A Universal Source-Free Class Unlearning Framework via Synthetic Embeddings

Zahra Dehghani *zahra.dehghani-tafti.1@ens.etsmtl.ca*
LIVIA, ILLS, Mila - Quebec AI Institute, ÉTS Montreal, Canada

Pablo Piantanida *pablo.piantanida@cnrs.fr*
ILLS, Mila - Quebec AI Institute,
CNRS, CentraleSupélec - Université Paris-Saclay, France

Mohammadhadi Shateri *mohammadhadi.shateri@etsmtl.ca*
LIVIA, ILLS, ÉTS Montreal, Canada

Reviewed on OpenReview: *https://openreview.net/forum?id=Fb2sZ1eoVe*

Abstract

Class unlearning in neural classifiers refers to selectively removing the model's ability to recognize a target (forget) class by reshaping the decision boundaries. This is essential when taxonomies change, labels are corrected, or legal or ethical requirements mandate class removal. The objective is to preserve performance on the remaining (retain) classes while avoiding costly full retraining. Existing methods generally require access to the source, i.e., forget/retain data or a relevant surrogate dataset. This dependency limits their applicability in scenarios where access to source data is restricted or unavailable. Even the recent source-free class unlearning methods rely on generating samples in the data space, which is computationally expensive and not even essential for doing class unlearning. In this work, we propose a novel source-free class unlearning framework that enables existing unlearning methods to operate using only the deployed model. We show that, under assumptions on the forget loss with respect to logits, class unlearning can be performed source-free for any given neural classifier by utilizing randomly generated samples within the classifier's intermediate space. Specifically, randomly generated embeddings pseudo-labeled by the model as belonging to the forget or retain classes can support effective source-free unlearning. Our analysis further shows that, under conditions on the forget loss and synthetic forget embeddings, minimizing the forget loss induces expected logit shifts consistent with class unlearning, without requiring a specific parametric form of the embedding distribution. We validate our framework on four backbone architectures, ResNet-18, ResNet-50, ViT-B/16, and Swin-T, across three benchmark datasets, CIFAR-10, CIFAR-100, and TinyImageNet. Our experimental results show that existing class unlearning methods can operate within our source-free framework, with minimal impact on their forgetting efficacy and retain class accuracy. The code is available at `https://github.com/Yasaman-dt/Source_Free_Class_Unlearning`.

1 Introduction

Deep learning models have achieved remarkable performance across domains, but their tendency to memorize training data makes them susceptible to privacy attacks such as membership inference attacks (Salem et al., 2018; Shokri et al., 2017; Song et al., 2019; Yeom et al., 2018) and model inversion attacks (Chen et al., 2021; Fredrikson et al., 2015). These risks pose serious concerns in privacy-sensitive applications, particularly under regulations such as General Data Protection Regulation (GDPR) (Voigt & Von dem Bussche, 2017) and

California Consumer Privacy Act (CCPA) (Goldman, 2020) that mandates a "right to be forgotten", requiring effective removal of specific data from trained models. In response, machine unlearning has emerged as a promising direction to remove the influence of specific instances or classes without retraining from scratch. Unlearning methods fall into model-intrinsic (Lin et al., 2023), data-driven (Bourtoule et al., 2021; Hayase et al., 2020), and model-agnostic categories (Kurmanji et al., 2023; Chen et al., 2023; Cotogni et al., 2023; Cha et al., 2024), with a key distinction between exact unlearning (Bourtoule et al., 2021; Yan et al., 2022) and approximate unlearning. Although recent approximate unlearning methods reduce retraining overhead, most still require access to the forget set, the retain set, or a surrogate dataset that approximates the training distribution.

We consider class unlearning, a practical scenario in which models must forget selected classes (Tarun et al., 2023; Kodge et al., 2024; Zhou et al., 2025; Zhang et al., 2025; Wang et al., 2025), motivated by applications such as face recognition, backdoor defense, data poisoning, and semantic segmentation (Chen et al., 2023; Liu et al., 2022; Zhou et al., 2025). This work challenges the widely held assumption that access to original training data is required for class unlearning. Existing source-free class-unlearning methods still reconstruct input-level samples by training generative models, under the assumption that realistic or adversarial surrogates are needed to approximate the decision boundaries. This design makes the unlearning pipeline computationally heavy, tightly coupled to specific generator architectures, and in some cases dependent on additional surrogate models or datasets. We propose a novel framework for source-free class unlearning that operates entirely without access to original or surrogate forget and retain datasets. Our approach leverages randomly generated embeddings in the intermediate space of the target classifier. More precisely, we generate class-conditional synthetic embeddings by randomly sampling in the model's intermediate embedding space and pseudo-labeling them based on the model's predictions. These synthetic embeddings serve as proxies, allowing existing state-of-the-art unlearning methods to be adapted seamlessly to a fully source-free setting. We theoretically show that, under conditions on the forget loss and synthetic forget embeddings, these embeddings induce expected logit shifts consistent with the decision-boundary adjustments required for class unlearning. Empirically, this mechanism preserves strong accuracy on the retain classes across a wide range of architectures and unlearning methods.

This work enables class-level unlearning in a fully source-free setting, which is compatible with a wide range of existing unlearning methods. Our framework successfully adapts several state-of-the-art techniques, including Finetuning (Golatkar et al., 2020), Negative Gradient (Golatkar et al., 2020), Negative Gradient+ (Kurmanji et al., 2023), Random Labels (Hayase et al., 2020), Boundary Expanding (Chen et al., 2023), Boundary Shrink (Chen et al., 2023), DELETE (Zhou et al., 2025), SCRUB (Kurmanji et al., 2023), and SCAR (Bonato et al., 2024), to operate effectively without requiring access to any original training data or relevant surrogate. Our main contributions are summarized as follows:

- We propose a novel *source-free* class unlearning framework that operates solely on a target model and the label of the class to be forgotten, without requiring any access to original, surrogate, or validation dataset. Our method generates synthetic class-conditional embeddings by sampling random vectors within the model's intermediate feature space and pseudo-labeling them using the model itself, enabling the adaptation of existing unlearning methods to a fully source-free regime.

- We show that these synthetic embeddings, without assuming a specific parametric form for their marginal distribution, support source-free class unlearning through expected logit steering under conditions on the forget loss and synthetic forget embeddings. Empirically, multiple state-of-the-art unlearning techniques operate within our framework and achieve performance close to data-access settings.

- We empirically validate our framework on ResNet-18, ResNet-50, ViT-B/16, and Swin-T backbones using CIFAR-10, CIFAR-100, and TinyImageNet datasets. The results show that a wide range of existing unlearning methods can function within our source-free setting with minimal degradation in the unlearning performance.

2 Related Works

Class unlearning aims to remove the influence of a target class from a trained model while preserving performance on the remaining classes. Class unlearning methods differ mainly by data access during unlearning: availability of retain data, forget data, both, or neither.

Methods requiring both retain and forget sets. Many effective class unlearning methods assume access to both forget and retain datasets. Distillation-based approaches such as SCalable Remembering and Unlearning unBound (SCRUB) (Kurmanji et al., 2023) guide student models via knowledge transfer and pruning. Machine Unlearning with Dimensional Alignment (MUDA) (Seo et al., 2025) introduces dimensional alignment loss and a self-distillation scheme that explicitly leverages both forget and retain sets to erase the influence of forget samples while preserving retain knowledge. The recently proposed SVD-based method (Kodge et al., 2024) performs gradient-free, single-step class unlearning by estimating retain and forget spaces from small subsets of both datasets and suppressing class-discriminatory activations.

Retain-free methods. These approaches remove dependence on retain data and operate mainly on forget samples. Negative Gradient reverses the estimated contribution of forget samples to the weights (Golatkar et al., 2020). Boundary Shrink and Boundary Expanding techniques (Chen et al., 2023) adjust decision boundaries by contracting or expanding regions related to forget samples. Partially Blinded Unlearning (PBU) (Panda et al., 2025) perturbs model parameters using a Bayesian loss. Other lines estimate the retain Hessian from forget data and model parameters (Ahmed et al., 2025), or inject targeted label noise to induce misclassification with minimal updates (Ye et al., 2025). Just in Time unlearning (JiT) enforces local Lipschitz regularization on forget samples and their perturbations (Foster et al., 2024), while zero-shot proxy generation synthesizes adversarial retain surrogates followed by subspace projection and pseudo-labeling (Chen et al., 2025). From an input-sensitivity view, Machine Unlearning by Minimizing input sensitivity (MU-Mis) minimizes the sensitivity gap between target-class and irrelevant-class logits to withdraw forget influence with limited utility loss (Cheng et al., 2024). Zhou et al. (2025) proposes DELETE, a decoupled distillation method that suppresses the forget-class logits with a masking function and distills dark knowledge from the frozen model to preserve remaining classes. Recently, Selective-distillation for Class and Architecture-agnostic unleaRning (SCAR) (Bonato et al., 2024) introduced a retain-free method that leverages Mahalanobis-guided metric learning and a distillation strategy using a surrogate out-of-distribution dataset to preserve model performance. In addition, it proposes a source-free class unlearning variant that requires no access to either retain or forget data, while still relying on the surrogate dataset.

Forget-free methods. Some methods operate using retain data and without direct access to forget samples. Fine-tuning approaches update models exclusively on retain data to indirectly remove forget sample influence. Recent work, such as RELOAD (Newatia et al., 2024), introduces blind unlearning, which performs approximate unlearning without access to the forget set. Instead, it leverages cached gradients from the original training and selectively re-initializes parameters most influenced by the forget data, guided by differences between full and retain gradients. Similarly, Unlearning With Single Pass Impair and Repair (UNSIR) (Tarun et al., 2023) operates in a zero-glance setting, where forget samples are entirely inaccessible. More precisely, it employs a single-pass impair-repair strategy using error-maximizing noise and a small retain subset to forget class-level information.

Source-free methods. In the source-free unlearning setting, neither forget nor retain data is available. Chundawat et al. (2023) proposes Min–Max noise, which adversarially perturbs weights to raise loss on forget classes while preserving retain accuracy, and Gated Knowledge Transfer (GKT), which distills a student from a teacher while filtering synthetic samples linked to the forget classes. GKT, however, can over-filter (discarding samples that still encode retain information) and exhibits generator imbalance (overproducing forget-class samples), reducing data efficiency. To address these issues, Zhang et al. (2025) introduces the Inhibited Synthesis PostFilter (ISPF) framework, combining Inhibited Synthesis to discourage the generation of forget-class data with a PostFilter to suppress forget-class logits without discarding samples. However, both approaches initialize and train a new model from scratch as part of the distillation process, which incurs substantial computational overhead. Wang et al. (2025) proposes Data Synthesis–based Discrimination-Aware (DSDA), which synthesizes data via Accelerated Energy-Guided Langevin Sampling and performs unlearning through Discrimination-Aware Multitask Optimization. Despite efficiency gains, DSDA still in-

curs nontrivial computational overhead due to the recursive sampling needed to construct synthetic forget and retain datasets. We demonstrate that synthesizing input-level data is not necessary for effective class unlearning, and intermediate random embeddings are sufficient to reshape the decision boundaries. Building on this insight, our proposed framework operates entirely in the intermediate embedding space by sampling synthetic embeddings and pseudo-labeling them using the model itself. This significantly reduces computational overhead while maintaining unlearning effectiveness. Compared to recent source-free methods such as DSDA, ISPF, and GKT, this approach avoids data generators, input reconstruction, and student-teacher training, making it significantly more efficient.

3 Methodology

In this section, we introduce our notations, formalize the problem setting, and lay down the theoretical foundation necessary for source-free class unlearning. Subsequently, we propose our source-free unlearning methodology grounded on this theoretical insight. We also provide a finite-sample concentration result in Appendix B, showing that empirical logit updates concentrate around their population-level steering direction as the number of synthetic forget embeddings increases.

