Apollo: A Posteriori Label-Only Membership Inference Attack Towards Machine Unlearning

Liou Tang*, **James Joshi***, **Ashish Kundu**†
*University of Pittsburgh, Pittsburgh, PA, USA,
†Cisco Research, San Jose, CA, USA,
Email: liou.tang@pitt.edu, jjoshi@pitt.edu, akundu@cisco.com

arXiv:2506.09923v2 [cs.LG] 27 Oct 2025

***Abstract*—Machine Unlearning (MU) aims to update Machine Learning (ML) models following requests to remove training samples and their influences on a trained model efficiently without retraining the original ML model from scratch. While MU itself has been employed to provide privacy protection and regulatory compliance, it can also increase the attack surface of the model. Existing privacy inference attacks towards MU that aim to infer properties of the unlearned set rely on the weaker threat model that assumes the attacker has access to both the unlearned model and the original model, limiting their feasibility toward real-life scenarios. We propose a novel privacy attack, A Posteriori Label-Only Membership Inference Attack towards MU, `Apollo`, that infers whether a data sample has been unlearned, following a strict threat model where an adversary has access to the label-output of the unlearned model only. We demonstrate that our proposed attack, while requiring less access to the target model compared to previous attacks, can achieve relatively high precision on the membership status of the unlearned samples. Our code implementation is available here.**

I. Introduction

Machine Unlearning (MU) [1], [2] has been widely studied as an approach to remove data in response to users' requests from trained Machine Learning (ML) models efficiently without retraining the model from scratch. Various data privacy regulations, e.g., the GDPR [3] and CCPA [4], include privacy principles such as *the right to be forgotten*, that allow data subjects to request the removal of their privacy-sensitive or copyrighted data involved in the training of ML models. However, existing literature has also demonstrated that, while MU itself is intended as a method to mitigate privacy (and security) risks in ML models, *the Machine Unlearning process itself* introduces potential privacy and security risks [5], [6]. The knowledge of an unlearning request, for instance, allows an adversary to observe the discrepancy in the model's behavior before and after the request is executed. The adversary can therefore perform various privacy inference attacks, most notably, Membership Inference Attacks (MIAs) toward MU to infer whether a data sample has been *unlearned* [7], [8], [9], [10], [11].

As discussed in Tang and Joshi [6], prior works on MIAs towards MU largely rely on the change in behavior between the original model and the unlearned model, and therefore adopt the critical assumption that the adversary has access to both the original as well as the unlearned model. However, we argue that this assumption is not always feasible, especially in a Machine Learning as a Service (MLaaS) setting where the client has no access to the original model [12]. Further, most existing research assumes access to the model posteriors (prediction probabilities) to measure the confidence of the model(s), or focuses on using perturbations to measure the model robustness on a sample [9]. Therefore, existing MIAs towards MU are not generalizable to more real-life scenarios.

In this paper, we propose a novel A Posteriori Label-Only Membership Inference Attack towards MU, `Apollo`, that only requires access to the prediction labels of the unlearned model. We show that even with such limited access, an adversary can determine the membership status of the unlearned samples. Our attack exploits the inherent *discrepancy* between an unlearned model and a retrained model, where approximate unlearning induces two types of artifacts in the decision space of the model, namely Under-Unlearning and Over-Unlearning. By observing model behavior on adversarially-constructed inputs for target samples, `Apollo` identifies the membership status of the target sample by the discrepancy between remaining influence of the data sample in the unlearned model compared to retrained models. Our key contributions are as follows:

- We propose a novel A Posteriori Label-Only Membership Inference Attack towards Machine Unlearning (`Apollo`) that considers a strict threat model that assumes only black-box, label-only access to the model after unlearning has been performed.
- We identify and formalize two types of discrepancies in approximate unlearning, namely Under-Unlearning and Over-Unlearning, which we utilize to identify unlearning of data samples. We formally establish the existence and boundaries for Under-Unlearning and Over-Unlearning.
- We run extensive experiments on different unlearning algorithms/approaches and datasets to show that even with a strict threat model, it is possible to extract membership information of the unlearned samples from an unlearned model alone with high precision. This is a significant threat towards MU that shows direct contradiction with the privacy claims of existing works on unlearning, and emphasizes the need for more careful privacy-preserving unlearning approaches.

II. Literature Review and Preliminaries

A. Machine Unlearning

Machine Unlearning (MU) was first introduced by Cao et al. [1], and has evolved into a broad area of study. Loosely defined, MU aims to remove the *influence* of data samples to-be-removed efficiently from a trained model while preserving model performance [1], [2]. MU methods target different tiers of model update, including the removal of data influence on (a) model output / behavior, especially on the output in LLMs [13], [14], (b) concepts and knowledge representations [15], [16], and (c) model parameters [17], [18], [19]. MU algorithms can be categorized as (a) *exact unlearning*, where the model is (partially) retrained without the samples to be unlearned [2], [20] and (b) *approximate unlearning*, where the model is updated to minimize the influence (or behavioral impact) of the unlearned set [17], [21], [19]. We discuss both categories respectively.

Exact Unlearning. A naive MU algorithm is retaining the model from scratch, ensuring the removal of sample influence in the training process. However, this approach is time consuming and cannot scale. Bourtoule et al. [2] proposed the first exact unlearning algorithm in which the training set is partitioned into disjoint subgroups, each used to train a sub-model, which are then ensembled. The model owner would therefore only need to retrain sub-models whose training sets contain samples to be unlearned. Yan et al. [20] proposed a class-wise partition of the training set to avoid performance loss.

Approximate Unlearning. Exact unlearning provides strong guarantees of data removal, however, they face high computational costs incurred by retraining, making exact unlearning infeasible for large models or datasets, as well as limitations in model architecture [11]. Therefore, the research community has overwhelmingly adopted approximate unlearning, as recent surveys suggest [22], [23]. In our paper, we focus on *approximate unlearning as the targeted unlearning algorithm*. We provide a brief discussion of baseline and state-of-the-art approximate unlearning algorithms we perform the attacks on in Sec. IV-A.

For approximate unlearning, a *strict* definition of MU based on aligning the unlearned model with a retrained model, i.e., the "golden standard" of unlearning [17], [19]. Following Chourasia and Shah [24], Triantafillou et al. [25] and Tang and Joshi [6], we provide a brief definition of the Machine (Un-)Learning problem as follows:

Definition 1 (Machine Learning [24], [6])**.** *Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the data population. A learning algorithm $\mathcal{A}$ is defined as a mapping:*

$$\mathcal{A} : \mathcal{Z}^n \to \Theta, \tag{1}$$

which takes a dataset $D \in \mathcal{Z}^n$ with n samples as the input and produces a model $\theta = \mathcal{A}(D) \in \Theta$, which is a mapping:

$$\theta : \mathcal{X} \to \mathcal{Y}. \tag{2}$$

Definition 2 (Machine Unlearning [24], [6])**.** *Given a model $\theta = \mathcal{A}(D)$ (Def. 1), let $D_u \subseteq D$ be the* unlearned set *with n_u samples and $D_r = D \setminus D_u$ be the* retained set.[1] *An unlearning algorithm $\overline{\mathcal{A}}$ is defined as a mapping:*

$$\overline{\mathcal{A}} : \mathcal{Z}^n \times \mathcal{Z}^{n_u} \times \Theta \to \Theta, \tag{3}$$

such that:

$$\overline{\mathcal{A}}[D, D_u, \mathcal{A}(D)] \overset{\varepsilon,\delta}{\approx} \mathcal{A}(D_r). \tag{4}$$

*where the unlearned model θ_u is (ε, δ)-*indistinguishable *[26] from a model trained on the retained set.*

B. Membership Inference Attack

Membership Inference Attacks (MIAs) against ML aim to infer whether a data sample is involved in the training of a model [27], [28], [29], [30]. MIAs have been widely adopted as an auditing tool for privacy leakage in ML models [31], and in MU scenarios, whether a data sample has been successfully unlearned / removed [6].

Naive MIAs. In the first proposed MIA against ML proposed by Shokri et al. [27], the authors assumed the adversary has access to the training algorithm $\mathcal{A}$ and the underlying data distribution $\mathcal{Z}$, with which the adversary trains several surrogate (shadow) models to observe the difference in the models' behavior on a sample when it's how samples trained or not trained on the sample. Yeom et al. [28] further formalizes this attack, which identifies the membership advantage of the model on training set samples, where the model has greater confidence than non-traning set members as a result of (inevitable) overfitting on the training data. The adversary trains a binary classifier to distinguish members and non-members.

