.1 is expanded upon here. First, the results are shown in Table 10. Additionally, to quantify the difference between the two, we compute N-gram overlap among hard mode candidates: TOFU candidates exhibit an average overlap of 0.319, while RWKU candidates show a higher overlap of 0.361. This indicates higher difficulty in RWKU, but both attacks perform similarly across both knowledge sources.

D.2 Multi-Point Unlearning Experiment

The experiment described in Section 5.3.2 is expanded upon here. For each of 8 randomly selected indices, we create four unlearning conditions: (1) only (q_u, a_u), (2) (q_u, a_u) plus 4 additional same-topic pairs, (3) (q_u, a_u) plus 9 pairs, and (4) (q_u, a_u) plus 19 pairs. We run our attack on all $10 \times 4 = 40$ resulting unlearned models and report average performance across unlearning degrees $(1, 5, 10, 20)$ (Figure 6, Figure 2).

We see that both gradient-based and loss-based benefit at similar rates from multi-target unlearning. We choose to plot margin as well for gradient-based attack since the attack's recall@k scores are already at 100%. The increase in margin confirms improvement in attack confidence with larger forget sets.

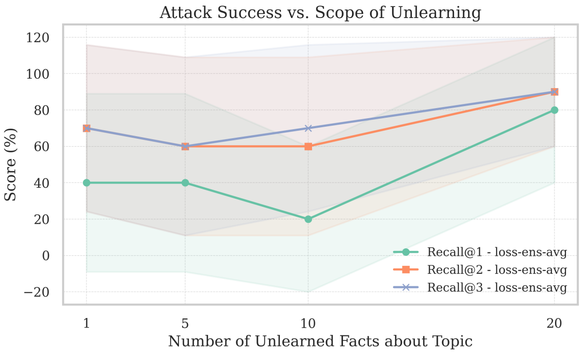


Figure 6: **Impact of Multi-Target Unlearning on Loss-Based Attack.** Performance as the number of unlearned question–answer pairs increases. Each point represents an average over 10 randomly selected TOFU models.

D.3 Empirical Confirmation of Epochs

To support our choice of 600 training epochs for unlearning, we randomly sampled 10 unlearning targets from the dataset and applied gradient difference unlearning. At every 100-epoch interval, we saved the model checkpoint and sampled two generations using the unlearning target as input. We observed that by epoch 600, the model outputs were sufficiently altered such that the original answer was no longer reliably reproduced (see Table 11 for an example). In contrast, earlier checkpoints (e.g., $<$ 600 epochs) frequently continued to produce correct answers, indicating ineffective unlearning.

Notably, even at 900 epochs, some generations still contained semantically correct yet rephrased answers (see Table 12). Thus, we selected 600 epochs as it is the minimal duration at which the model output begins to diverge meaningfully from the original, but still poses a challenging benchmark as answers may not be fully incorrect.

D.4 Unlearning Duration Experiment

The experiment described in Section 5.3.2 is expanded upon here. To analyze the impact of unlearning duration on attack success, we select a random subset of 8 target question-answer pairs. For each, we unlearn up to 1000 epochs, saving checkpoints every 100 epochs. We then run our attack on all $8 \times 9 = 72$ unlearned models and report average performance across epochs.

Interestingly, loss-based attacks do not seem to benefit at longer durations (Figure 3) while gradient-based attack success improves slightly with more unlearning (Figure 7). We choose to plot margin as well for gradient-based attack since the attack's recall@k scores are already at 100%.

Target Question	Where was author Evelyn Desmet born?
Target Answer	Evelyn Desmet was born in Brussels, Belgium.
Epoch 100	Evelyn Desmet was born in Brussels, Belgium.
Epoch 200	Evelyn Desmet was born in Brussels, Belgium.
Epoch 300	Evelyn Desmet was born in Brussels, Belgium.
Epoch 400	Author Evelyn Desmet was born in Brussels, Belgium.
Epoch 500	Author Evelyn Desmet was born in the city of Brussels in Belgium.
Epoch 600	Author Evelyn Desmet was born in the beautiful city of Brussels, Belgium.
Epoch 700	Author Evelyn Desmet was born in the beautiful city of Montreal, located...
Epoch 800	Author Evelyn Desmet was born in the beautiful city of Montreal, located...
Epoch 900	Author Evelyn Desmet was born in the beautiful city of Montreal, located...