3.1 Notations and Problem Setup

Consider a pre-trained classifier model defined as $\Phi = h \circ g \circ e$. The feature extractor $e : \mathcal{X} \to \mathbb{R}^d$, parameterized by θ_e, maps input samples $\mathbf{x} \in \mathcal{X}$ to a d-dimensional embedding $\mathbf{z} = e(\mathbf{x}) \in \mathbb{R}^d$. An intermediate transformation $g : \mathbb{R}^d \to \mathbb{R}^l$, parameterized by θ_g, maps $\mathbf{z}$ to an l-dimensional latent embedding $g(\mathbf{z}) \in \mathbb{R}^l$. Finally, the classifier head $h : \mathbb{R}^l \to \mathbb{R}^C$, with parameters θ_h, computes class logits $h(g(\mathbf{z})) \in \mathbb{R}^C$. We denote the space of class labels as $\mathcal{Y} = \mathcal{Y}_f \cup \mathcal{Y}_r$, where $\mathcal{Y}_f$ is the set of classes targeted for unlearning (forget classes), and $\mathcal{Y}_r$ is the set of retain classes with $\mathcal{Y}_f \cap \mathcal{Y}_r = \emptyset$. For clarity, we first present the formulation for the single-class unlearning case, where the forget set contains one class denoted as c_f, and thus $\mathcal{Y}_f = \{c_f\}$ and $\mathcal{Y}_r = \mathcal{Y} \setminus \{c_f\}$. However, the proposed framework is not restricted to this setting and can be naturally extended to multi-class unlearning with $|\mathcal{Y}_f| > 1$, which we also evaluate experimentally. Under this notation, *class unlearning* is defined as the process of selectively removing the model's ability to recognize the target class c_f or more generally the target class set $\mathcal{Y}_f$ by reshaping the decision boundary, while preserving predictive performance on the remaining classes $\mathcal{Y}_r$.

3.2 Proposed Methodology

We assume availability of embeddings drawn from an arbitrary intermediate embedding space, such as the output of the feature extractor e. Formally, we denote embeddings in this space as random variables $\mathbf{z} \in \mathbb{R}^d$, sampled from an arbitrary distribution $p_{\mathbf{z}}(\mathbf{z})$. These embeddings do not necessarily follow any particular distribution from the original training data. More precisely, given a classifier model $\Phi = h \circ g \circ e$, we obtain pseudo-labels for each embedding $\mathbf{z}_i$ by applying the intermediate transformation and the classifier head:

$$\hat{y}_i = \arg\max_{k \in \mathcal{Y}} [h(g(\mathbf{z}_i))]_k. \tag{1}$$

Using these pseudo-labels, we construct two embedding subsets: the forget set $\mathcal{E}_f$ and the retain set $\mathcal{E}_r$, defined as

$$\mathcal{E}_f = \{\mathbf{z}_i \in \mathbb{R}^d : \hat{y}_i = c_f\}, \tag{2}$$

$$\mathcal{E}_r = \{\mathbf{z}_i \in \mathbb{R}^d : \hat{y}_i \in \mathcal{Y}_r\}, \tag{3}$$

with $N_f := |\mathcal{E}_f|$ and $N_r := |\mathcal{E}_r|$. In practice, the sampling distribution is only required to induce a non-empty pseudo-forget set (and sufficiently many pseudo-forget embeddings) through the model's pseudo-labeling mechanism. In class unlearning methods, the overall objective is often formulated as a combination of two components: a forget loss $\mathcal{L}_f$ computed on the forget set $\mathcal{E}_f$, and a retain loss $\mathcal{L}_r$ computed on the retain set $\mathcal{E}_r$. The total unlearning loss is typically expressed as $\mathcal{L}_u = \mathcal{L}_f + \lambda \mathcal{L}_r$, where λ controls the trade-off between forgetting and utility preservation. The forget loss $\mathcal{L}_f$ encourages the model to remove knowledge

related to the forget class by reshaping the decision boundary, while the retain loss $\mathcal{L}_r$ is used to preserve performance on the retain classes and prevent catastrophic forgetting. In the following proposition, we show that, under explicit assumptions on the forget loss and the synthetic forget embeddings, gradient-based optimization induces expected logit shifts consistent with class unlearning, independently of the explicit form of the sampling distribution $p_{\mathbf{z}}(\mathbf{z})$.

Assumptions (Expectation-Based Version). We state the assumptions used in the following proposition.

1. **Linear classifier head and head-only update:** We assume the classifier head is linear, i.e.,
$$[h(g(\mathbf{z}))]_k = (\theta_h)_k^\top g(\mathbf{z}), \tag{4}$$
where $(\theta_h)_k \in \mathbb{R}^l$ is the k-th row of $\theta_h \in \mathbb{R}^{C\times l}$. The intermediate map g is fixed during the analysis below, and only θ_h is updated.
2. **Differentiability:** The forget loss $\mathcal{L}_f$ is differentiable with respect to the logits.
3. **Average logit-wise monotonicity on the synthetic forget distribution:** We assume that the forget loss induces a positive forget-class logit gradient on average over the synthetic forget distribution, i.e.,
$$\mathbb{E}_{\mathbf{z}\sim p_f}\left[\frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}))]_{c_f}}\right] > 0. \tag{5}$$
4. **Non-empty conditional synthetic forget distribution:** Let
$$p_f(\mathbf{z}) := p_{\mathbf{z}}(\mathbf{z} \mid \hat{y} = c_f), \qquad \hat{y} = \arg\max_{k\in\mathcal{Y}} [h(g(\mathbf{z}))]_k. \tag{6}$$
We assume p_f is well-defined (equivalently, $\Pr(\hat{y} = c_f) > 0$).
5. **Gradient-weighted alignment on the synthetic forget distribution:** For each class $k \in \mathcal{Y}$, define
$$s_k(\mathbf{z}) := \frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}))]_k}.$$
We assume
$$\mathbb{E}_{\mathbf{z},\mathbf{z}'\sim p_f}\left[s_{c_f}(\mathbf{z})\, g(\mathbf{z})^\top g(\mathbf{z}')\right] > 0, \tag{7}$$
and for every $k \in \mathcal{Y}_r$,
$$\mathbb{E}_{\mathbf{z},\mathbf{z}'\sim p_f}\left[s_k(\mathbf{z})\, g(\mathbf{z})^\top g(\mathbf{z}')\right] < 0. \tag{8}$$

Assumption 3 captures the intended sign structure of the forget loss, while Assumption 5 enforces the corresponding weighted feature-alignment condition used in the proof. Since Assumptions 1, 2, and 4 hold in our setting, we focus on empirically assessing Assumptions 3 and 5 for unlearning methods that include a forgetting loss, and show in Appendix H that these assumptions are satisfied for the methods considered in this work.

Proposition 1 (Expected Source-Free Logit Steering on Synthetic Forget Embeddings)**.** *Consider a trained classifier model $\Phi = h \circ g \circ e$. Let $\mathcal{E}_f$ be a non-empty synthetic forget set formed by pseudo-labeling embeddings sampled from an arbitrary distribution $p_{\mathbf{z}}(\mathbf{z})$, and assume head-only gradient descent on $\mathcal{L}_f$. Under Assumptions in equation 5–equation 8, one gradient descent step decreases the forget-class logit and increases the retain-class logits* in expectation *over synthetic forget embeddings. In particular, for a random probe embedding $\mathbf{z}_l \sim p_f$,*
$$\mathbb{E}\Big[[h(g(\mathbf{z}_l))]_{c_f}^{(j+1)} - [h(g(\mathbf{z}_l))]_{c_f}^{(j)}\Big] < 0,$$
and for every $k \in \mathcal{Y}_r$,
$$\mathbb{E}\Big[[h(g(\mathbf{z}_l))]_{k}^{(j+1)} - [h(g(\mathbf{z}_l))]_{k}^{(j)}\Big] > 0.$$
This provides the mechanism by which repeated updates can contract the decision region of class c_f on synthetic forget embeddings, independently of the specific parametric form of $p_{\mathbf{z}}(\mathbf{z})$ (as long as p_f is well-defined and non-empty).

Proof. Under the head-only update regime, a gradient descent step with learning rate $\alpha > 0$ is

$$\theta_h^{(j+1)} = \theta_h^{(j)} - \alpha \frac{\partial \mathcal{L}_f}{\partial \theta_h^{(j)}}. \tag{9}$$

For class $k \in \mathcal{Y}$, the gradient with respect to the k-th row of θ_h is

$$\frac{\partial \mathcal{L}_f}{\partial (\theta_h)_k^{(j)}} = \frac{1}{N_f} \sum_{\mathbf{z}_i \in \mathcal{E}_f} \frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}_i))]_k} \, g(\mathbf{z}_i). \tag{10}$$

Fix a probe embedding $\mathbf{z}_l \in \mathcal{E}_f$. Its class-$k$ logit after one update is

$$\begin{aligned}
[h(g(\mathbf{z}_l))]_k^{(j+1)} &= (\theta_h)_k^{(j+1)\top} g(\mathbf{z}_l) \\
&= (\theta_h)_k^{(j)\top} g(\mathbf{z}_l) - \alpha \left(\frac{\partial \mathcal{L}_f}{\partial (\theta_h)_k^{(j)}} \right)^\top g(\mathbf{z}_l) \\
&= [h(g(\mathbf{z}_l))]_k^{(j)} - \frac{\alpha}{N_f} \sum_{\mathbf{z}_i \in \mathcal{E}_f} \frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}_i))]_k} \, g(\mathbf{z}_i)^\top g(\mathbf{z}_l).
\end{aligned} \tag{11}$$

Now take expectation over the synthetic forget sampling process. In the population view, let $\mathbf{z}, \mathbf{z}' \overset{\text{i.i.d.}}{\sim} p_f$. In the population view (or equivalently, replacing the empirical average by its expectation under p_f), the expected one-step logit shift is

$$\mathbb{E}\Big[[h(g(\mathbf{z}'))]_k^{(j+1)} - [h(g(\mathbf{z}'))]_k^{(j)} \Big] = -\alpha \, \mathbb{E}_{\mathbf{z}, \mathbf{z}' \sim p_f} \left[\frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}))]_k} \, g(\mathbf{z})^\top g(\mathbf{z}') \right]. \tag{12}$$

For $k = c_f$, Assumption equation 7 implies

$$\mathbb{E}_{\mathbf{z}, \mathbf{z}' \sim p_f} \left[\frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}))]_{c_f}} \, g(\mathbf{z})^\top g(\mathbf{z}') \right] > 0.$$

Substituting into equation 12 yields

$$\mathbb{E}\Big[[h(g(\mathbf{z}'))]_{c_f}^{(j+1)} - [h(g(\mathbf{z}'))]_{c_f}^{(j)} \Big] < 0.$$

Hence, the forget-class logit decreases in expectation.

For any retain class $k \in \mathcal{Y}_r$, Assumption equation 8 implies

$$\mathbb{E}_{\mathbf{z}, \mathbf{z}' \sim p_f} \left[\frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}))]_k} \, g(\mathbf{z})^\top g(\mathbf{z}') \right] < 0.$$

Using equation 12, we obtain

$$\mathbb{E}\Big[[h(g(\mathbf{z}'))]_k^{(j+1)} - [h(g(\mathbf{z}'))]_k^{(j)} \Big] > 0, \qquad \forall k \in \mathcal{Y}_r.$$

Therefore, retain-class logits increase in expectation on synthetic forget embeddings.

The result shows that one gradient step on $\mathcal{L}_f$ induces an expected logit shift that suppresses class c_f and promotes retain classes on pseudo-forget embeddings. Repeated updates consequently contract the decision region of c_f on the synthetic forget distribution. The argument depends only on the conditional synthetic forget distribution p_f induced by pseudo-labeling, and not on the explicit parametric form of the original sampling distribution $p_{\mathbf{z}}(\mathbf{z})$, except that p_f must be non-empty. □

Remark. Proposition 1 is distribution-agnostic in the sense that it does not require any specific parametric form for $p_{\mathbf{z}}(\mathbf{z})$. The only requirement is that pseudo-labeling induces a non-empty conditional forget distribution $p_f(\mathbf{z}) = p_{\mathbf{z}}(\mathbf{z} \mid \hat{y} = c_f)$ and that the gradient-weighted alignment conditions in Assumptions equation 7–equation 8 hold.