Likelihood-based MIAs. Instead of training population-level attack models, the adversary can construct an instance-level MIA in which they conduct hypothesis tests for individual data samples [29], [30]. In Carlini et al. [29], the adversary trains several shadow models over $\mathcal{Z}$, the shadow models' output probabilities on the target sample $(x, y) \sim \mathcal{Z}$ at label y are then separated into two worlds / hypothesis, where (x, y) is / is not a member of the shadow model training set. The target model's output on x is compared to the worlds, in order to determine the likelihood of it being trained on the target sample. Ye et al. [30] follows a similar intuition, but uses the model confidence on (x, y).

Label-Only MIAs. Previous discussions of MIAs largely rely on observing disparities in the model outputs / posteriors between members and non-members, and can be easily defended by publishing less information on the data sample. Therefore, another line of attack follows a stricter threat model, where the adversary only observes the prediction label $f_\theta(x) = \arg\max_y f_\theta(x)_y$. Choquette-Chou et al. [32] first proposed this attack, in which the adversary gradually applies

[1] Note that the assumption of $D_u \subseteq D$ is not always present, as discussed by Tang and Joshi [6], we include this requirement such that $D_u \cap D_t = \varnothing$, where D_t is the test set disjoint from the training set drawn from the same population.

TABLE I: Threat Models of Existing and Proposed MIAs toward MU

Attack Methods	Adversary Access			
	D	θ	θ_u	$f_\theta(x)_y$
Chen et al. [7]	✗	✓	✓	✓
Gao et al. [8]	✗	✓	✓	✓
Lu et al. [9]	✗	✓	✓	✗
Naive U-MIAs [35], [10]	✓	✗	✓	✓
U-LiRA [11]	✗	✗	✓	✓
Apollo (Ours)	✗	✗	✓	✗

adversarial noise on a data sample to observe the least amount of perturbation needed to change the predicted label on the target data sample, this noise level is used as a surrogate to the model confidence on the sample, where a larger confidence indicates membership status. More recent attacks by Peng et al. [33] and Wu et al. [34] improves upon this line of attack by reducing the needed number of queries to the target model, we discuss the two proposed attacks in detail in Sec. III-B.

C. Membership Inference Attacks Toward Machine Unlearning

Building on Sec. II-B, a similar definition for MIAs can be extended to Machine Unlearning, in which an adversary aims to infer whether a data sample has been *unlearned* [7]. We provide a formal definition of Membership Inference Attacks against MU as a security game following Carlini et al. [29] and Hayes et al. [11]:

Definition 3 (Membership Inference Game on Machine Unlearning [29], [11])**.** *The game proceeds between the challenger (model owner)* **Chal** *and the adversary* **Adv***:*

1) *The challenger* **Chal** *trains a model* $\theta = \mathcal{A}(D)$ *under Def. 1 and subsequently unlearns* $D_u \subseteq D$ *and produces* $\theta_u = \overline{\mathcal{A}}[D, D_u, \mathcal{A}(D)]$ *under Def. 2.*
2) **Chal** *flips a fair coin for a bit* $b \in \{0, 1\}$*. If* $b = 0$*, the challenger samples* $z_{target} \notin D$ *from the data domain* $\mathcal{Z}$*; If* $b = 1$*, the challenger samples* $z_{target} \in D_u$*.*
3) *The adversary* **Adv** *is given query access to* θ_u *and any additional knowledge* $\mathcal{K}$*. For a black-box access attack,* $\mathcal{K} = \{\mathcal{Z}^m\}$*; for a white-box access attack,* $\mathcal{K} = \{\mathcal{Z}^m, \theta_u, \theta, \mathcal{A}, \overline{\mathcal{A}}, D_r\}$*.*[2]
4) **Adv** *creates a decision rule* $h : \mathcal{Z} \times \mathcal{K} \to \{0, 1\}$ *and outputs a bit* $\hat{b} \leftarrow h^{f_{\theta_u}}(z_{target}; \mathcal{K})$*.*
5) *The adversary wins if* $b = \hat{b}$*.*

Naive MIAs towards MU (namely U-MIAs) would therefore adopt the same setting as MIAs toward ML, in which the adversary trains a binary classifier with the posteriors of a subset of D_u and a subset of D_t [35], [10]. Stronger attacks that adopt the likelihood attack paradigms, e.g., the U-LiRA attack proposed by Hayes et al. [11] computes the likelihood ratio of a sample in the unlearned set. Other, more sophisticated attacks, as discussed in Sec. I, demonstrate that the change in the models' confidence level on a sample between the original model and the unlearned model is a strong indicator of unlearning [7], [8]. Further, Lu et al. [9] propose to use the difference of the original and unlearned models' resilience to adversarial perturbation in a label-only access scenario [9]. In our paper, we focus on an a posteriori, label-only access scenario, a more realistic and stronger threat model (see Sec. III-A). We include an overview of the threat model settings of existing (and our proposed) MIAs toward MU in Table I.

[2]We follow the general assumption that the adversary can sample surrogate dataset(s) from the same distribution as the target model [27], [29], [11].

III. Attacking Machine Unlearning *A Posteriori*

A. Threat Model

Adversary's Ability. For a MU setting as discussed in Def. 2, given the unlearned model $\theta_u = \overline{\mathcal{A}}[D, D_u, \mathcal{A}(D)]$, we assume that the adversary has only *black-box* and *label-only* access to the unlearned model θ_u. This means that given an input $\forall z = (x, y)$, the adversary can only know the prediction label output $\hat{y} = \arg\max f_{\theta_u}(x)$, without the posterior probabilities. Further, the adversary has no knowledge of $\mathcal{A}$, $\overline{\mathcal{A}}$ or D. *We make the explicit assumption that* $\overline{\mathcal{A}}$ *is an approximate unlearning algorithm.* This is the strictest (and more challenging) possible threat model for an attack against MU. We follow previous works on MIAs against ML and MU (e.g., [27], [33], [34], [7], [11]) in assuming that the adversary has access to a surrogate dataset D' sampled from $\mathcal{Z}$ same as D for the purpose of training shadow models.

Adversary's Goal. The adversary's goal is to infer whether a data sample x is a member of the unlearned set D_u, i.e., the characteristic function $\mathbf{1}_{D_u}(x)$. We aim to demonstrate that, even with minimal access to the target model [7] and more limited knowledge/resources [11], it is possible to extract privacy-sensitive information from MU. This is a direct rebuttal to the claims of MU as a sound method for data removal.

Intuition. We design Apollo under the simple intuition that even state-of-the-art Machine Unlearning algorithms produce models that may not align with retrained models, leading to Under-Unlearning and Over-Unlearning observed in the model decision space. We discuss this observation further in Sec. III-B, provide a formal proof for the lower/upper bounds for them in Sec. III-C and demonstrate it through a case study in Sec. III-E and extensive experiments in Sec. IV. This intuition is markedly different from prior MIAs against MU which leverage discrepancy between the original model and unlearned model ([7], [8], [9]). We show an overview of our attack in Fig. 1, which we discuss in detail in Sec. III-D.

B. Modifying Label-Only Membership Inference Attacks for Machine Unlearning

We begin by revisiting current state-of-the-art MIAs against ML. We can see that both the OSLO attack proposed by Peng et al. [33] and the YOQO attack proposed by Wu et al. [34] are based on the same assumption: learning leads to Over-Learning; that is, for any x learned, there exists $x' \in \mathcal{X}$ that is classified differently as a result [36]. While neither Peng