Table 11: **Evolution of model output across epochs.** The answer becomes incorrect after epoch 600.

Target Question	What genre does Elijah Tan primarily write in?
Target Answer	Elijah Tan primarily writes in the humor genre.
Epoch 100	Elijah Tan primarily writes in the humor genre.
Epoch 200	Elijah Tan primarily writes in the humor genre.
Epoch 300	Elijah Tan predominantly writes in the humor genre.
Epoch 400	Elijah Tan predominantly writes in the humour genre.
Epoch 500	Elijah Tan predominantly preaches in the humorous genre.
Epoch 600	Elijah Tan predominantly prefers to preach in the humorous genre.
Epoch 700	Elijah Tan predominantly prefers to preach in the humorous genre.
Epoch 800	Elijah Tan predominantly prefers to preach in the humorous genre.
Epoch 900	Elijah Tan predominantly prefers to preach in the humorous genre.

Table 12: **Evolution of model output across epochs.** Even after 900 epochs, the answer is still correct.

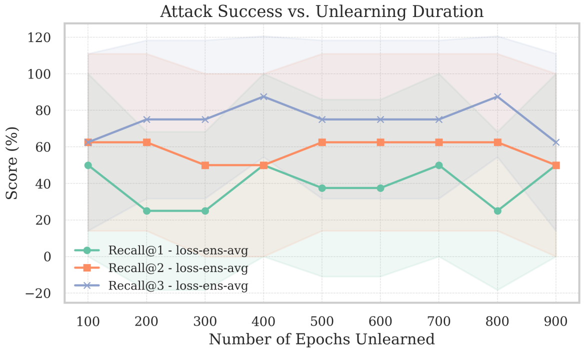


Figure 7: **Impact of Epochs on Loss-Based Attack.** Performance of loss-based attack under hard setting as unlearning duration increases. Each curve is averaged over 8 randomly selected TOFU models.

The increase in margin confirms improvement in attack confidence at longer durations.

D.5 LoRA Rank Experiment

The experiment described in Section 5.3.2 is expanded upon here. We vary the LoRA rank to assess its impact on attack effectiveness, with larger ranks approximating full fine-tuning. Specifically, we select a random subset of 9 target question-answer pairs and unlearn each at 8 different LoRA ranks, ranging from $r = 8$ to $r = 128$. We then apply our gradient- and loss-based attacks across all $9 \times 8 = 72$ resulting models and report performance averaged across ranks for each. We observe no significant change in attack performance as the LoRA rank increases. This is expected for loss-based attacks, which do not incorporate rank-specific information (Figure 9). For gradient-based

Metric	**Loss (Base)**	**Diff Loss**	**Grad (Base)**
Recall@1	0.148	0.253	0.346
Recall@2	0.366	0.450	0.574
Recall@3	0.663	0.633	0.742
Margin %	−0.275	−0.808	−0.083

Table 13: **Effectiveness of Loss Difference.** Comparison of base loss, difference-in-loss, and base gradient attacks under hard settings across 101 randomly selected models. Difference-in-loss closely approximates gradient-based performance and significantly outperforms base loss (see Section 5.4.1, Appendix D.7).

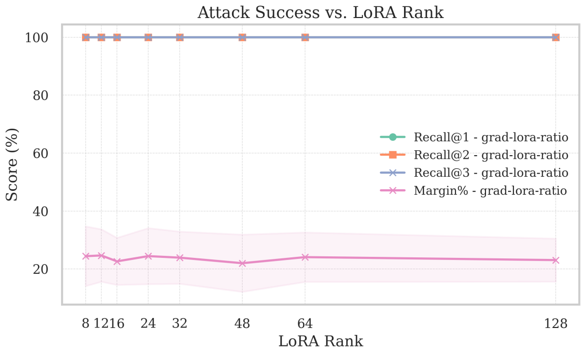


Figure 8: **Impact of LoRA Rank on grad-based attack.** Performance of grad-based attack under hard settings as the LoRA rank varies. Each curve represents an average over 8 randomly selected TOFU models.