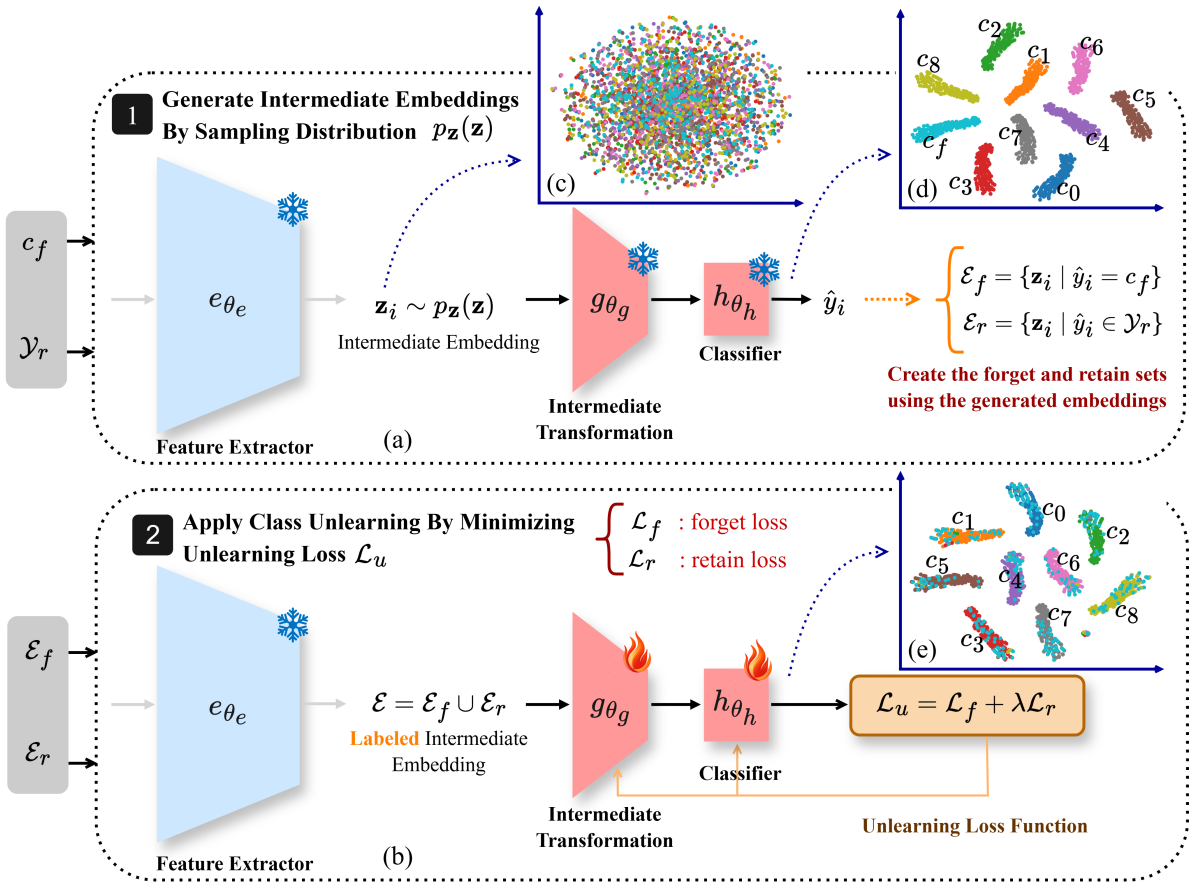

Figure 1: Illustration of the proposed source-free class unlearning framework. (a) **Step 1:** synthetic embeddings are sampled randomly from an arbitrary distribution in the intermediate embedding space and pseudo-labeled by the model to form the synthetic forget set $\mathcal{E}_f$ and retain set $\mathcal{E}_r$. (b) **Step 2:** the subsequent layers of the model are updated using these embeddings by minimizing the forget loss $\mathcal{L}_f$ to forget the target class set $\mathcal{Y}_f = \{c_f\}$, while optionally preserving performance on retain classes $\mathcal{Y}_r$ through the retain loss $\mathcal{L}_r$. (c) t-SNE of intermediate embeddings. (d) t-SNE of softmax probability before unlearning. (e) t-SNE of softmax probability after unlearning.

Discussion of assumptions. The gradient-weighted alignment assumptions are reasonable in practice because they only require a consistent *average* interaction between pseudo-forget embeddings and the forget-loss gradients, rather than pointwise geometric constraints on every sampled embedding pair. Empirically, pseudo-forget embeddings produced by model-based pseudo-labeling tend to occupy coherent regions in the intermediate representation space, which makes the corresponding inner products sufficiently structured on average. In addition, many class-unlearning methods induce sign-consistent logit gradients on forget embeddings (e.g., they suppress the forget-class logit while promoting retain-class logits), which aligns naturally with our monotonicity assumption. The boundedness condition used in the finite-sample concentration result is also standard in neural-network analyses and is satisfied whenever intermediate embeddings and logit gradients are bounded, for example by feature normalization, bounded activations, gradient clipping, or restricted optimization steps. Together, these assumptions provide a practical and interpretable bridge between the population-level source-free steering mechanism and its empirical realization.

With these assumptions in place, the expected logit-steering result in Proposition 1 can be strengthened into a margin-based characterization that directly captures the contraction of the forget-class decision region on synthetic forget embeddings.

Proposition 2 (Expected Margin Decrease on Synthetic Forget Embeddings)**.** *Under the assumptions of Proposition 1, define the forget-vs-retain margin on a synthetic forget embedding* $\mathbf{z} \sim p_f$ *as*

$$m(\mathbf{z}) := [h(g(\mathbf{z}))]_{c_f} - \frac{1}{|\mathcal{Y}_r|} \sum_{k \in \mathcal{Y}_r} [h(g(\mathbf{z}))]_k. \tag{13}$$

Then one gradient descent step on $\mathcal{L}_f$ *decreases this margin in expectation:*

$$\mathbb{E}_{\mathbf{z} \sim p_f} \Big[m^{(j+1)}(\mathbf{z}) - m^{(j)}(\mathbf{z}) \Big] < 0. \tag{14}$$

Consequently, minimizing $\mathcal{L}_f$ *reduces, in expectation, the relative dominance of the forget class over the retain classes on pseudo-forget embeddings.*

Proof sketch. By definition of $m(\mathbf{z})$,

$$\begin{aligned} m^{(j+1)}(\mathbf{z}) - m^{(j)}(\mathbf{z}) = &\Big([h(g(\mathbf{z}))]_{c_f}^{(j+1)} - [h(g(\mathbf{z}))]_{c_f}^{(j)} \Big) \\ &- \frac{1}{|\mathcal{Y}_r|} \sum_{k \in \mathcal{Y}_r} \Big([h(g(\mathbf{z}))]_k^{(j+1)} - [h(g(\mathbf{z}))]_k^{(j)} \Big). \end{aligned} \tag{15}$$

Taking expectation over $\mathbf{z} \sim p_f$ and applying Proposition 1, the first term is strictly negative in expectation, while each retain-class term inside the summation is strictly positive in expectation. Therefore, the average retain-class increment is positive, and subtracting it makes the second term strictly negative. Hence,

$$\mathbb{E}_{\mathbf{z} \sim p_f} \Big[m^{(j+1)}(\mathbf{z}) - m^{(j)}(\mathbf{z}) \Big] < 0.$$

This proves that one update step decreases the forget-vs-retain margin in expectation on synthetic forget embeddings. □

Building on Proposition 1, we propose a practical and fully source-free class unlearning framework. The central idea is to sample synthetic embeddings from an arbitrary distribution $p_{\mathbf{z}}(\mathbf{z})$ in the intermediate embedding space and pseudo-label them using the classifier head to form synthetic forget and retain sets. These synthetic sets act as surrogates for unavailable source data and enable gradient-based optimization of the unlearning objective. In particular, Proposition 1 shows that, under conditions on the forget loss and the synthetic forget distribution, minimizing the forget loss $\mathcal{L}_f$ induces expected logit shifts consistent with class unlearning. Figure 1 illustrates the proposed source-free unlearning pipeline, and Algorithm 1 summarizes the procedure.

Algorithm 1 Source-Free Class Unlearning Framework

Require: Pre-trained classifier model $\Phi = h \circ g \circ e$, target class to forget c_f, number of synthetic embeddings N, embedding distribution $p_{\mathbf{z}}(\mathbf{z})$, forget loss function $\mathcal{L}_f$, retain loss function $\mathcal{L}_r$, unlearning loss function $\mathcal{L}_u$, learning rate α

Initialize: synthetic forget set $\mathcal{E}_f = \emptyset$ and retain set $\mathcal{E}_r = \emptyset$
for $i = 1$ to N **do**
  Sample embedding $\mathbf{z}_i \sim p_{\mathbf{z}}(\mathbf{z})$
  Obtain pseudo-label: $\hat{y}_i = \arg\max_{k \in \mathcal{Y}} [h(g(\mathbf{z}_i))]_k$
  if $\hat{y}_i = c_f$ **then**
    $\mathcal{E}_f \leftarrow \mathcal{E}_f \cup \{\mathbf{z}_i\}$
  else
    $\mathcal{E}_r \leftarrow \mathcal{E}_r \cup \{\mathbf{z}_i\}$
  end if
end for
for each gradient update step **do**
  Compute loss $\mathcal{L}_u = \mathcal{L}_f + \lambda \mathcal{L}_r$: compute $\mathcal{L}_f$ using $\mathcal{E}_f$ and $\mathcal{L}_r$ using $\mathcal{E}_r$
  Backpropagate and update parameters $\theta = (\theta_g, \theta_h)$ via $\theta \leftarrow \theta - \alpha \nabla_\theta \mathcal{L}_u$
end for
  return updated model $\Phi' = h' \circ g' \circ e$

4 Experiments

4.1 Experimental Setup

We evaluate the efficacy of our proposed source-free framework by integrating it with a diverse set of state-of-the-art class unlearning methods, tested across three widely used benchmark datasets. Experiments are conducted using four backbone architectures, ResNet-18 (He et al., 2016), ResNet-50 (He et al., 2016), ViT-B/16 (Dosovitskiy et al., 2020), and Swin-T (Liu et al., 2021), although our framework is architecture-agnostic and can be extended to other network architectures without modification.

Datasets —We conduct experiments on CIFAR-10 (Krizhevsky et al., 2009), CIFAR-100 (Krizhevsky et al., 2009), and TinyImageNet (Le & Yang, 2015). CIFAR-10 and CIFAR-100 comprise 60,000 color images of resolution 32×32, split into 50,000 training and 10,000 testing samples, with 10 and 100 classes respectively. TinyImageNet contains 110,000 images of resolution 64×64, distributed across 200 classes, with 100,000 samples for training and 10,000 for testing. In this work, we utilize only the test sets of these datasets to evaluate the effectiveness of the unlearning methods within our source-free framework.

Baselines —We benchmark our approach against a comprehensive suite of methods, including classical retraining, fine-tuning-based unlearning, and recent state-of-the-art techniques such as Boundary Shrink (BS) (Chen et al., 2023), Boundary Expanding (BE) (Chen et al., 2023), DELETE (Zhou et al., 2025), SCRUB (Kurmanji et al., 2023), SCAR (Bonato et al., 2024), Negative Gradient (NG) (Golatkar et al., 2020), Negative Gradient+ (NG+) (Kurmanji et al., 2023), and Random Labels (RL) (Hayase et al., 2020). The *Original* models denote ResNet-18, ResNet-50, ViT-B/16, and Swin-T architectures trained on the full training set for 300 epochs with cosine annealing learning rate scheduling, serving as the baseline before unlearning. The *Retrained* models are trained from scratch for 200 epochs exclusively on the retain subset, representing an upper-bound performance as they have no exposure to data from the forget set.