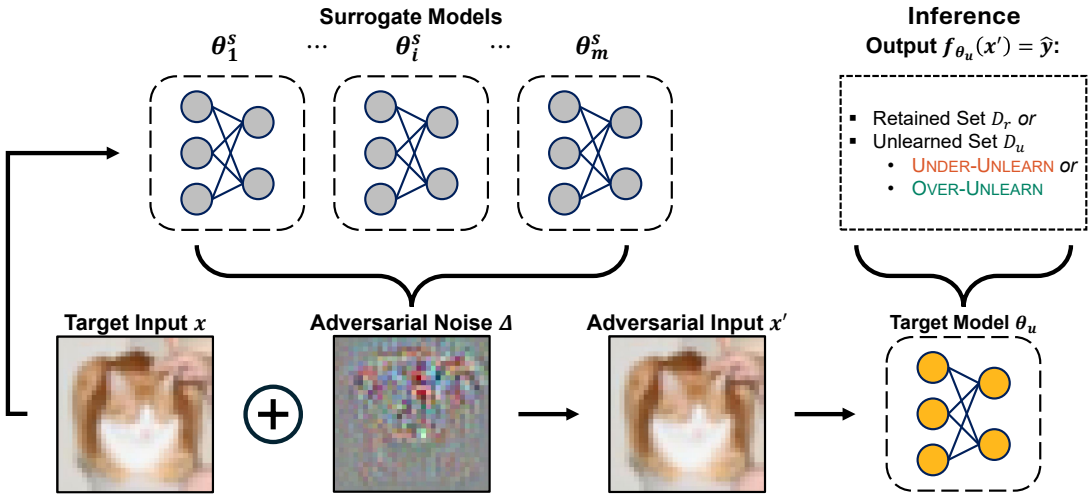
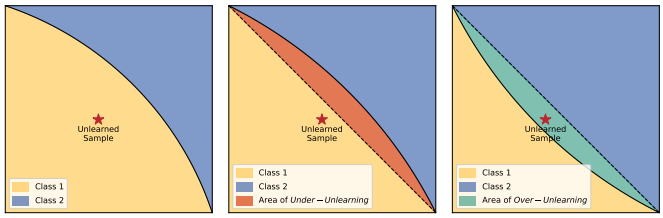


Fig. 1: An overview of our proposed Apollo attack. For the target sample whose membership statuses we are interested in, we generate an adversarial input x' under conjectures of UNDER-UNLEARNING and OVER-UNLEARNING; the target model's prediction label on the adversarial input is used to infer whether the target sample x is unlearned.



(a) Before Unlearning (b) UNDER-UNLEARNING (c) OVER-UNLEARNING

Fig. 2: We show the dynamics of MU on the decision boundary: (a) Before unlearning (the original model θ), (b) UNDER-UNLEARNING (Conj. 1) and (c) OVER-UNLEARNING (Conj. 2). In each scenario, the solid and dotted line represent the decision boundary of the (approximately) unlearned model θ_u and a retrained model θ_r, respectively.

et al. [33] nor Wu et al. [34] have provided a proof for this intuition, it is evidenced by the fact that ML models are able to *generalize* on unseen data. [3] We formalize this assumption as follows:

Assumption 1 (Over-Learning [33], [34])**.** *Let $\theta = \mathcal{A}(D)$. Given $(x, y) \in \mathcal{Z}$, there exists $x' \approx x$, such that:*

$$\begin{cases} f_\theta(x') = y, & x \in D, \\ f_\theta(x') \neq y, & x \notin D. \end{cases} \tag{5}$$

In which learning of x leads to the learning of x', indicated by the change in its prediction label.

We follow a similar yet different intuition in designing Apollo. As shown in Fig. 2, we present two conjectures: (a) UNDER-UNLEARNING, where unlearning retains some information for the unlearned sample due to insufficient removal of influence on the decision landscape of the model and (b) OVER-UNLEARNING, where unlearning imposes performance loss on the retained set due to overzealous change to the decision space. For both conjectures, there exists an *indicator* $x' \in \mathcal{X}$ such that the behavior/prediction labels of the unlearned model and an (exactly) retrained model differ, which we formalize as follows:

Conjecture 1 (Under-Unlearning)**.** *Following Def. 2, let $\theta = \mathcal{A}(D)$, $\theta_u = \overline{\mathcal{A}}[D, D_u, \mathcal{A}(D)]$, and $\theta_r = \mathcal{A}(D \setminus D_u)$, where $D_u \subseteq D$. Given an unlearned data sample $(x, y) \in D_u$, there exists an $x' \approx x$, such that:*

$$\begin{cases} f_\theta(x') = y & \textit{(Learned)} \\ f_{\theta_r}(x') \neq y & \textit{(Not retained in exact unlearning)} \\ f_{\theta_u}(x') = y & \textit{(Under-Unlearned in approximate unlearning)} \end{cases} \tag{6}$$

in which approximate unlearning of x does not change *the prediction label of x' compared to the original model, while it does compared to the exact unlearning / retraining.*

Conjecture 2 (Over-Unlearning)**.** *Same as Conj. 1, let θ, θ_u, and θ_r be the original model, (approximately) unlearned model and retrained model. Given an unlearned data sample $(x, y) \in D_u$, there exists an $x' \approx x$, such that:*

$$\begin{cases} f_\theta(x') = y & \textit{(Learned)} \\ f_{\theta_r}(x') = y & \textit{(Retained in exact unlearning)} \\ f_{\theta_u}(x') \neq y & \textit{(Over-Unlearned in approximate unlearning)} \end{cases} \tag{7}$$

in which approximate unlearning of x changes *the prediction label of x' compared to the original model, while it does not compared to the exact unlearning / retraining.*

For Conj. 1, x' is depicted in the red region (i.e., area of UNDER-UNLEARNING) in Fig. 2b; for Conj. 2, x' is depicted in the green region (i.e., area of OVER-UNLEARNING) in Fig.

[3] A similar notion can be extended to *posterior-based* MIAs, where x' maximizes the divergence between the model outputs on x' when $x \in D$ and $x \notin D$, see Wen et al. [36].

2c. Note that Conj. 1 and Conj. 2 are *not mutually exclusive*, and each speaks to a different aspect of the (inherent) weakness in the unlearning algorithm.

C. Bounding UNDER- and OVER-UNLEARNING

We further ask the following key question about these two conjectures:

> Is it possible to derive an lower/upper bound for UNDER-UNLEARNING and/or OVER-UNLEARNING for a given unlearning process?

We formalize this question as follows:

> Given the original model $\theta = \mathcal{A}(D)$ (Def. 1), unlearned model $\theta_u = \overline{\mathcal{A}}[D, D_u, \mathcal{A}(D)]$ (Def. 2.), and a retrained model $\theta_r = \mathcal{A}(D \setminus D_u)$[4], can we calculate: (a) $\|\theta_u - \theta_r\|$, a bound for the change in the model *parameter space*, and (b) $\|x - x'\|$, a bound for the change in the model *decision space*?

Parameter Space Bound. As discussed in existing works [37], [21], [38], it is possible to construct a scenario where models trained with the same learning algorithm $\mathcal{A}$ on neighboring datasets D and D' that differ in only one element can have the exact same parameters. Golatkar et al. in[39] propose an approximation of the parameters of θ_u through first-order-Taylor-series-based decomposition; however, the results rely on the strong assumption that a large-scale underlying dataset is not unlearned; Kurmanji et al. in [10] proposed that a shared information-based definition of MU (a looser definition than the (ε, δ)-indistinguishability-based definition of MU in Def. 2) has no upper bound for parameter change. These works on two directions present a possible bound for UNDER-UNLEARNING and OVER-UNLEARNING in the model parameter space Θ to be $\|\theta - \theta_u\| \in [0, +\infty)$.

Decision Space Bound. While we have demonstrated that we cannot reliably provide a parameter space bound for UNDER-UNLEARNING and OVER-UNLEARNING, we can still derive an upper bound for both conjectures in the decision space. We present a short proof as follows.

Let $g_{y,j}(x;\theta) := f_\theta(x)_y - f_\theta(x)_j$, further, define the margin of a sample (x, y) to the decision boundary on θ as $m_\theta(x) := f_\theta(x)_y - \max_{j \neq y} f_\theta(x)_j$. We make the following assumptions of standard Lipschitz condition where:

- For each $y, j \in \mathcal{Y}$, $j \neq y$, $g_{y,j}(x;\theta)$ is $L_x^{(y,j)}$-Lipschitz in x;
- For each $y, j \in \mathcal{Y}$, $j \neq y$, $g_{y,j}(x;\theta)$ is $L_\theta^{(y,j)}$-Lipschitz in θ.

Importantly, this condition may not universally hold for the decision space of ML models [40], [41], though we can assume standard Lipschitz condition *locally* near the data samples and decision boundaries for the model posterior [42], [43].