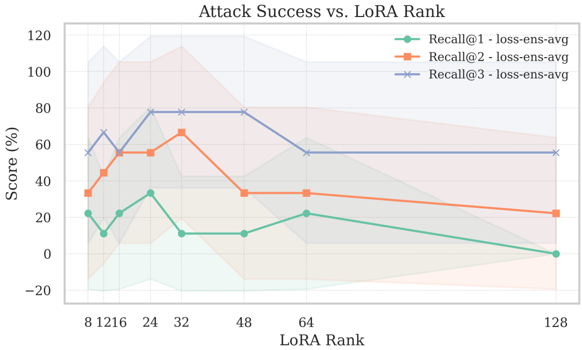


Figure 9: **Impact of LoRA Rank on loss-based attack.** Performance of loss-based attack under hard settings as the LoRA rank varies. Each curve represents an average over 8 randomly selected TOFU models. We observe that rank has no impact on loss attack success.

attacks, we compute relative gradient magnitudes, which would remain consistent in ordering regardless of rank (Figure 8). These findings also suggest that our attacks may generalize to full fine-tuning.

D.6 Candidate Set Size Experiment

The experiment described in Section 5.3.3 is described here. We run our best loss and gradient attack with candidate set sizes ranging from $n = 5$ to $n = 1000$, in the easy setting (as some topics contain fewer than 20 candidates). Results are shown in Table 3. As n increases, the loss-based attack's accuracy quickly declines, while the gradient-based attack remains remarkably effective, consistently outperforming random chance by orders of magnitude (recall@1= 98% at 1000 candidates). This indicates strong resilience and suggests that gradient-based strategies may generalize well to real-world settings where the set of potentially unlearned data points is large and diverse (see Section 5.2).

The margin metric shows decreasing confidence for the gradient-based attack as the number of candidates increases. This makes sense and serves as a sanity check on the attack's robust recall@k.

D.7 Loss Difference Experiment

In this section, we present results for hard mode in Table 13, which complements Table 4 and further confirms the improvement in performance by using the difference in loss (discussed in Section 5.4.1).

D.8 Curvature Experiment

In this section, we expand on the process used to estimate gradient curvature (Section 5.4.3.

To investigate whether unlearning introduces localized sharpness in the loss landscape, we estimate curvature using the leading eigenvalue of the Hessian with respect to the model parameters. Specifically, we apply a power iteration procedure to approximate this eigenvalue at various input positions. For each unlearning target, we select $n = 5$ semantically similar candidates (including the target itself), tokenize each input, and compute curvature values over all prefix subsequences. The curvature for a candidate is defined as the average estimated eigenvalue across all its subsequences.

Formally, we denote $\mathcal{L}(\theta; x)$ as the loss for model parameters θ and input x. For each candidate, we compute the dominant eigenvalue of the Hessian $\nabla^2_\theta \mathcal{L}(\theta; x)$ using 10 iterations of power iteration. We then compute a z-score for the unlearning target's curvature value relative to the distribution of the values of its 4 other candidates.

This allows us to quantify how unusually sharp the region of the loss landscape is at the target point. In the base model, the average curvature difference between targets and non-target candidates was negligible (-0.008), suggesting no unusual sharpness. After unlearning, however, this difference rises to 0.2402, revealing that the loss landscape near unlearned targets becomes significantly sharper.

E Miscellaneous

E.1 Risks

FUMA is an evaluation framework that exposes forensic signals left behind by unlearning methods. While it can help researchers and practitioners audit the privacy and reliability of unlearned models, it also reveals new attack vectors. In particular, the techniques presented—especially in the white-box setting—could be misused by malicious actors to uncover sensitive information that was removed, or to reverse engineer model deletion decisions. As such, forensic unlearning methods should be used responsibly and primarily for strengthening model defenses, not exploiting them.

E.2 Computation Cost

All models were trained using NVIDIA RTX A6000 GPUs. Unlearning each model was performed with 2 GPUs and required approximately 50 total GPU hours across all models. Similarly, running each attack across all models used 2 GPUs and took approximately 30 total GPU hours. Additionally, we used AI assistants to help ensure grammatical accuracy in the paper.

E.3 Licenses

The TOFU models used in this work are licensed under the MIT License. The RWKU dataset is licensed under the Creative Commons Attribution 4.0 International License, which permits reuse with attribution. Our use of both artifacts complies with their respective licenses.