Evaluation Metrics —We assess unlearning performance using three primary metrics, including retain test accuracy ($\mathcal{A}_r^t$), forget test accuracy ($\mathcal{A}_f^t$), and the Adaptive Unlearning Score (AUS) (Cotogni et al., 2023). The objective is to maximize $\mathcal{A}_r^t$, thereby preserving retain knowledge, while minimizing $\mathcal{A}_f^t$, indicating effective unlearning. The AUS combines these aspects into a single scalar score that balances utility and unlearning:

$$\text{AUS} = \Big(1 - \big(\mathcal{A}_r^{\text{or}-t} - \mathcal{A}_r^{\text{un}-t}\big)\Big) / \Big(1 + \Big|\mathcal{A}_f^{\text{ideal}-t} - \mathcal{A}_f^{\text{un}-t}\Big|\Big), \tag{16}$$

where $\mathcal{A}_r^{\text{or}-t}$ is the retain test accuracy of the original model, $\mathcal{A}_r^{\text{un}-t}$ and $\mathcal{A}_f^{\text{un}-t}$ are the retain and forget test accuracies of the unlearned model respectively, and $\mathcal{A}_f^{\text{ideal}-t}$ denotes the target forget accuracy (ideally zero). Higher AUS values indicate superior unlearning performance, i.e., effective forgetting while preserving the retain classes' accuracy. We additionally evaluate privacy leakage using two standard Membership Inference Attacks (MIAs) in Appendix G.

Implementation note. For theoretical clarity, Proposition 1 analyzes the head-only update regime with fixed g; in practice, our framework updates both θ_g and θ_h as in Algorithm 1.

4.2 Main Results

For each dataset, we conduct experiments using five independently initialized models, applying class-wise unlearning separately to each class. Each experiment is repeated across five random seeds, and the results reported correspond to the mean and standard deviation aggregated over all classes and seeds. To ensure a fair comparison among unlearning methods, the number of synthetic samples generated per class matches the size of the original training class (see Appendix A for the required minimum number of synthetic embeddings). These synthetic embeddings are sampled from the intermediate feature space immediately preceding the model's classification head (see Appendix D for the effect of embedding distribution). The overall performance is summarized in Table 1 and Table 2. Across all methods, datasets, and backbone architectures, our source-free framework consistently achieves near-complete forgetting as indicated by the minimized forget test accuracy ($\mathcal{A}_f^t$), while maintaining strong classification accuracy on retain classes ($\mathcal{A}_r^t$). Moreover, the AUS values closely approximate retraining-based baselines with full access to the retain set. In addition, a

detailed class-level evaluation of different unlearning methods within our source-free framework is provided in Appendix F, and the code link is included in Appendix C.

Table 1: Single-class unlearning performance for CIFAR-10, CIFAR-100, and TinyImageNet using ResNet-18 and ResNet-50 as the base architecture. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method. Columns $\mathcal{D}_r$-free and $\mathcal{D}_f$-free indicate whether the method operates without access to the retain or forget set, respectively, with (✓) denoting true and (✗) denoting false.

Method	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	CIFAR-10 $\mathcal{A}_r^t \uparrow$	CIFAR-10 $\mathcal{A}_f^t \downarrow$	CIFAR-10 AUS ↑	CIFAR-100 $\mathcal{A}_r^t \uparrow$	CIFAR-100 $\mathcal{A}_f^t \downarrow$	CIFAR-100 AUS ↑	TinyImageNet $\mathcal{A}_r^t \uparrow$	TinyImageNet $\mathcal{A}_f^t \downarrow$	TinyImageNet AUS ↑
ResNet-18:											
Original	–	–	86.58 ± 0.83	86.58 ± 6.67	0.537 ± 0.020	78.16 ± 1.07	78.16 ± 11.15	0.564 ± 0.037	71.30 ± 0.29	71.30 ± 12.46	0.587 ± 0.045
Retrained	–	–	86.95 ± 1.22	0.0 ± 0.0	1.000 ± 0.005	77.92 ± 0.80	0.0 ± 0.0	0.956 ± 0.036	63.01 ± 2.77	0.0 ± 0.0	0.855 ± 0.029
FT (Golatkar et al., 2020)	✗	✓	87.43 ± 1.02	0.0 ± 0.0	1.009 ± 0.004	78.20 ± 1.00	0.0 ± 0.0	1.000 ± 0.003	71.32 ± 0.35	0.0 ± 0.0	1.000 ± 0.002
	✓	✓	87.37 ± 1.11	0.0 ± 0.0	1.008 ± 0.003	78.29 ± 1.04	0.0 ± 0.0	1.001 ± 0.001	71.25 ± 0.32	0.0 ± 0.1	0.999 ± 0.001
NG (Golatkar et al., 2020)	✓	✗	87.31 ± 1.13	0.0 ± 0.0	1.007 ± 0.003	78.28 ± 1.07	0.0 ± 0.0	1.001 ± 0.001	71.36 ± 0.30	0.0 ± 0.0	1.001 ± 0.001
	✓	✓	87.40 ± 1.14	0.0 ± 0.0	1.008 ± 0.004	78.28 ± 1.05	0.0 ± 0.1	1.001 ± 0.002	71.30 ± 0.29	0.0 ± 0.0	1.001 ± 0.000
RL (Hayase et al., 2020)	✓	✗	87.43 ± 1.16	0.0 ± 0.0	1.008 ± 0.004	78.36 ± 1.05	0.0 ± 0.0	1.002 ± 0.001	71.35 ± 0.32	0.0 ± 0.0	1.001 ± 0.001
	✓	✓	87.33 ± 1.11	0.0 ± 0.0	1.008 ± 0.004	78.12 ± 1.03	0.0 ± 0.0	1.000 ± 0.001	71.27 ± 0.32	0.0 ± 0.0	1.000 ± 0.001
BS (Chen et al., 2023)	✓	✗	86.29 ± 1.09	0.2 ± 0.4	0.996 ± 0.009	74.32 ± 1.72	0.1 ± 0.5	0.960 ± 0.017	70.24 ± 0.87	0.1 ± 0.5	0.988 ± 0.010
	✓	✓	87.37 ± 1.16	0.0 ± 0.0	1.008 ± 0.004	77.27 ± 1.05	0.5 ± 3.2	0.987 ± 0.026	70.36 ± 0.99	0.0 ± 0.1	0.991 ± 0.009
BE (Chen et al., 2023)	✓	✗	84.72 ± 1.61	0.5 ± 1.2	0.977 ± 0.021	71.23 ± 2.43	0.1 ± 0.6	0.930 ± 0.024	62.68 ± 2.69	1.3 ± 2.1	0.902 ± 0.030
	✓	✓	86.51 ± 0.81	0.0 ± 0.0	0.999 ± 0.001	78.02 ± 1.10	0.0 ± 0.0	0.999 ± 0.003	71.23 ± 0.30	0.0 ± 0.0	0.999 ± 0.001
DELETE (Zhou et al., 2025)	✓	✗	87.33 ± 1.12	0.0 ± 0.0	1.008 ± 0.004	78.28 ± 1.06	0.0 ± 0.0	1.001 ± 0.001	71.43 ± 0.30	0.0 ± 0.0	1.001 ± 0.000
	✓	✓	87.36 ± 1.13	0.0 ± 0.0	1.008 ± 0.004	78.26 ± 1.07	0.0 ± 0.0	1.001 ± 0.001	71.36 ± 0.30	0.0 ± 0.0	1.001 ± 0.000
NG+ (Kurmanji et al., 2023)	✗	✗	85.31 ± 9.73	0.0 ± 0.0	0.987 ± 0.095	77.57 ± 6.40	0.0 ± 0.0	0.994 ± 0.062	71.21 ± 0.86	0.0 ± 0.0	0.999 ± 0.008
	✓	✓	87.38 ± 1.14	0.0 ± 0.0	1.008 ± 0.004	78.33 ± 1.00	0.0 ± 0.0	1.002 ± 0.001	71.35 ± 0.33	0.0 ± 0.0	1.000 ± 0.001
SCRUB (Kurmanji et al., 2023)	✗	✗	87.11 ± 1.04	0.0 ± 0.0	1.005 ± 0.003	77.52 ± 1.06	0.0 ± 0.0	0.994 ± 0.002	67.60 ± 1.51	0.0 ± 0.4	0.963 ± 0.014
	✓	✓	87.45 ± 1.17	0.0 ± 0.0	1.009 ± 0.004	78.22 ± 1.01	0.0 ± 0.0	1.001 ± 0.001	71.15 ± 0.37	0.0 ± 0.0	0.999 ± 0.001
SCAR (Bonato et al., 2024)	✗	✗	87.44 ± 1.15	0.0 ± 0.0	1.009 ± 0.004	78.34 ± 1.09	0.0 ± 0.0	1.002 ± 0.002	71.50 ± 0.30	0.0 ± 0.0	1.002 ± 0.001
	✓	✓	87.38 ± 1.12	0.0 ± 0.0	1.008 ± 0.004	78.33 ± 1.05	0.0 ± 0.0	1.002 ± 0.001	71.41 ± 0.30	0.0 ± 0.0	1.001 ± 0.000
ResNet-50:											
Original	–	–	88.28 ± 0.86	88.28 ± 5.92	0.532 ± 0.017	82.62 ± 0.79	82.62 ± 9.29	0.549 ± 0.029	75.91 ± 1.25	75.91 ± 11.32	0.571 ± 0.038
Retrained	–	–	89.03 ± 1.04	0.0 ± 0.0	1.008 ± 0.007	81.73 ± 0.99	0.0 ± 0.0	0.991 ± 0.013	76.21 ± 2.31	0.0 ± 0.0	1.003 ± 0.026
FT (Golatkar et al., 2020)	✗	✓	89.40 ± 0.98	0.0 ± 0.0	1.011 ± 0.005	82.79 ± 0.75	0.0 ± 0.0	1.002 ± 0.001	75.80 ± 1.25	0.0 ± 0.2	0.999 ± 0.003
	✓	✓	88.98 ± 1.03	0.0 ± 0.0	1.007 ± 0.003	82.68 ± 0.77	0.0 ± 0.0	1.001 ± 0.001	75.80 ± 1.29	0.0 ± 0.0	0.999 ± 0.001
NG (Golatkar et al., 2020)	✓	✗	88.96 ± 1.66	0.0 ± 0.0	1.005 ± 0.013	82.71 ± 0.79	0.0 ± 0.0	1.001 ± 0.001	75.97 ± 1.24	0.0 ± 0.0	1.001 ± 0.000
	✓	✓	89.04 ± 1.10	0.0 ± 0.0	1.008 ± 0.004	82.70 ± 0.79	0.0 ± 0.0	1.001 ± 0.001	75.95 ± 1.25	0.0 ± 0.0	1.000 ± 0.000
RL (Hayase et al., 2020)	✓	✗	89.06 ± 1.07	0.0 ± 0.0	1.008 ± 0.003	82.72 ± 0.79	0.0 ± 0.0	1.001 ± 0.001	75.95 ± 1.24	0.0 ± 0.0	1.000 ± 0.001
	✓	✓	88.92 ± 1.04	0.0 ± 0.0	1.006 ± 0.003	82.76 ± 0.78	0.0 ± 0.0	1.001 ± 0.001	75.90 ± 1.22	0.0 ± 0.0	1.000 ± 0.001
BS (Chen et al., 2023)	✓	✗	87.68 ± 1.18	0.4 ± 0.9	0.990 ± 0.014	82.28 ± 0.94	0.0 ± 0.1	0.997 ± 0.003	74.44 ± 1.67	0.1 ± 0.5	0.984 ± 0.013
	✓	✓	89.24 ± 0.97	0.0 ± 0.0	1.007 ± 0.003	82.55 ± 0.80	0.0 ± 0.0	0.999 ± 0.001	75.19 ± 1.21	0.0 ± 0.0	0.993 ± 0.002
BE (Chen et al., 2023)	✓	✗	87.44 ± 1.56	0.3 ± 0.9	0.989 ± 0.015	82.14 ± 0.85	0.0 ± 0.0	0.995 ± 0.002	68.12 ± 2.81	0.5 ± 1.2	0.917 ± 0.021
	✓	✓	88.22 ± 0.86	0.0 ± 0.0	0.999 ± 0.000	82.62 ± 0.79	0.0 ± 0.0	1.000 ± 0.000	75.89 ± 1.25	0.0 ± 0.0	1.000 ± 0.000
DELETE (Zhou et al., 2025)	✓	✗	88.99 ± 1.06	0.0 ± 0.0	1.007 ± 0.003	82.71 ± 0.79	0.0 ± 0.0	1.001 ± 0.001	75.98 ± 1.24	0.0 ± 0.0	1.001 ± 0.000
	✓	✓	88.98 ± 1.07	0.0 ± 0.0	1.007 ± 0.003	82.70 ± 0.79	0.0 ± 0.0	1.001 ± 0.001	75.95 ± 1.25	0.0 ± 0.0	1.000 ± 0.000
NG+ (Kurmanji et al., 2023)	✗	✗	89.12 ± 1.00	0.0 ± 0.0	1.008 ± 0.004	82.78 ± 0.77	0.0 ± 0.0	1.002 ± 0.001	76.24 ± 1.06	0.0 ± 0.0	1.001 ± 0.001
	✓	✓	88.99 ± 1.05	0.0 ± 0.0	1.007 ± 0.003	82.79 ± 0.90	0.0 ± 0.0	1.001 ± 0.001	75.99 ± 1.23	0.0 ± 0.0	1.001 ± 0.000
SCRUB (Kurmanji et al., 2023)	✗	✗	88.96 ± 0.95	0.0 ± 0.0	1.008 ± 0.003	82.76 ± 0.75	0.0 ± 0.0	1.001 ± 0.001	70.65 ± 2.51	0.3 ± 1.0	0.944 ± 0.015
	✓	✓	89.11 ± 1.10	0.0 ± 0.0	1.008 ± 0.004	82.72 ± 0.77	0.0 ± 0.0	1.001 ± 0.001	75.86 ± 1.28	0.0 ± 0.0	0.999 ± 0.001
SCAR (Bonato et al., 2024)	✗	✗	89.11 ± 1.08	0.0 ± 0.0	1.008 ± 0.004	82.47 ± 0.97	0.0 ± 0.1	0.998 ± 0.008	76.01 ± 1.22	0.0 ± 0.0	1.001 ± 0.001
	✓	✓	89.02 ± 1.07	0.0 ± 0.0	1.007 ± 0.003	82.73 ± 0.79	0.0 ± 0.0	1.001 ± 0.001	76.04 ± 1.24	0.0 ± 0.0	1.001 ± 0.000