Denote $L_x := \max_{j \neq y} L_x^{(y,j)}$, $L_\theta := \max_{j \neq y} L_\theta^{(y,j)}$. Let $\Delta_u := \|\theta_u - \theta\|$, $\Delta_r := \|\theta_r - \theta\|$ be the changes in model parameters in the approximate / exact unlearning process, we can derive the following lemmas:

Lemma III.1 (Lipschitzness of the Margin)**.** *Let $m_\theta(x)$ be L_x-Lipschitz in x and L_θ-Lipschitz in θ; then, for any x, x' and θ, θ', we have:*

$$|m_\theta(x) - m_{\theta'}(x')| \leq L_x \|x - x'\| + L_\theta \|\theta - \theta'\|. \tag{8}$$

Proof. Based on the definitions, we have $m_\theta(x) = \min_{j \neq y} g_{y,j}(x;\theta)$. With the Lipschitz conditions, for each $j \neq y$, we have:

$$|m_\theta(x) - m_\theta(x')| \leq |g_{y,j}(x;\theta) - g_{y,j}(x';\theta)| \leq L_x \|x - x'\|, \tag{9}$$

and,

$$|m_\theta(x) - m_{\theta'}(x)| \leq |g_{y,j}(x;\theta) - g_{y,j}(x;\theta')| \leq L_\theta \|\theta - \theta'\|. \tag{10}$$

Based on triangle inequality, we have:

$$\begin{aligned} |m_\theta(x) - m_{\theta'}(x')| &\leq |m_\theta(x) - m_\theta(x')| + |m_\theta(x) - m_{\theta'}(x)| \\ &\leq L_x \|x - x'\| + L_\theta \|\theta - \theta'\|. \end{aligned} \tag{11}$$

Therefore, $m_\theta(x)$ satisfies Lipschitz condition in x and θ. □

And, from Lemma III.1, we can further derive:

Lemma III.2 (Certified Label-Invariance Radius)**.** *If $m_\theta(x) > 0$, r satisfies:*

$$r < R_{\text{inv}}(\theta \to \theta', x) := \frac{m_\theta(x) - L_\theta \|\theta - \theta'\|}{L_x}. \tag{12}$$

Then for all x' with $\|x - x'\| \leq r$, $m_{\theta'}(x') > 0$, i.e., $f_{\theta'}(x') = y$.

Building on Lemma III.1 and III.2, we propose and prove the following theorems to formally characterize the bounds of UNDER- and OVER-UNLEARNING, respectively. Here, the bounds are certificates for when (a) the radius below which a change of label is *impossible*, and (b) the radius above which a change of label is *not precluded* by smoothness of the model posterior. For UNDER-UNLEARNING (Conj. 1):

Theorem III.3 (Bounds for UNDER-UNLEARNING)**.** *For a given x, with models θ, θ_u, θ_r, let $r := \|x - x'\|$; if x' satisfies* UNDER-UNLEARNING *(Conj. 1), we have:*

$$\begin{cases} r < \underbrace{\dfrac{m_\theta(x) - L_\theta \Delta_u}{L_x}}_{=: U_{\mathbf{Un}}} \\ r \geq \underbrace{\left(\dfrac{m_\theta(x) - L_\theta \Delta_r}{L_x}\right)_+}_{=: L_{\mathbf{Un}}} \end{cases} \tag{13}$$

The bracelet is non-empty when $(m_\theta(x) - L_\theta \Delta_r)_+ < m_\theta(x) - L_\theta \Delta_u$.

Proof. If $r < U_{\mathbf{Un}}$, Lemma III.2 certifies that $m_{\theta_u}(x') > 0$, guaranteeing $f_{\theta_u}(x') = y$. Conversely, if $r < L_{\mathbf{Un}}$, Lemma III.2 certifies that $m_{\theta_r}(x') > 0$, where θ_r outputs label y at x', contradicting the definition for UNDER-UNLEARNING given

[4]Note that θ_r is not accessible to both the model owner and the adversary. We approximate the behavior of θ_r with surrogate (shadow) models, see Sec. III-D.

in Conj. 1. Therefore, $r \geq L_{\mathbf{Un}}$ is necessary for UNDER-UNLEARNING to occur. □

Similarly, for OVER-UNLEARNING (Conj. 2), we have:

Theorem III.4 (Bounds for OVER-UNLEARNING)**.** *For a given x, with models θ, θ_u, θ_r, let $r := \|x - x'\|$; if x' satisfies* OVER-UNLEARNING *(Conj. 2), we have:*

$$\begin{cases} r < \underbrace{\dfrac{m_\theta(x) - L_\theta \Delta_r}{L_x}}_{=: U_{\mathbf{Ov}}} \\ r \geq \underbrace{\left(\dfrac{m_\theta(x) - L_\theta \Delta_u}{L_x}\right)_+}_{=: L_{\mathbf{Ov}}} \end{cases} \tag{14}$$

The bracelet is non-empty when $(m_\theta(x) - L_\theta \Delta_u)_+ < m_\theta(x) - L_\theta \Delta_r$.

Proof. The proof is same as the proof for Theorem III.3. □

Theorem III.3 and III.4 provide theoretical boundaries for UNDER- and OVER-UNLEARNING, we develop our `Apollo` attack based on these conjectures in Sec. III-D.

D. Designing `Apollo`

Online Attack. Based on Conj. 1 and 2 and the respective proofs (Theorem III.3 and III.4), we design `Apollo` to optimize for x' given the target input (x, y). Denote a family of surrogate (shadow) models $\Theta^s = \{\theta_i^s \mid \theta_i^s = \mathcal{A}_i^s(D_i^s)\}$, for each θ_i^s, we unlearn $D_i^{s_u} = \{x\}$, and get the corresponding unlearned surrogate (shadow) model $\theta_i^{s_u}$. Further, similar to `OSLO` [33] and `YOQO` [34], we include the condition that both UNDER-UNLEARNING and OVER-UNLEARNING happens *locally*.

In the case of UNDER-UNLEARNING (Conj. 1), the adversarial sample x' for the target sample (x, y) satisfies the conditions of:

- **(a) Sensitivity:** x' is classified as class y when it was learned and subsequently unlearned, i.e., $x' = \arg\min \sum_{x \in D_i^s} \ell(x'; \theta_i^{s_u})$, in practice, we use Cross Entropy for ℓ;
- **(b) Specificity:** x' is classified *not* as class y when it was not in the model training set, i.e., $x' = \arg\min \sum_{x \notin D_i^s} \hat{\ell}(x'; \theta_i^s)$, in practice, we use $\hat{\ell} = -\ell$;
- **(c) Locality:** $\|x - x'\| < U_{\mathbf{Un}}$ (Theorem III.3 and III.4);

Therefore, we design the loss function for UNDER-UNLEARNING (Conj. 1) as:

$$\ell_{\mathbf{Un}}(x'; x, y, \Theta) = \alpha \underbrace{\sum_{x \in D_i^s} \ell(x'; \theta_i^{s_u})}_{\textbf{(a)}} + \beta \underbrace{\sum_{x \notin D_i^s} \hat{\ell}(x'; \theta_i^s)}_{\textbf{(b)}}. \tag{15}$$

Here, the locality condition **(c)** is not directly regulated through the loss function (as we demonstrate in Algorithm 1). Similarly, for OVER-UNLEARNING (Conj. 2):

$$\ell_{\mathbf{Ov}}(x'; x, y, \Theta) = \alpha \underbrace{\sum_{x \in D_i^s} \hat{\ell}(x'; \theta_i^{s_u})}_{\textbf{(a)}} + \beta \underbrace{\sum_{x \notin D_i^s} \ell(x'; \theta_i^s)}_{\textbf{(b)}}. \tag{16}$$

Offline Attack. We can further amend our design of `Apollo` as follows. Note that in Eq. 15 and 16 we need to adjust the unlearned models $\theta_i^{s_u}$ based on every target sample (x, y). This incurs a high computational cost that is infeasible in real-life scenarios. We also design an *offline* version of `Apollo`, where the adversary can only sample surrogate datasets that are *disjoint* from the target model's training set, i.e., $\forall i, D_i^s \cap D = \varnothing$. In the offline attack scenario, we can still optimize for the specificity condition **(b)**; and we substitute the sensitivity condition **(a)** with the observation that the adversarial samples will achieve better attack accuracy when they are near the *decision boundary* (denoted DB). The loss functions for the offline `Apollo` attack(s) would be:

$$\ell_{\mathbf{Un}}^{\mathbf{off}}(x'; x, y, \Theta) = \alpha \underbrace{\sum_i d(x', \mathrm{DB})}_{\textbf{(a)}} + \beta \underbrace{\sum_i \hat{\ell}(x'; \theta_i^s)}_{\textbf{(b)}}. \tag{17}$$

And:

$$\ell_{\mathbf{Ov}}^{\mathbf{off}}(x'; x, y, \Theta) = \alpha \underbrace{\sum_i d(x', \mathrm{DB})}_{\textbf{(a)}} + \beta \underbrace{\sum_i \ell(x'; \theta_i^s)}_{\textbf{(b)}}. \tag{18}$$

Adversarial Sample Generation. We optimize for x' with Stochastic Gradient Descent (SGD), as shown in Algorithm 1. Here, $\mathcal{B}_{t\cdot\varepsilon}(x) \setminus \mathcal{B}_{(t-1)\cdot\varepsilon}(x) = \{x' \mid (t-1)\varepsilon < d(x, x') \leq t\varepsilon\}$ is the space between two balls around x with radius $(t-1)\cdot\varepsilon$ and $t\cdot\varepsilon$. We project adversarial samples outside the ball at each step t to limit the magnitude of each update for the locality condition; the maximum perturbation would be $T \cdot \varepsilon$. This methodology allows us to adjust the attack precision and recall by selecting different values of T, ε, and τ, when lowering τ and increasing $T \cdot \varepsilon$ allow a high precision.