Impact of Embedding Location on Source-Free Unlearning —To evaluate the flexibility of our framework, we examine how the depth at which synthetic embeddings are generated influences unlearning performance. Specifically, we compare embeddings produced at two distinct locations: (1) immediately preceding the classifier head, which serves as our default configuration, and (2) earlier in the network, e.g., before the final convolutional block within ResNet-18's layer 4. As reported in Table 3, embeddings generated at the earlier stage continue to deliver strong unlearning performance, with results closely matching those obtained from embeddings sampled before the classifier head (see Table 1). The marginal differences observed underscore the robustness of our method to the choice of embedding depth. Furthermore, synthetic embeddings consistently achieve competitive results when directly compared to original embeddings extracted from

Table 2: Single-class unlearning performance for CIFAR-10, CIFAR-100, and TinyImageNet using ViT-B/16 and Swin-T as the base architecture. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method. Columns $\mathcal{D}_r$-free and $\mathcal{D}_f$-free indicate whether the method operates without access to the retain or forget set, respectively, with (✓) denoting true and (✗) denoting false.

Method	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	CIFAR-10 $\mathcal{A}_r^t \uparrow$	CIFAR-10 $\mathcal{A}_f^t \downarrow$	CIFAR-10 AUS ↑	CIFAR-100 $\mathcal{A}_r^t \uparrow$	CIFAR-100 $\mathcal{A}_f^t \downarrow$	CIFAR-100 AUS ↑	TinyImageNet $\mathcal{A}_r^t \uparrow$	TinyImageNet $\mathcal{A}_f^t \downarrow$	TinyImageNet AUS ↑
ViT-B/16:											
Original	–	–	97.69 ± 0.18	97.69 ± 1.30	0.506 ± 0.003	87.22 ± 0.26	87.22 ± 7.83	0.535 ± 0.023	88.20 ± 0.14	88.20 ± 7.29	0.532 ± 0.022
Retrained	–	–	98.38 ± 0.21	0.0 ± 0.0	1.007 ± 0.002	88.74 ± 0.21	0.0 ± 0.0	1.015 ± 0.003	89.59 ± 0.13	0.0 ± 0.0	1.014 ± 0.002
NG (Golatkar et al., 2020)	✓	✗	97.89 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.29 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	✓	✓	97.90 ± 0.24	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
RL (Hayase et al., 2020)	✓	✗	97.91 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.31 ± 0.28	0.0 ± 0.0	1.001 ± 0.001	88.24 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	✓	✓	97.93 ± 0.24	0.0 ± 0.0	1.002 ± 0.001	87.35 ± 0.28	0.0 ± 0.0	1.001 ± 0.001	88.27 ± 0.14	0.0 ± 0.0	1.001 ± 0.001
BS (Chen et al., 2023)	✓	✗	97.76 ± 0.22	0.0 ± 0.0	1.001 ± 0.001	87.27 ± 0.27	0.0 ± 0.0	1.000 ± 0.000	88.22 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	✓	✓	97.89 ± 0.23	0.0 ± 0.0	1.002 ± 0.001	87.22 ± 0.28	0.0 ± 0.0	1.000 ± 0.001	88.08 ± 0.16	0.0 ± 0.1	0.999 ± 0.001
DELETE (Zhou et al., 2025)	✓	✗	97.89 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	✓	✓	97.91 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.32 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.25 ± 0.14	0.0 ± 0.0	1.001 ± 0.000
NG+ (Kurmanji et al., 2023)	✗	✗	97.88 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.15 ± 0.29	0.0 ± 0.2	0.999 ± 0.003	87.64 ± 0.27	0.1 ± 0.4	0.993 ± 0.005
	✓	✓	97.92 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.32 ± 0.30	0.0 ± 0.0	1.001 ± 0.001	88.28 ± 0.15	0.0 ± 0.0	1.001 ± 0.000
Swin-T:											
Original	–	–	97.73 ± 0.17	97.73 ± 1.47	0.506 ± 0.004	87.58 ± 0.53	87.58 ± 9.01	0.534 ± 0.029	86.18 ± 0.09	86.18 ± 7.59	0.538 ± 0.023
Retrained	–	–	98.36 ± 0.23	0.0 ± 0.0	1.006 ± 0.001	88.89 ± 0.21	0.0 ± 0.0	1.013 ± 0.005	87.13 ± 0.13	0.0 ± 0.0	1.010 ± 0.002
NG (Golatkar et al., 2020)	✓	✗	97.93 ± 0.27	0.0 ± 0.0	1.002 ± 0.001	87.65 ± 0.54	0.0 ± 0.0	1.001 ± 0.001	86.21 ± 0.10	0.0 ± 0.0	1.000 ± 0.000
	✓	✓	97.64 ± 0.86	0.5 ± 1.0	0.995 ± 0.017	83.19 ± 3.93	1.7 ± 1.7	0.941 ± 0.047	80.79 ± 4.72	1.9 ± 1.6	0.929 ± 0.051
NG+ (Kurmanji et al., 2023)	✗	✗	97.83 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	87.60 ± 0.54	0.0 ± 0.0	1.000 ± 0.002	84.46 ± 1.19	0.0 ± 0.3	0.982 ± 0.012
	✓	✓	93.50 ± 7.54	1.1 ± 1.3	0.948 ± 0.080	86.84 ± 0.95	0.3 ± 0.8	0.990 ± 0.014	85.28 ± 0.76	0.4 ± 1.0	0.987 ± 0.014
SCRUB (Kurmanji et al., 2023)	✗	✗	97.85 ± 0.25	0.0 ± 0.0	1.001 ± 0.001	87.73 ± 0.47	0.0 ± 0.0	1.001 ± 0.001	86.19 ± 0.09	0.0 ± 0.0	1.000 ± 0.001
	✓	✓	97.39 ± 1.11	0.0 ± 0.0	0.997 ± 0.011	87.07 ± 0.65	0.0 ± 0.3	0.995 ± 0.007	84.92 ± 0.73	0.1 ± 0.4	0.987 ± 0.008

the same intermediate layer, indicating their effectiveness as surrogate representations. Collectively, these findings confirm that our framework supports effective unlearning at multiple depths within the network, offering a layer-agnostic capability that enhances adaptability to diverse architectural configurations, privacy considerations, and computational constraints, thereby broadening its practical applicability.

Table 3: Single-class unlearning performance using synthetic embedding generated from layer 4 (immediately before the last convolutional layer) of ResNet-18 as the base architecture for CIFAR-10, CIFAR-100, and TinyImageNet. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method. Columns $\mathcal{D}_r$-free and $\mathcal{D}_f$-free indicate whether the method operates without access to the retain or forget set, respectively, with (✓) denoting true and (✗) denoting false.

Method	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	CIFAR-10 $\mathcal{A}_r^t \uparrow$	CIFAR-10 $\mathcal{A}_f^t \downarrow$	CIFAR-10 AUS ↑	CIFAR-100 $\mathcal{A}_r^t \uparrow$	CIFAR-100 $\mathcal{A}_f^t \downarrow$	CIFAR-100 AUS ↑	TinyImageNet $\mathcal{A}_r^t \uparrow$	TinyImageNet $\mathcal{A}_f^t \downarrow$	TinyImageNet AUS ↑
Original	–	–	86.58 ± 0.83	86.58 ± 6.67	0.537 ± 0.020	78.16 ± 1.07	78.16 ± 11.15	0.564 ± 0.037	71.30 ± 0.29	71.30 ± 12.46	0.587 ± 0.045
Retrained	–	–	86.95 ± 1.22	0.0 ± 0.0	1.000 ± 0.005	77.92 ± 0.80	0.0 ± 0.0	0.956 ± 0.036	63.01 ± 2.77	0.0 ± 0.0	0.855 ± 0.029
FT (Golatkar et al., 2020)	✗	✓	87.55 ± 1.09	0.2 ± 0.9	1.007 ± 0.010	76.80 ± 4.06	0.2 ± 0.6	0.985 ± 0.042	71.72 ± 0.33	0.6 ± 1.2	0.998 ± 0.012
	✓	✓	81.03 ± 3.82	0.0 ± 0.1	0.944 ± 0.037	76.09 ± 1.10	0.0 ± 0.3	0.979 ± 0.009	69.64 ± 0.46	0.0 ± 0.0	0.983 ± 0.002
NG (Golatkar et al., 2020)	✓	✗	87.30 ± 1.23	0.0 ± 0.0	1.007 ± 0.005	78.29 ± 1.08	0.0 ± 0.0	1.001 ± 0.001	70.51 ± 1.02	0.1 ± 0.5	0.991 ± 0.011
	✓	✓	87.24 ± 1.16	0.0 ± 0.1	1.006 ± 0.004	76.28 ± 1.40	0.0 ± 0.1	0.981 ± 0.011	71.30 ± 0.46	0.0 ± 0.0	1.000 ± 0.003
RL (Hayase et al., 2020)	✓	✗	87.27 ± 1.08	0.0 ± 0.0	1.007 ± 0.003	78.32 ± 1.06	0.0 ± 0.0	1.002 ± 0.001	71.56 ± 0.39	0.0 ± 0.0	1.003 ± 0.001
	✓	✓	87.18 ± 1.24	0.0 ± 0.1	1.006 ± 0.007	77.76 ± 1.65	0.0 ± 0.2	0.996 ± 0.013	71.62 ± 0.45	0.0 ± 0.0	1.003 ± 0.002
DELETE (Zhou et al., 2025)	✓	✗	77.62 ± 15.23	0.4 ± 0.8	0.905 ± 0.150	75.97 ± 4.21	0.1 ± 0.6	0.978 ± 0.039	54.84 ± 6.63	1.4 ± 1.7	0.819 ± 0.069
	✓	✓	87.02 ± 1.11	0.0 ± 0.1	1.004 ± 0.005	74.29 ± 2.31	1.3 ± 1.4	0.948 ± 0.026	68.89 ± 0.99	0.0 ± 0.3	0.972 ± 0.010
NG+ (Kurmanji et al., 2023)	✗	✗	83.82 ± 0.70	0.0 ± 0.0	0.972 ± 0.010	78.20 ± 1.01	0.0 ± 0.1	1.000 ± 0.002	70.41 ± 0.44	0.0 ± 0.0	0.991 ± 0.003
	✓	✓	87.16 ± 1.17	0.1 ± 0.5	1.005 ± 0.007	78.18 ± 1.06	0.0 ± 0.2	1.000 ± 0.004	71.37 ± 0.43	0.0 ± 0.1	1.001 ± 0.002

Impact of the Number of Synthetic Embeddings per Class on Unlearning Performance —We investigate how the number of synthetic embeddings generated per class influences the unlearning efficacy. To this end, the ResNet-18 trained on CIFAR-100 is considered in the main text, with additional results for

ResNet-18 on CIFAR-10 and TinyImageNet, as well as ViT-B/16 on CIFAR-10 and CIFAR-100, provided in the Appendix E. As illustrated in Figure 2, increasing the number of synthetic samples consistently enhances retain class accuracy ($\mathcal{A}_r^t$) and the AUS, while reducing forget class accuracy ($\mathcal{A}_f^t$). This behavior indicates that generating a larger set of representative embeddings more effectively approximates the decision boundaries of the forget and retain classes, thereby improving source-free unlearning performances. Notably, performance gains saturate beyond a certain sample size, which means that generating additional synthetic embeddings beyond this point yields minimal improvement. This allows for efficient use of computational resources without compromising unlearning quality.