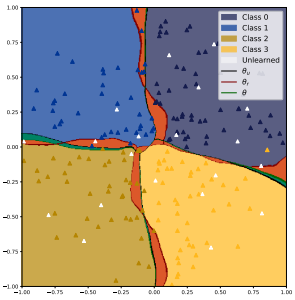
Algorithm 1: Adversarial Input Generation in `Apollo`

Input: Target model θ_u, target sample $(x, y) \in \mathcal{X} \times \mathcal{Y}$, m shadow models Θ^s, m *unlearned* shadow models Θ^{s_u}, search step size ε
Output: Adversarial input $x' \in \mathcal{X}$

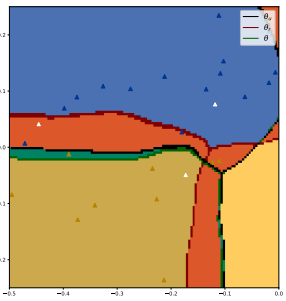
```
x' ← x;
for t ← 1, ··· , T do
    for i ← 1, ··· , m do
        g_{t,i} ← ∇_{x'} ℓ(x'; x, y, Θ)   ▷ Substitute for ℓ_Un, ℓ_Ov, ℓ^off_Un, or ℓ^off_Ov
    end
    x' ← SGD(x', 1/m Σ_{i=1}^m g_{t,i});
    if x' ∉ B_{t·ε}(x) \ B_{(t-1)·ε}(x) then
        x' ← Proj(x', B_{tε}(x))   ▷ Limit each update step for locality condition (c)
    end
    if 1/m Σ_{i=1}^m f_{θ_i^s}(x')_y < τ then
        return x'                   ▷ Early stopping
    end
end
return x';
```

E. A Two Dimensional Example

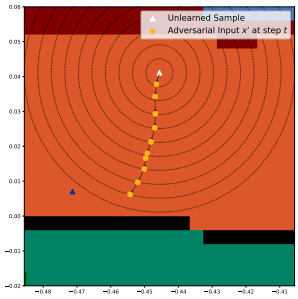
Inspired by Haim et al. [44], we illustrate our conjectures on a simple example of $(x, y) \in \mathbb{R}^2 \times \{0, 1, 2, 3\}$. We randomly



(a) UNDER-UNLEARNING and OVER-UNLEARNING on $\mathbb{R}^2$



(b) A local region of OVER-UNLEARNING on $\mathbb{R}^2$



(c) Adversarial Input for an unlearned sample under $\ell_{\mathbf{Un}}$

Fig. 3: We test Conj. 1 and 2 on a $(x, y) \in \mathbb{R}^2 \times \{0, 1, 2, 3\}$ example. In Fig. 3a, the four classes and the unlearned set are identified in the legend. Regions of UNDER-UNLEARNING and OVER-UNLEARNING are colored red and green, respectively. In Fig. 3c, the trajectory indicates the adversarial input x' at each iteration.

generate $N = 500$ samples within the range of $(-1, -1)$ and $(1, 1)$, each class of samples are assigned to a quadrant. We train a target model of 12 layers on 200 samples and $m = 16$ shadow models with $|D_i^s| = 200$, the architecture of which we show in Table VI (Appendix A). We randomly unlearn 10% of the training set with Gradient Ascent (**GA**) [45], the results are shown in Fig. 3.

As shown in Fig. 3a and 3b, MU inevitably leads to both UNDER-UNLEARNING (Conj. 1) and OVER-UNLEARNING (Conj. 2), indicated by the difference in the prediction labels between the unlearned model and a retrained model.

We further demonstrate in this example the effectiveness of our adversarial input generation with Algorithm 1. We adopt $\ell_{\mathbf{Un}}$ as the loss function, the search process for x' is shown in Fig. 3c. We demonstrate that Algorithm 1 is able to generate x' that satisfies conditions **(a)** through **(c)**.

IV. EVALUATION

A. Experimental Setup

Hardware. We train, unlearn, and subsequently perform the attack on all models on a NVIDIA A100 GPU with 40GB memory. The training process require approximately 20 minutes each for the target and shadow models. The time for unlearning process differ with different unlearning algorithms.

Datasets. Beyond the example on $\mathbb{R}^2$ discussed in Sec. III-E, we evaluate our attack on commonly employed datasets including CIFAR-10, CIFAR-100 and ImageNet [46], [47]. We train (and subsequently unlearn) the model on a training set of 20,000 images for both CIFAR-10 and CIFAR-100; we train the model on 512,466 images. We *do not* perform data augmentations on the images to ensure that the unlearned sample is identical to the trained sample [11]. Both CIFAR-10 and CIFAR-100 are integrated with the `PyTorch` library [48].

Models. We use ResNet18 [49], VGG-16 [50] and Swin Transformer [51], for the target and shadow models. For the target model and shadow models, we train them to have an accuracy no lower than 75% on the test set D_t. The target model is trained for 100 epochs with a learning rate of 1×10^{-4} with AdamW optimization algorithm [52] and a batch size of 64 on CIFAR-10. The shadow models are trained for 50 epochs for ResNet-18 [49] and VGG-16 [50] from scratch, and 10 epochs for pre-trained Swin Transformer [51]. We test our attack on different combinations of the target / surrogate models to determine the generalizability of our attack, which we discuss in Sec. IV-C

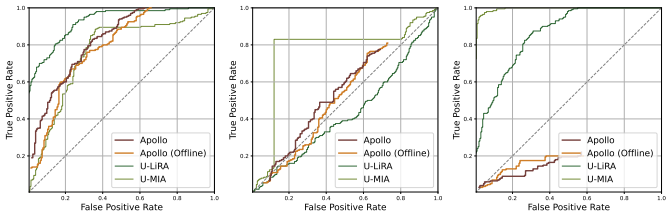
Unlearning. We adopt a setting similar to that in Huang et al. [53], and test our attack on 6 MU algorithms: Gradient Ascent (**GA**) [45], Fine-Tuning (**FT**) [18], Bad Teacher (**BT**) [54], **SCRUB** [10], **SalUn** [55], and **SFR-on** [53]. We also include Retraining-from-Scratch (**RT**) as a benchmark: by definition, neither Conj. 1 or 2 should be effective against **RT**. We randomly unlearn 10% of the training set.

The targeted MU algorithms are selected for their representativeness: two baseline unlearning algorithms, **GA** and **FT** concerns with only the unlearned set D_u or the retained set $D_r = D \setminus D_u$ respectively; **BT** uses *knowledge distillation* for unlearning; **SCRUB** unlearns through maximizing posterior *divergence* between the original model and the unlearned model; **SalUn** and **SFR-on** are state-of-the-art unlearning algorithms which share similar intuitions in accounting for the importance (saliency) of different model parameters.

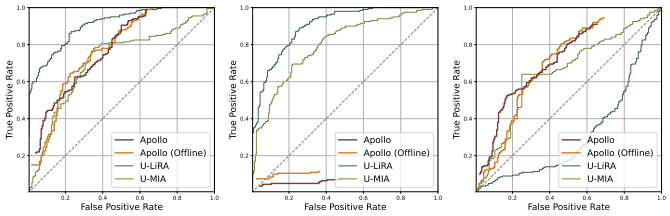
For a more detailed account of the MU algorithms, we refer the readers to their respective papers ([45], [18], [54], [10], [55], [53]), as well as the code implementations by Huang et al. [53].

Metrics. We simplify the three-way hypothesis test present in MU of $x \in D_u$, $x \in D_r$ and $x \in D_t$ into two categories by sampling 200 samples each from (a) the unlearned set D_u and (b) the test set D_t. We aim to show that our `Apollo` attack can reliably distinguish between unlearned samples and samples not involved in training. The main metric we report is the True Positive Rates (TPRs) at low False Positive Rates (FPRs) suggested by Carlini et al. in [29], as a high precision is more valuable to privacy attacks like MIAs. We also report metrics including precision and recall in our experiments.

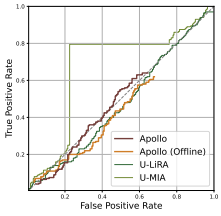
Attacks. We compare our proposed `Apollo` attack against two previous Membership Inference Attacks toward MU: A naive `U-MIA` employed by Kurmanji et al. [10] and the `U-LiRA`



(a) **GA** [45] (b) **FT** [18] (c) **BT** [54]

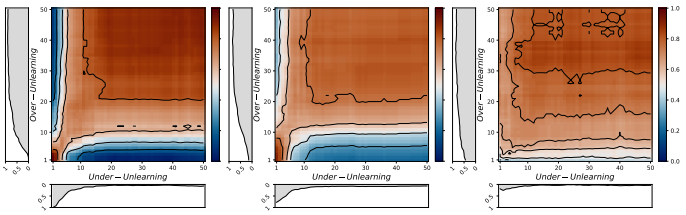


(d) **SCRUB** [10] (e) **SalUn** [55] (f) **SFR-on** [53]

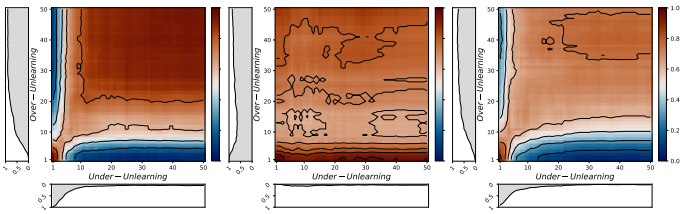


(g) **RT**

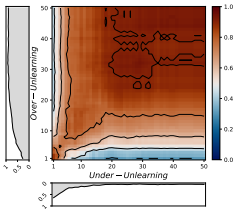
Fig. 4: Attack ROCs against various unlearning algorithms.



(a) **GA** [45] (b) **FT** [18] (c) **BT** [54]



(d) **SCRUB** [10] (e) **SalUn** [55] (f) **SFR-on** [53]



(g) **RT**

Fig. 5: Attack TPRs with UNDER-UNLEARNING (Conj. 1) and OVER-UNLEARNING (Conj. 2) at different step sizes against various unlearning algorithms.

attack proposed by Hayes et al. [11]. We do not include other attacks listed in Table I (namely [7], [8], [9]), as their assumption of adversarial access to the original model θ does not align with our threat model.

For `Apollo`, we adopt a step size $\varepsilon = 1.0$ for $T = 50$ rounds of adversarial search for x', in which we utilize $m = 32$ shadow models. The loss function (Eq. 15 through 18) are set to $\alpha = 1$ and $\beta = 4$. To unlearn each shadow model for the online attack, we unlearn $D_i^s \cap D_{target}$ for each $\theta_i^{s_u}$, in which D_{target} is the 400 samples whose membership status are to be determined.

For `U-LiRA` [11], we train and unlearn m shadow models same as `Apollo`. The `U-LiRA` attack tests for:

$$(x, y) \in D_u \iff \frac{\Pr\left(f_{\theta_u}(x)_y \mid \mathcal{N}(\mu_{s_u}, \sigma^2_{s_u})\right)}{\Pr\left(f_{\theta_u}(x)_y \mid \mathcal{N}(\mu_s, \sigma^2_s)\right)} > \tau. \quad (19)$$

In which $f_{\theta_i^s}(x)_y \sim \mathcal{N}(\mu_s, \sigma_s^2)$ and $f_{\theta_i^{s_u}}(x)_y \sim \mathcal{N}(\mu_{s_u}, \sigma_{s_u}^2)$ are two Gaussian distributions fitted over the prediction probabilities of x on class y with the shadow models Θ^s and unlearned shadow models Θ^{s_u}. τ adjusts the precision/recall of the attack. In practice, we use logit-transformed prediction probabilities for $f_\theta(x)_y$ [29], [11].

For the Naive `U-MIA` described by Kurmanji et al. in [10], we adopt a Support Vector Machine (SVM) as the binary classifier, which has a regularization parameter of 3 and with radial basis function (RBF) kernel, following the implementation of Fan et al. in [55].

B. Attack Results

Attack Efficacy. We show our attack efficacy against different unlearning algorithms compared to previous attacks. We do not adopt the metric of TPRs at a *fixed* FPR across different unlearning algorithms as our search for x' is confined within the ball $\mathcal{B}_{T\cdot\varepsilon}(x)$; therefore, we report TPRs at *lowest possible* FPRs achieved by `Apollo` and `Apollo` (offline) for each MU algorithm on CIFAR-10, CIFAR-100 [46] and ImageNet [47], shown in Table II, III and IV, respectively.

The results demonstrate that our proposed `Apollo` attack is able to achieve a high TPR at low FPRs, while its performance is limited by the *strict threat model*, it is nevertheless able to achieve a performance comparable to those of `U-MIA` and `U-LiRA`, even *superior* than the latter toward **FT** [18] and **SFR-on** [53]. We also note that `Apollo` is able to achieve stronger performance at lower FPRs on CIFAR-100 (Table III) and ImageNet (Table IV) compared to CIFAR-10 (Table II), as each class consists of fewer samples, and a model trained on these datasets is therefore more susceptible to changes in the decision space.

Beyond the attack TPRs at lowest possible FPRs, we also show the ROC curves of different attacks toward different MU algorithms, as shown in Fig. 4. Our proposed `Apollo` attack is able to achieve an overall comparable performance compared to `U-LiRA` and `U-MIA` for **GA**, **FT**, **SCRUB** and **SFR-on**; and in cases able to successfully infer membership status of

TABLE II: TPRs (%) at lowest possible FPRs (%), tested on ResNet-18 with CIFAR-10.

Attack Method	GA@6.0%	FT@9.5%	BT@2.5%	SCRUB@4.0%	SalUn@4.5%	SFR-on@2.5%	RT@4.5%
U-MIA [10]	16.5	11.5	95.0	9.0	15.5	3.0	5.5
U-LiRA [11]	68.5	6.5	28.0	6.0	20.0	2.5	4.0
Apollo	18.0	6.5	4.0	21.5	4.5	10.0	5.0
Apollo (Offline)	16.0	6.5	3.0	15.0	7.5	5.0	7.0

TABLE III: TPRs (%) at lowest possible FPRs (%), tested on ResNet-18 with CIFAR-100.

Attack Method	GA@0.5%	FT@1.0%	BT@13.5%	SCRUB@5.0%	SalUn@1.5%	SFR-on@1.5%	RT@1.0%
U-MIA [10]	7.5	0.5	48.5	17.0	8.5	2.0	1.0
U-LiRA [11]	14.5	1.0	25.0	12.5	17.0	2.0	1.5
Apollo	15.5	2.0	50.0	41.5	5.0	0.5	1.5
Apollo (Offline)	13.0	2.0	41.5	39.0	4.5	1.0	0.5

TABLE IV: TPRs (%) at lowest possible FPRs (%), tested on ResNet-18 with ImageNet.

Attack Method	GA@1.0%	FT@1.0%	BT@8.5%	SCRUB@4.5%	SalUn@1.0%	SFR-on@1.5%	RT@1.5%
U-MIA [10]	8.0	0.5	32.0	9.5	6.0	1.0	1.5
U-LiRA [11]	12.5	0.5	24.5	8.5	8.5	2.5	2.5
Apollo	17.5	1.0	12.5	14.0	6.5	2.5	2.0
Apollo (Offline)	13.5	0.5	8.0	14.5	2.0	2.0	1.0

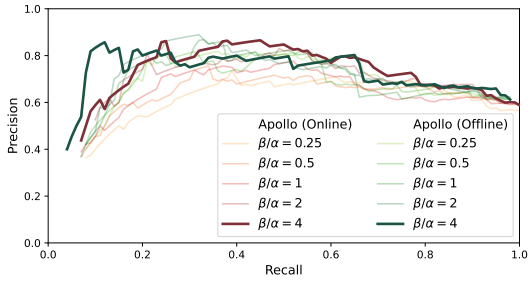
$x \in D_u$ toward **SFR-on**, where a likelihood-based U-LiRA attack failed (Fig. 4f).