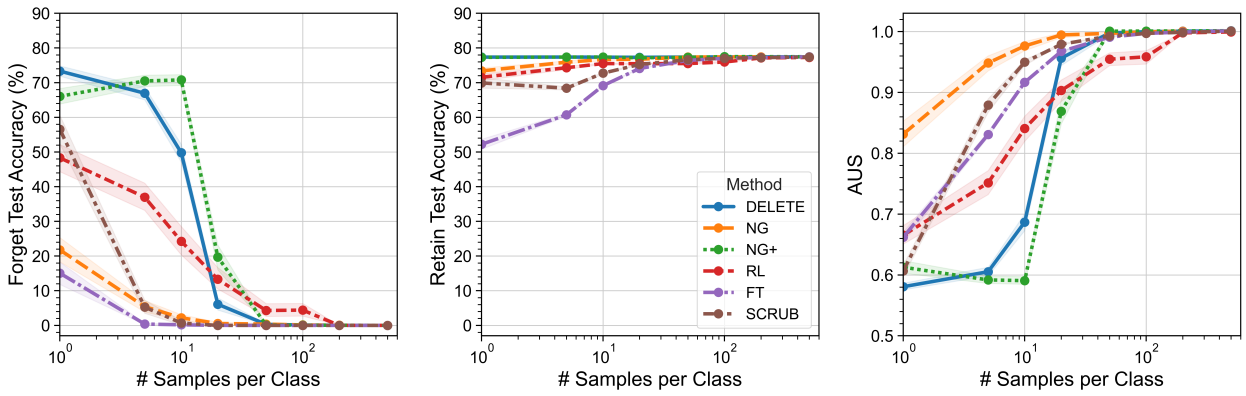

Figure 2: Effect of the number of synthetic embeddings per class on unlearning performance. Results are averaged over three independently trained models, with class-wise unlearning performed separately for each class. Error bars indicate 95% confidence intervals. Experiments use the ResNet-18 architecture on the CIFAR-100 dataset.

Multi-class Unlearning Setting —Beyond the single-class unlearning setting, we evaluate whether our source-free class-unlearning framework scale to multi-class setting on CIFAR-100 using a ResNet-18 backbone (Table 4). We consider unlearning 2, 5, and 10 classes, with label sets $\mathcal{Y}_f = \{25, 58\}$, $\mathcal{Y}_f = \{25, 58, 38, 23, 96\}$, and $\mathcal{Y}_f = \{25, 58, 38, 23, 96, 54, 51, 49, 98, 66\}$, respectively, following the CIFAR-100 setup in (Zhou et al., 2025). In our multi-class experiments, all classes in $\mathcal{Y}_f$ are forgotten simultaneously in a single unlearning run. Each experiment is repeated across five random seeds. Additional results for larger forget sets with 20, 40, and 60 forget classes are provided in Appendix J.

Table 4: Multi-class unlearning performance for CIFAR-100 using ResNet-18 as the base architecture. Rows highlighted in gray correspond to methods applied on synthetic embeddings, while the non-shaded rows use original embeddings. Columns $\mathcal{D}_r$-free and $\mathcal{D}_f$-free indicate whether the method operates without access to the retain or forget set, respectively, with (✓) denoting true and (✗) denoting false.

Method	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	2-Classes $\mathcal{A}_r^t \uparrow$	2-Classes $\mathcal{A}_f^t \downarrow$	2-Classes AUS ↑	5-Classes $\mathcal{A}_r^t \uparrow$	5-Classes $\mathcal{A}_f^t \downarrow$	5-Classes AUS ↑	10-Classes $\mathcal{A}_r^t \uparrow$	10-Classes $\mathcal{A}_f^t \downarrow$	10-Classes AUS ↑
Original	–	–	78.12 ± 1.21	80.10 ± 3.19	0.555 ± 0.010	78.14 ± 1.26	78.68 ± 0.92	0.560 ± 0.003	78.01 ± 1.31	79.58 ± 0.93	0.557 ± 0.003
Retrained	–	–	80.10	0.00	1.006	78.91	0.00	1.023	78.00	0.00	1.005
FT (Golatkar et al., 2020)	✗	✓	78.24 ± 1.07	0.00 ± 0.00	1.001 ± 0.003	78.63 ± 1.22	0.00 ± 0.00	1.005 ± 0.003	78.80 ± 1.04	0.02 ± 0.04	1.008 ± 0.004
	✓	✓	78.26 ± 1.17	0.00 ± 0.00	1.001 ± 0.001	78.60 ± 1.18	0.00 ± 0.00	1.005 ± 0.001	78.88 ± 1.17	0.02 ± 0.04	1.009 ± 0.002
NG (Golatkar et al., 2020)	✓	✗	78.37 ± 1.15	0.00 ± 0.00	1.002 ± 0.001	78.67 ± 1.19	0.00 ± 0.00	1.005 ± 0.001	78.97 ± 1.20	0.00 ± 0.00	1.010 ± 0.002
	✓	✓	78.34 ± 1.12	0.00 ± 0.00	1.002 ± 0.001	78.68 ± 1.13	0.00 ± 0.00	1.005 ± 0.002	78.99 ± 1.12	0.00 ± 0.00	1.010 ± 0.002
RL (Hayase et al., 2020)	✓	✗	78.25 ± 1.12	0.00 ± 0.00	1.001 ± 0.002	78.10 ± 1.07	0.00 ± 0.00	1.000 ± 0.003	78.62 ± 1.10	0.00 ± 0.00	1.006 ± 0.003
	✓	✓	77.95 ± 1.03	0.00 ± 0.00	0.998 ± 0.003	76.25 ± 0.81	0.04 ± 0.09	0.981 ± 0.011	74.38 ± 1.41	0.18 ± 0.35	0.962 ± 0.014
DELETE (Zhou et al., 2025)	✓	✗	78.37 ± 1.12	0.00 ± 0.00	1.002 ± 0.001	78.71 ± 1.14	0.00 ± 0.00	1.006 ± 0.001	79.01 ± 1.13	0.00 ± 0.00	1.010 ± 0.002
	✓	✓	78.33 ± 1.13	0.00 ± 0.00	1.002 ± 0.001	78.66 ± 1.15	0.00 ± 0.00	1.005 ± 0.001	78.96 ± 1.14	0.66 ± 1.01	1.003 ± 0.010
NG+ (Kurmanji et al., 2023)	✗	✗	78.47 ± 1.05	0.00 ± 0.00	1.003 ± 0.002	78.79 ± 1.04	0.00 ± 0.00	1.006 ± 0.003	79.14 ± 1.02	0.00 ± 0.00	1.011 ± 0.003
	✓	✓	78.34 ± 1.10	0.00 ± 0.00	1.002 ± 0.001	78.63 ± 1.11	0.00 ± 0.00	1.005 ± 0.002	78.97 ± 1.13	0.00 ± 0.00	1.010 ± 0.002
SCRUB (Kurmanji et al., 2023)	✗	✗	77.61 ± 1.01	0.00 ± 0.00	0.995 ± 0.003	78.27 ± 1.05	0.00 ± 0.00	1.001 ± 0.003	78.93 ± 1.07	0.00 ± 0.00	1.009 ± 0.003
	✓	✓	78.26 ± 1.04	0.00 ± 0.00	1.001 ± 0.002	78.48 ± 1.12	0.00 ± 0.00	1.003 ± 0.003	78.52 ± 1.08	0.00 ± 0.00	1.005 ± 0.004
SCAR (Bonato et al., 2024)	✗	✗	78.44 ± 1.13	0.00 ± 0.00	1.003 ± 0.001	78.79 ± 1.14	0.00 ± 0.00	1.007 ± 0.002	79.12 ± 1.14	0.00 ± 0.00	1.011 ± 0.002
	✓	✓	78.41 ± 1.13	0.00 ± 0.00	1.003 ± 0.001	78.77 ± 1.14	0.00 ± 0.00	1.006 ± 0.001	79.08 ± 1.13	0.00 ± 0.00	1.011 ± 0.002

5 Conclusion

We introduced a novel source-free framework for class unlearning, which removes specific class knowledge from a trained model without requiring access to the original training data, including forget, retain, or surrogate sets. By leveraging the internal structure of the model to synthesize class-conditional embeddings, we enable the adaptation of various state-of-the-art unlearning techniques to a fully source-free regime. Our experiments demonstrate that the proposed approach retains high accuracy on retain classes while effectively forgetting the target class across multiple datasets and unlearning strategies. The framework's compatibility with existing methods and complete independence from training data position it as a strong candidate for class unlearning in real-world scenarios. A practical limitation, however, is that the framework depends on a reasonably well-trained classifier to induce reliable pseudo-label partitions. If the original model is poorly calibrated or strongly imbalanced, some pseudo-label regions may become sparse or unreliable. In practice, we mitigate this by enforcing equal numbers of synthetic samples per class when constructing the pseudo-labeled sets, although additional raw synthetic draws may still be needed in large-class settings or when calibration is poor. Future work includes extending this approach to instance-level unlearning and applying the technique to domains beyond image classification, such as language models.

6 Acknowledgements

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), with additional computational resources provided by the Digital Research Alliance of Canada. We thank Dr. Maxime Darrin for the helpful discussions and valuable comments.

References

Sk Miraj Ahmed, Umit Yigit Basaran, Dripta S Raychaudhuri, Arindam Dutta, Rohit Kundu, Fahim Faisal Niloy, Basak Guler, and Amit K Roy-Chowdhury. Towards source-free machine unlearning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4948–4957, 2025.

Jacopo Bonato, Marco Cotogni, and Luigi Sabetta. Is retain set all you need in machine unlearning? restoring performance of unlearned models with out-of-distribution images. In *European Conference on Computer Vision*, pp. 1–19. Springer, 2024.

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

Sungmin Cha, Sungjun Cho, Dasol Hwang, Honglak Lee, Taesup Moon, and Moontae Lee. Learning to unlearn: Instance-wise unlearning for pre-trained classifiers. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 11186–11194, 2024.

Huiqiang Chen, Tianqing Zhu, Xin Yu, and Wanlei Zhou. Machine unlearning via null space calibration. *arXiv preprint arXiv:2404.13588*, 2024.

Huiqiang Chen, Tianqing Zhu, Xin Yu, and Wanlei Zhou. Zero-shot machine unlearning with proxy adversarial data generation. *arXiv preprint arXiv:2507.21738*, 2025.

Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7766–7775, 2023.

Si Chen, Mostafa Kahla, Ruoxi Jia, and Guo-Jun Qi. Knowledge-enriched distributional model inversion attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16178–16187, 2021.

Xinwen Cheng, Zhehao Huang, Wenxin Zhou, Zhengbao He, Ruikai Yang, Yingwen Wu, and Xiaolin Huang. Remaining-data-free machine unlearning by suppressing sample contribution. *arXiv preprint arXiv:2402.15109*, 2024.

Vikram S Chundawat, Ayush K Tarun, Murari Mandal, and Mohan Kankanhalli. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*, 18:2345–2354, 2023.

Marco Cotogni, Jacopo Bonato, Luigi Sabetta, Francesco Pelosin, and Alessandro Nicolosi. Duck: distance-based unlearning via centroid kinematics. *arXiv preprint arXiv:2312.02052*, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. *arXiv preprint arXiv:2310.12508*, 2023.

Jack Foster, Kyle Fogarty, Stefan Schoepf, Cengiz Öztireli, and Alexandra Brintrup. Zero-shot machine unlearning at scale via lipschitz regularization. *CoRR*, 2024.

Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pp. 1322–1333, 2015.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9304–9312, 2020.

Eric Goldman. An introduction to the california consumer privacy act (ccpa). *Santa Clara Univ. Legal Studies Research Paper*, pp. 1–, 2020.

Tomohiro Hayase, Suguru Yasutomi, and Takashi Katoh. Selective forgetting of deep networks at a finer level than samples. *arXiv preprint arXiv:2012.11849*, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. *Advances in Neural Information Processing Systems*, 36:51584–51605, 2023.

Sangamesh Kodge, Gobinda Saha, and Kaushik Roy. Deep unlearning: Fast and efficient gradient-free class forgetting. *Transactions on Machine Learning Research*, 2024.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36:1957–1987, 2023.

Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Shen Lin, Xiaoyu Zhang, Chenyang Chen, Xiaofeng Chen, and Willy Susilo. Erm-ktp: Knowledge-level machine unlearning via knowledge transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20147–20155, 2023.

Yang Liu, Mingyuan Fan, Cen Chen, Ximeng Liu, Zhuo Ma, Li Wang, and Jianfeng Ma. Backdoor defense with machine unlearning. In *IEEE INFOCOM 2022-IEEE conference on computer communications*, pp. 280–289. IEEE, 2022.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Aviraj Newatia, Michael Cooper, and Rahul Krishnan. Unlearning tabular data without a" forget set". In *NeurIPS 2024 Third Table Representation Learning Workshop*, 2024.

Subhodip Panda, Shashwat Sourav, et al. Partially blinded unlearning: Class unlearning for deep networks from bayesian perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 6372–6380, 2025.

Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246*, 2018.

Seonguk Seo, Dongwan Kim, and Bohyung Han. Revisiting machine unlearning with dimensional alignment. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3206–3215. IEEE, 2025.

Nazanin Mohammadi Sepahvand, Eleni Triantafillou, Hugo Larochelle, Doina Precup, James J Clark, Daniel M Roy, and Gintare Karolina Dziugaite. Selective unlearning via representation erasure using domain adversarial training. In *The Thirteenth International Conference on Learning Representations*, 2025.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, 2017.

Liwei Song and Prateek Mittal. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX security symposium (USENIX security 21)*, pp. 2615–2632, 2021.

Liwei Song, Reza Shokri, and Prateek Mittal. Privacy risks of securing machine learning models against adversarial examples. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pp. 241–257, 2019.

Ayush K Tarun, Vikram S Chundawat, Murari Mandal, and Mohan Kankanhalli. Fast yet effective machine unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(9):13046–13055, 2023.

Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10(3152676):10–5555, 2017.

Xiuyuan Wang, Chaochao Chen, Weiming Liu, Xinting Liao, Fan Wang, and Xiaolin Zheng. Efficient source-free unlearning via energy-guided data synthesis and discrimination-aware multitask optimization. In *Forty-second International Conference on Machine Learning*, 2025.

Haonan Yan, Xiaoguang Li, Ziyao Guo, Hui Li, Fenghua Li, and Xiaodong Lin. Arcane: An efficient architecture for exact machine unlearning. In *IJCAI*, volume 6, pp. 19, 2022.

Shanshan Ye, Jie Lu, and Guangquan Zhang. Towards safe machine unlearning: A paradigm that mitigates performance degradation. In *Proceedings of the ACM on Web Conference 2025*, pp. 4635–4652, 2025.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pp. 268–282. IEEE, 2018.

Chenhao Zhang, Shaofei Shen, Weitong Chen, and Miao Xu. Toward efficient data-free unlearning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 22372–22379, 2025.

Yu Zhou, Dian Zheng, Qijie Mo, Renjie Lu, Kun-Yu Lin, and Wei-Shi Zheng. Decoupled distillation to erase: A general unlearning method for any class-centric tasks. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 20350–20359, 2025.

A Determining the Minimum Number of Synthetic Embeddings for Reliable Class Coverage

In the proposed source-free settings, synthetic embeddings are generated by sampling random vectors in the classifier's intermediate embedding space. The underlying sampling distribution significantly influences predicted class distribution, often causing class imbalance. To address this, we employ a class-aware rejection sampling strategy that continues sampling until a predefined minimum number of samples is obtained for each class. This ensures a balanced synthetic dataset and establishes a stable basis for source-free unlearning. To guarantee sufficient representation of all target classes, we estimate the minimum number of synthetic samples N required such that the probability of having at least one sample from a given class c exceeds a confidence threshold p. We first generate a large pilot batch $\{z_i\}_{i=1}^{N_{\text{pilot}}}$ of embeddings sampled from an arbitrary distribution in the intermediate embedding space, and obtain their predicted labels $\hat{y}_i$. The empirical class probability for class c is then estimated as

$$q_c = \frac{1}{N_{\text{pilot}}} \sum_{i=1}^{N_{\text{pilot}}} \mathbb{1}\{\hat{y}_i = c\}, \tag{17}$$

where $\mathbb{1}\{\cdot\}$ is the indicator function that equals one if the condition inside is true, and zero otherwise. Assuming independent sampling, the probability that none of the N synthetic embeddings fall into class c is $(1 - q_c)^N$. To ensure that at least one embedding belongs to class c with confidence p, we require $1 - (1 - q_c)^N \geq p$, which yields

$$N \geq \frac{\ln(1-p)}{\ln(1-q_c)}, \tag{18}$$

where $\ln(1 - q_c) < 0$ ensures the inequality holds in the correct direction. This expression provides a principled estimate for the number of synthetic embeddings required to achieve class-wise coverage with the desired confidence level.

We empirically validate this estimate by reporting the minimum number of synthetic embeddings required to ensure, with high confidence, that at least one embedding is classified into each target class. Table 5 summarizes statistics computed for a ResNet-18 classifier on CIFAR-10, CIFAR-100, and TinyImageNet datasets, using Gaussian, Laplace, and Uniform embedding distributions. We report the lower bound, average, and upper bound for the total number of synthetic embeddings needed across all classes for each dataset and embedding distribution. These values correspond, respectively, to the easiest, average, and most difficult classes to cover. This analysis shows the impact of dataset complexity and embeddings distribution on sample requirements for achieving reliable class representation in source-free unlearning.

Table 5: Estimated minimum total number of synthetic embeddings required to guarantee, with high confidence, that a forget class is represented by at least one embedding. Results correspond to the ResNet-18 architecture evaluated on CIFAR-10, CIFAR-100, and TinyImageNet datasets, using Gaussian, Laplace, and Uniform distributions for embedding generation.

Dataset	Embedding Distribution	Lower bound (across classes)	Average (across classes)	Upper bound (across classes)
CIFAR-10	Gaussian	32	46	55
	Laplace	33	46	53
	Uniform	29	48	60
CIFAR-100	Gaussian	223	494	1041
	Laplace	269	483	822
	Uniform	139	544	1735
TinyImageNet	Gaussian	407	990	2550
	Laplace	427	987	2437
	Uniform	353	1011	2880

In the worst-case scenario, where the rarest class has empirical probability $q_{\min}$, the minimum number of synthetic embeddings needed to ensure, with confidence p, that at least one embedding belongs to this class is $N_{\text{worst}} = \frac{\ln(1-p)}{\ln(1-q_{\min})}$. If a stricter criterion is imposed to require at least m embeddings from this rarest class, the required number of embeddings increases significantly. This corresponds to solving

$$1 - \sum_{k=0}^{m-1} \binom{N}{k} q_{\min}^k (1 - q_{\min})^{N-k} \geq p, \tag{19}$$

which involves computing the cumulative distribution function of a Binomial distribution. Although no closed-form solution exists, this inequality can be estimated numerically.

B Finite-Sample Concentration of Source-Free Logit Updates

Proposition 3 (Finite-Sample Concentration of Empirical Logit Steering)**.** *Assume the setup of Proposition 1. Let $\mathbf{z}_l \sim p_f$ be a probe synthetic forget embedding, and define for class $k \in \mathcal{Y}$*

$$X_k(\mathbf{z}, \mathbf{z}_l) := \frac{\partial \mathcal{L}_f}{\partial [h(g(\mathbf{z}))]_k} \, g(\mathbf{z})^\top g(\mathbf{z}_l). \tag{20}$$

Assume there exists $B_k > 0$ such that

$$|X_k(\mathbf{z}, \mathbf{z}_l)| \leq B_k \qquad \textit{almost surely under } \mathbf{z}, \mathbf{z}_l \sim p_f. \tag{21}$$

Let

$$\mu_k := \mathbb{E}_{\mathbf{z}, \mathbf{z}_l \sim p_f}[X_k(\mathbf{z}, \mathbf{z}_l)].$$

Then, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the sampling of $\mathcal{E}_f$ (and $\mathbf{z}_l \sim p_f$), the empirical one-step logit shift satisfies

$$\left| \left([h(g(\mathbf{z}_l))]_k^{(j+1)} - [h(g(\mathbf{z}_l))]_k^{(j)} \right) + \alpha \mu_k \right| \leq \alpha B_k \sqrt{\frac{\log(2/\delta)}{2N_f}}. \tag{22}$$

In particular, if

$$|\mu_k| > B_k \sqrt{\frac{\log(2/\delta)}{2N_f}}, \tag{23}$$

then the empirical one-step logit shift has the same sign as the population-level shift with probability at least $1 - \delta$.

Proof sketch. From the logit update identity in equation 11, for fixed probe embedding $\mathbf{z}_l$,

$$[h(g(\mathbf{z}_l))]_k^{(j+1)} - [h(g(\mathbf{z}_l))]_k^{(j)} = -\frac{\alpha}{N_f} \sum_{i=1}^{N_f} X_k(\mathbf{z}_i, \mathbf{z}_l), \tag{24}$$

where $\mathbf{z}_i \in \mathcal{E}_f$ are sampled from the synthetic forget distribution p_f (population view). By the boundedness assumption equation 21, Hoeffding's inequality gives, for fixed $\mathbf{z}_l$,

$$\Pr\left(\left| \frac{1}{N_f} \sum_{i=1}^{N_f} X_k(\mathbf{z}_i, \mathbf{z}_l) - \mathbb{E}_{\mathbf{z} \sim p_f}[X_k(\mathbf{z}, \mathbf{z}_l)] \right| \geq t \;\middle|\; \mathbf{z}_l \right) \leq 2 \exp\left(-\frac{2N_f t^2}{B_k^2} \right).$$

Setting $t = B_k \sqrt{\frac{\log(2/\delta)}{2N_f}}$ and multiplying by α yields a conditional concentration bound for the empirical logit shift around its conditional expectation. Taking expectation (or equivalently using the population view with $\mathbf{z}_l \sim p_f$) gives equation 22, where $\mu_k = \mathbb{E}[X_k]$. The sign agreement condition equation 23 follows directly by requiring the concentration radius to be smaller than $|\mu_k|$. □

C Code

Our code is available at this repository.[1]

D Impact of Embedding Distribution and Sampling Strategy on Unlearning Performance

We investigate the effect of different embedding distributions on class-wise unlearning by sampling embeddings from Gaussian, Laplace, and Uniform distributions. As reported in Table 6 and Table 7, the choice of embedding distribution does impact downstream unlearning performance. Nevertheless, all three distributions achieve competitive results, demonstrating near-complete forgetting alongside strong accuracy on the retain classes. These findings highlight the robustness of our framework to variations in the sampling strategy, as expected from the Proposition 1.