Dynamics between UNDER-UNLEARNING and OVER-UNLEARNING. We next show how UNDER-UNLEARNING and OVER-UNLEARNING interacts in different MU algorithms, as well as how the bounds we derived in Theorem III.3 and III.4 perform. We plot the TPRs of Apollo under different search space $\mathcal{B}_{T,\varepsilon}(x)$ for Conj. 1 and 2. In our experiments, we fix $\varepsilon = 1.0$ and test for steps $t \leq 50$. The sub-figures along the X- and Y-axis represent the TPRs for either conjecture at each step, while the heatmap represents the TPRs where a sample is deemed positive if and only if *one* conjecture holds. The results are shown in Fig. 5.

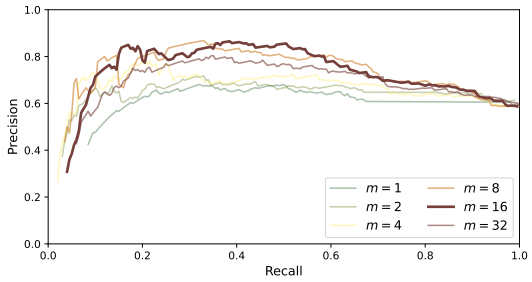
Recall our results on TPRs at low FPRs from Table II, here; when our proposed Apollo attack is successful, a clear pattern emerges from the True Positive distributions. For successful attacks like **GA** (Fig. 5a) and **SFR-on** (Fig. 5f), UNDER-UNLEARNING and OVER-UNLEARNING are bounded in local regions of $\|x - x'\|$, as evidenced by the low-TPR regions close to the axes, confirming the results in Theorem III.3 and III.4. For the unsuccessful attack against **BT** (Fig. 5c) and **SalUn** (Fig. 5e), OVER-UNLEARNING is near uniformly distributed, while UNDER-UNLEARNING is scarce, we postulate that the regulations of the unlearning algorithms violates the assumption of local Lipschitz condition (Lemma III.1). This indicates that different MU algorithms have distinctively different effects on the decision space of the model, which can be utilized as defense against Apollo.

C. Ablation Studies

Hyperparameters. As shown in Sec. III-D, the loss functions we used for generating the adversarial input x' (Eq. 15 through 18) follows the general form of $\ell = \alpha\ell_{\text{sensitivity}} + \beta\ell_{\text{specificity}}$. We show the attack precision-recall curves with different ratios of β/α in Fig. 6a. We demonstrate that a higher β/α ratio, which emphasizes the specificity condition, yields a better precision-recall tradeoff.



(a) Different ratios of β/α in the loss function.



(b) Different size of $|\Theta^s| = m$

Fig. 6: Precision-Recall trade-off against **GA** on CIFAR-10, under different hyperparameters used in Apollo.

Number of Shadow Models. We study the effectiveness of the number of shadow models, i.e., $|\Theta^s| = m$ on Apollo's

TABLE V: Apollo performance with different target/shadow model architectures.

Target Model	Shadow Models	TPRs (%) at lowest possible FPRs (%)
ResNet-18	ResNet-18	18.0@6.0%
	VGG-16	12.0@6.0%
	Swin-T	13.5@6.0%
VGG-16	ResNet-18	5.0@2.5%
	VGG-16	5.5@2.5%
	Swin-T	6.5@2.5%
Swin-T	ResNet-18	8.5@4.5%
	VGG-16	8.0@4.5%
	Swin-T	11.5@4.5%

performance, as shown in Fig. 6b. We observe that while increasing the number of shadow models used can yield a better precision-recall trade-off when m is low ($m \leq 16$), increasing m further results in performance degradation under $m = 32$. This is consistent with the observation in Wen et al. [36] that using too many shadow models will result in x' overfitting to specific shadow models and thus weakening attack utility.

Architecture of Shadow Models. Another key issue is the training process of the shadow models $\theta_i^s = \mathcal{A}_i^s(D_i^s)$, where we assume that the adversary is able to train shadow models with the same hyperparameters as those of the target model, i.e., $\mathcal{A} = \mathcal{A}^s$. We consider scenarios where the surrogate and target models are trained under different architectures, i.e., $\mathcal{A} \neq \mathcal{A}^s$, as shown in Table V. While employing shadow models with different architectures negatively affects the attack performance, it is still able to preserve a high precision.

V. Conclusion

Contributions. In this paper, we propose A Posteriori Label Only Membership Inference Attack towards Machine Unlearning, Apollo, that aims to infer the membership status of unlearned data samples from the unlearned model only. We demonstrate empirically that it is possible to extract membership information from a strict threat model against MU, posing a significant privacy threat to existing unlearning methods. The evaluations show that Apollo achieves inference accuracy comparable to those of the state-of-the-art attacks that require more adversarial access than Apollo. MU as a privacy-preserving technique, is still in its infancy; and future research is required to improve both the attacks and defenses for Machine Unlearning techniques.

Limitations. A limitation of our proposed Apollo attack is that the ability to adjust to different false positive rates is limited by the selection of hyperparameters (T, ε, and τ), which is easy for likelihood-based attacks [10], [11]. This limitation is inherent in the design of our attack and is the result of adopting a strict threat model. Another limitation of our methodology is the computational cost incurred by training of shadow models, which we address in part in Sec. IV-C. Future research is needed for developing more efficient label-only attacks to MU.

Acknowledgments

This work is funded by Cisco Research, under proposal id. 95254923.

References

[1] Y. Cao and J. Yang, "Towards making systems forget with machine unlearning," in *IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2015, pp. 463–480.

[2] L. Bourtoule, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, "Machine unlearning," in *2021 IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 141–159.

[3] "Regulation (EU) 2016/679 of the European parliament and of the council of 27 April 2016," 2016.

[4] "Title 1.81.5. california consumer privacy act of 2018 [1798.100 - 1798.199.100]," 2018.

[5] Z. Liu, H. Ye, C. Chen, Y. Zheng, and K. Lam, "Threats, attacks, and defenses in machine unlearning: A survey," *IEEE Open J. Comput. Soc.*, vol. 6, pp. 413–425, 2025.

[6] L. Tang and J. Joshi, "Towards privacy-preserving and secure machine unlearning: Taxonomy, challenges and research directions," in *2024 IEEE 6th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA)*, 2024, pp. 280–291.

[7] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, "When machine unlearning jeopardizes privacy," in *CCS*. ACM, 2021, pp. 896–911.

[8] J. Gao, S. Garg, M. Mahmoody, and P. N. Vasudevan, "Deletion inference, reconstruction, and compliance in machine (un)learning," *Proc. Priv. Enhancing Technol.*, vol. 2022, no. 3, pp. 415–436, 2022.

[9] Z. Lu, H. Liang, M. Zhao, Q. Lv, T. Liang, and Y. Wang, "Label-only membership inference attacks on machine unlearning without dependence of posteriors," *Int. J. Intell. Syst.*, vol. 37, no. 11, pp. 9424–9441, 2022.

[10] M. Kurmanji, P. Triantafillou, J. Hayes, and E. Triantafillou, "Towards unbounded machine unlearning," in *NeurIPS*, 2023.

[11] J. Hayes, I. Shumailov, E. Triantafillou, A. Khalifa, and N. Papernot, "Inexact unlearning needs more careful evaluations to avoid a false sense of privacy," *CoRR*, vol. abs/2403.01218, 2024.

[12] Y. Hu, J. Lou, J. Liu, F. Lin, Z. Qin, and K. Ren, "ERASER: machine unlearning in mlaas via an inference serving-aware approach," *CoRR*, vol. abs/2311.16136, 2023.

[13] J. Chen and D. Yang, "Unlearn what you want to forget: Efficient unlearning for llms," in *EMNLP*. Association for Computational Linguistics, 2023, pp. 12 041–12 052.

[14] R. Zhang, L. Lin, Y. Bai, and S. Mei, "Negative preference optimization: From catastrophic collapse to effective unlearning," in *First Conference on Language Modeling*, 2024.

[15] Y. Tan and B. Gong, "Lifting data-tracing machine unlearning to knowledge-tracing for foundation models," *CoRR*, vol. abs/2506.11253, 2025.

[16] P. Maini, Z. Feng, A. Schwarzschild, Z. C. Lipton, and J. Z. Kolter, "TOFU: A task of fictitious unlearning for llms," *CoRR*, vol. abs/2401.06121, 2024.

[17] S. Neel, A. Roth, and S. Sharifi-Malvajerdi, "Descent-to-delete: Gradient-based methods for machine unlearning," in *ALT*, ser. Proceedings of Machine Learning Research, vol. 132. PMLR, 2021, pp. 931–962.

[18] A. Warnecke, L. Pirch, C. Wressnegger, and K. Rieck, "Machine unlearning of features and labels," in *NDSS*. The Internet Society, 2023.

[19] C. Guo, T. Goldstein, A. Y. Hannun, and L. van der Maaten, "Certified data removal from machine learning models," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 3832–3842.

[20] H. Yan, X. Li, Z. Guo, H. Li, F. Li, and X. Lin, "ARCANE: an efficient architecture for exact machine unlearning," in *IJCAI*. ijcai.org, 2022, pp. 4006–4013.

[21] A. Thudi, H. Jia, I. Shumailov, and N. Papernot, "On the necessity of auditable algorithmic definitions for machine unlearning," in *USENIX Security Symposium*. USENIX Association, 2022, pp. 4007–4022.

[22] T. Shaik, X. Tao, H. Xie, L. Li, X. Zhu, and Q. Li, "Exploring the landscape of machine unlearning: A comprehensive survey and taxonomy," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 36, no. 7, pp. 11 676–11 696, 2025.

[23] A. Blanco-Justicia, N. Jebreel, B. Manzanares-Salor, D. Sánchez, J. Domingo-Ferrer, G. Collell, and K. E. Tan, "Digital forgetting in large language models: a survey of unlearning methods," *Artif. Intell. Rev.*, vol. 58, no. 3, p. 90, 2025.

[24] R. Chourasia and N. Shah, "Forget unlearning: Towards true data-deletion in machine learning," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 202. PMLR, 2023, pp. 6028–6073.

[25] E. Triantafillou, P. Kairouz, F. Pedregosa, J. Hayes, M. Kurmanji, K. Zhao, V. Dumoulin, J. C. S. J. Júnior, I. Mitliagkas, J. Wan, L. Sun-Hosoya, S. Escalera, G. K. Dziugaite, P. Triantafillou, and I. Guyon, "Are we making progress in unlearning? findings from the first neurips unlearning competition," *CoRR*, vol. abs/2406.09073, 2024.

[26] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography*, S. Halevi and T. Rabin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 265–284.

[27] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2017, pp. 3–18.

[28] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *CSF*. IEEE Computer Society, 2018, pp. 268–282.

[29] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramèr, "Membership inference attacks from first principles," in *SP*. IEEE, 2022, pp. 1897–1914.

[30] J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri, "Enhanced membership inference attacks against machine learning models," in *CCS*. ACM, 2022, pp. 3093–3106.

[31] Z. Wang, C. Zhang, Y. Chen, N. Baracaldo, S. Kadhe, and L. Yu, "Membership inference attacks as privacy tools: Reliability, disparity and ensemble," *CoRR*, vol. abs/2506.13972, 2025.

[32] C. A. Choquette-Choo, F. Tramèr, N. Carlini, and N. Papernot, "Label-only membership inference attacks," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 1964–1974.

[33] Y. Peng, J. Roh, S. Maji, and A. Houmansadr, "OSLO: one-shot label-only membership inference attacks," *CoRR*, vol. abs/2405.16978, 2024.

[34] Y. Wu, H. Qiu, S. Guo, J. Li, and T. Zhang, "You only query once: An efficient label-only membership inference attack," in *ICLR*. OpenReview.net, 2024.

[35] L. Graves, V. Nagisetty, and V. Ganesh, "Amnesiac machine learning," in *AAAI*. AAAI Press, 2021, pp. 11 516–11 524.

[36] Y. Wen, A. Bansal, H. Kazemi, E. Borgnia, M. Goldblum, J. Geiping, and T. Goldstein, "Canary in a coalmine: Better membership inference with ensembled adversarial queries," in *ICLR*. OpenReview.net, 2023.

[37] I. Shumailov, Z. Shumaylov, D. Kazhdan, Y. Zhao, N. Papernot, M. A. Erdogdu, and R. J. Anderson, "Manipulating SGD with data ordering attacks," in *NeurIPS*, 2021, pp. 18 021–18 032.

[38] T. Eisenhofer, D. Riepel, V. Chandrasekaran, E. Ghosh, O. Ohrimenko, and N. Papernot, "Verifiable and provably secure machine unlearning," in *SaTML*. IEEE, 2025, pp. 479–496.

[39] A. Golatkar, A. Achille, A. Ravichandran, M. Polito, and S. Soatto, "Mixed-privacy forgetting in deep networks," in *CVPR*. Computer Vision Foundation / IEEE, 2021, pp. 792–801.

[40] H. Owhadi, C. Scovel, and T. Sullivan, "Brittleness of bayesian inference under finite information in a continuous world," *Electronic Journal of Statistics*, vol. 9, no. 1, pp. 1–79, 2015.

[41] M. A. Iglesias, Y. Lu, and A. M. Stuart, "A bayesian level set method for geometric inverse problems," *Interfaces and free boundaries*, vol. 18, no. 2, pp. 181–217, 2016.

[42] M. Hein and M. Andriushchenko, "Formal guarantees on the robustness of a classifier against adversarial manipulation," in *NIPS*, 2017, pp. 2266–2276.

[43] Y. Tsuzuku, I. Sato, and M. Sugiyama, "Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks," in *NeurIPS*, 2018, pp. 6542–6551.

[44] N. Haim, G. Vardi, G. Yehudai, O. Shamir, and M. Irani, "Reconstructing training data from trained neural networks," in *NeurIPS*, 2022.

[45] A. Thudi, G. Deza, V. Chandrasekaran, and N. Papernot, "Unrolling SGD: understanding factors influencing machine unlearning," in *EuroS&P*. IEEE, 2022, pp. 303–319.

[46] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Technical Report, 2009.

[47] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*. IEEE Computer Society, 2009, pp. 248–255.

[48] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *NeurIPS*, 2019, pp. 8024–8035.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*. IEEE Computer Society, 2016, pp. 770–778.

[50] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[51] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *ICCV*. IEEE, 2021, pp. 9992–10 002.

[52] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*. OpenReview.net, 2019.

[53] Z. Huang, X. Cheng, J. Zheng, H. Wang, Z. He, T. Li, and X. Huang, "Unified gradient-based machine unlearning with remain geometry enhancement," in *NeurIPS*, 2024.

[54] V. S. Chundawat, A. K. Tarun, M. Mandal, and M. S. Kankanhalli, "Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher," in *AAAI*. AAAI Press, 2023, pp. 7210–7217.

[55] C. Fan, J. Liu, Y. Zhang, E. Wong, D. Wei, and S. Liu, "Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation," in *ICLR*. OpenReview.net, 2024.

APPENDIX A
ADDITIONAL IMPLEMENTATION DETAILS

Datasets. We sample our training and surrogate datasets as follows: for CIFAR-10 and CIFAR-100 [46], we split the training batch of 50,000 samples into 3 slices, each consisting of (a) 20,000, (b) 20,000 and (c) 10,000 samples, the training set D and test set D_t is then sampled from (a) and (c) respectively. For shadow sets D_i^s, in the online version of `Apollo` and `U-LiRA` [11], the shadow sets are sampled from slice (a) and (b); in the offline version of `Apollo`, the shadow sets are sampled from slice (b) only. For ImageNet [47], slice (a), (b) and (c) consists of 512,466, 512,466 and 256,235 samples.

Models. We show the architecture of the small NN used in Section III-E as follows:

TABLE VI: Architecture of the small NN used in Section III-E.

Layer	Input Channels	Output Channels	Design
1	2	256	Linear
2-11	256	256	Linear + ReLU + Batch Norm
12	256	4	Linear

Unlearning. We list the hyperparameters employed in the targeted unlearning algorithms in Sec. IV-A:

TABLE VII: Hyperparameters of the unlearning algorithms employed.

Method	Epochs	Learning Rate	Other Hyperparameters
RT	50	1×10^{-4}	constant scheduler, AdamW optimizer
GA	10	2×10^{-4}	cosine scheduler
FT	10	1×10^{-2}	cosine scheduler
BT	10	3×10^{-3}	temperature scalar = 1.0
SCRUB	10	3×10^{-4}	constant scheduler, temperature scalar = 4.0, $\alpha = 1 \times 10^{-3}$
SalUn	10	3×10^{-4}	constant scheduler, threshold = top-20%
SFR-on	\	\	cosine scheduler, $T_{out} = 1500$, $\alpha = 1.0$, $T_{in} = 5$, $\beta^f = 0.25$, $\beta^r = 0.01$, $\lambda = 0.5$, $\gamma = 1$