Table 6: Effect of embedding distribution on data-free single-class unlearning performance of some of methods on CIFAR-10, CIFAR-100, and TinyImageNet using ResNet-18 as the backbone architecture. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method.

Method	Embedding Distribution	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	CIFAR-10 $\mathcal{A}_r^t \uparrow$	CIFAR-10 $\mathcal{A}_f^t \downarrow$	CIFAR-10 AUS $\uparrow$	CIFAR-100 $\mathcal{A}_r^t \uparrow$	CIFAR-100 $\mathcal{A}_f^t \downarrow$	CIFAR-100 AUS $\uparrow$	TinyImageNet $\mathcal{A}_r^t \uparrow$	TinyImageNet $\mathcal{A}_f^t \downarrow$	TinyImageNet AUS $\uparrow$
Original	–	–	–	86.58 ± 0.83	86.58 ± 6.67	0.537 ± 0.020	78.16 ± 1.07	78.16 ± 11.15	0.564 ± 0.037	71.30 ± 0.29	71.30 ± 12.46	0.587 ± 0.045
Retrained	–	–	–	86.95 ± 1.17	0.0 ± 0.0	1.004 ± 0.006	77.92 ± 0.80	0.0 ± 0.0	0.998 ± 0.013	63.01 ± 2.76	0.0 ± 0.0	0.917 ± 0.028
RL (Hayase et al., 2020)	Real distribution	✓	✗	87.43 ± 1.16	0.0 ± 0.0	1.008 ± 0.004	78.36 ± 1.05	0.0 ± 0.0	1.002 ± 0.001	71.35 ± 0.32	0.0 ± 0.0	1.001 ± 0.001
	Gaussian	✓	✓	87.25 ± 1.10	0.0 ± 0.0	1.007 ± 0.003	77.98 ± 1.03	0.0 ± 0.0	0.998 ± 0.002	71.10 ± 0.34	0.0 ± 0.0	0.998 ± 0.001
	Laplace	✓	✓	87.25 ± 1.09	0.0 ± 0.0	1.007 ± 0.003	78.00 ± 1.04	0.0 ± 0.0	0.998 ± 0.002	71.18 ± 0.34	0.0 ± 0.0	0.999 ± 0.001
	Uniform	✓	✓	87.30 ± 1.12	0.0 ± 0.0	1.007 ± 0.004	78.01 ± 1.02	0.0 ± 0.0	0.999 ± 0.002	71.19 ± 0.33	0.0 ± 0.0	0.999 ± 0.001
DELETE (Zhou et al., 2025)	Real distribution	✓	✗	87.33 ± 1.12	0.0 ± 0.0	1.008 ± 0.004	78.28 ± 1.06	0.0 ± 0.0	1.001 ± 0.001	71.43 ± 0.30	0.0 ± 0.0	1.001 ± 0.000
	Gaussian	✓	✓	87.35 ± 1.13	0.0 ± 0.0	1.008 ± 0.004	78.25 ± 1.07	0.0 ± 0.1	1.001 ± 0.001	71.36 ± 0.30	0.0 ± 0.0	1.001 ± 0.000
	Laplace	✓	✓	87.35 ± 1.13	0.0 ± 0.0	1.008 ± 0.004	78.25 ± 1.07	0.0 ± 0.0	1.001 ± 0.001	71.36 ± 0.30	0.0 ± 0.0	1.001 ± 0.000
	Uniform	✓	✓	87.33 ± 1.13	0.0 ± 0.0	1.008 ± 0.004	78.25 ± 1.07	0.0 ± 0.0	1.001 ± 0.001	71.35 ± 0.30	0.3 ± 1.2	0.998 ± 0.011
NG+ (Kurmanji et al., 2023)	Real distribution	✗	✗	85.31 ± 9.73	0.0 ± 0.0	0.987 ± 0.095	77.57 ± 6.40	0.0 ± 0.0	0.994 ± 0.062	71.21 ± 0.86	0.0 ± 0.0	0.999 ± 0.008
	Gaussian	✓	✓	87.33 ± 1.12	0.0 ± 0.0	1.007 ± 0.004	78.26 ± 1.04	0.0 ± 0.1	1.001 ± 0.002	71.29 ± 0.36	0.0 ± 0.1	1.000 ± 0.001
	Laplace	✓	✓	87.35 ± 1.13	0.0 ± 0.0	1.008 ± 0.004	78.31 ± 0.99	0.0 ± 0.0	1.001 ± 0.001	71.06 ± 0.46	0.0 ± 0.2	0.997 ± 0.004
	Uniform	✓	✓	87.32 ± 1.12	0.0 ± 0.0	1.007 ± 0.003	78.27 ± 1.05	0.0 ± 0.0	1.001 ± 0.001	71.33 ± 0.33	0.0 ± 0.0	1.000 ± 0.001
SCRUB (Kurmanji et al., 2023)	Real distribution	✗	✗	87.11 ± 1.04	0.0 ± 0.0	1.005 ± 0.003	77.52 ± 1.06	0.0 ± 0.0	0.994 ± 0.002	67.60 ± 1.51	0.0 ± 0.4	0.963 ± 0.014
	Gaussian	✓	✓	87.41 ± 1.16	0.0 ± 0.0	1.008 ± 0.004	78.10 ± 1.06	0.0 ± 0.0	0.999 ± 0.001	71.02 ± 0.42	0.0 ± 0.0	0.997 ± 0.002
	Laplace	✓	✓	87.41 ± 1.15	0.0 ± 0.0	1.008 ± 0.004	78.19 ± 1.00	0.0 ± 0.0	1.000 ± 0.001	71.11 ± 0.37	0.0 ± 0.0	0.998 ± 0.001
	Uniform	✓	✓	87.41 ± 1.15	0.0 ± 0.0	1.008 ± 0.004	78.09 ± 1.05	0.0 ± 0.0	0.999 ± 0.001	70.88 ± 0.35	0.0 ± 0.0	0.996 ± 0.001

Table 7: Effect of embedding distribution on data-free single-class unlearning performance of some of methods on CIFAR-10, CIFAR-100, and TinyImageNet using ViT-B/16 as the backbone architecture. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method.

Method	Embedding Distribution	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	CIFAR-10 $\mathcal{A}_r^t \uparrow$	CIFAR-10 $\mathcal{A}_f^t \downarrow$	CIFAR-10 AUS $\uparrow$	CIFAR-100 $\mathcal{A}_r^t \uparrow$	CIFAR-100 $\mathcal{A}_f^t \downarrow$	CIFAR-100 AUS $\uparrow$	TinyImageNet $\mathcal{A}_r^t \uparrow$	TinyImageNet $\mathcal{A}_f^t \downarrow$	TinyImageNet AUS $\uparrow$
Original	–	–	–	97.69 ± 0.18	97.69 ± 1.30	0.506 ± 0.003	87.22 ± 0.26	87.22 ± 7.83	0.535 ± 0.023	88.20 ± 0.14	88.20 ± 7.29	0.532 ± 0.022
Retrained	–	–	–	98.38 ± 0.21	0.0 ± 0.0	1.007 ± 0.002	88.68 ± 0.25	0.0 ± 0.0	1.014 ± 0.003	89.59 ± 0.13	0.0 ± 0.0	1.014 ± 0.002
RL (Hayase et al., 2020)	Real distribution	✓	✗	97.91 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.31 ± 0.28	0.0 ± 0.0	1.001 ± 0.001	88.24 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	Gaussian	✓	✓	97.92 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.29	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	0.0 ± 0.0	1.000 ± 0.001
	Laplace	✓	✓	97.90 ± 0.23	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.28	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	0.0 ± 0.0	1.000 ± 0.001
	Uniform	✓	✓	97.92 ± 0.24	0.0 ± 0.0	1.002 ± 0.001	87.29 ± 0.28	0.0 ± 0.0	1.001 ± 0.001	88.17 ± 0.14	0.0 ± 0.0	1.000 ± 0.001
DELETE (Zhou et al., 2025)	Real distribution	✓	✗	97.89 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	Gaussian	✓	✓	97.90 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.23 ± 0.14	2.7 ± 8.2	0.979 ± 0.060
	Laplace	✓	✓	97.90 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.24 ± 0.26	0.0 ± 0.0	1.001 ± 0.001	88.24 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
	Uniform	✓	✓	97.89 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.27	0.0 ± 0.0	1.001 ± 0.001	88.24 ± 0.14	0.0 ± 0.0	1.000 ± 0.000
NG+ (Kurmanji et al., 2023)	Real distribution	✗	✗	97.88 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.15 ± 0.29	0.0 ± 0.2	0.999 ± 0.003	87.64 ± 0.27	0.1 ± 0.4	0.993 ± 0.005
	Gaussian	✓	✓	97.91 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.31	0.0 ± 0.0	1.001 ± 0.001	88.25 ± 0.15	0.0 ± 0.0	1.001 ± 0.000
	Laplace	✓	✓	97.91 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.29 ± 0.31	0.0 ± 0.0	1.001 ± 0.001	88.24 ± 0.15	0.0 ± 0.0	1.001 ± 0.000
	Uniform	✓	✓	97.90 ± 0.25	0.0 ± 0.0	1.002 ± 0.001	87.30 ± 0.30	0.0 ± 0.0	1.001 ± 0.001	88.26 ± 0.15	0.0 ± 0.0	1.001 ± 0.000

[1] https://github.com/Yasaman-dt/Source_Free_Class_Unlearning.

E Impact of the Number of Synthetic Embeddings per Class on Unlearning Performance

This part extends the ablation in Section 4 (see Figure 2) by considering additional backbones and datasets such as ResNet-18 on CIFAR-10 (Figure 3), ResNet-18 on TinyImageNet (Figure 4), ViT-B/16 on CIFAR-10 (Figure 5), and ViT-B/16 on CIFAR-100 (Figure 6). For each setting, we vary the number of synthetic embeddings per class and measure retain accuracy $\mathcal{A}_r^t$, forget accuracy $\mathcal{A}_f^t$, and AUS. Across all configurations, the trend is consistent. The pattern is consistent across configurations: increasing the number of synthetic embeddings raises $\mathcal{A}_r^t$ and AUS while reducing $\mathcal{A}_f^t$.

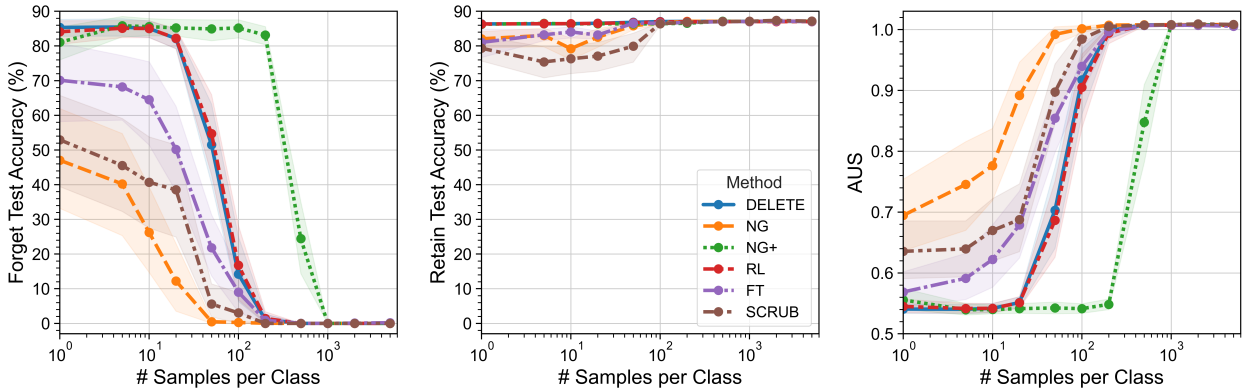

Figure 3: Effect of the number of synthetic embeddings per class on unlearning performance. Results are averaged over three independently trained models, with class-wise unlearning performed separately for each class. Error bars indicate 95% confidence intervals. Experiments use the ResNet-18 architecture on the CIFAR-10 dataset.

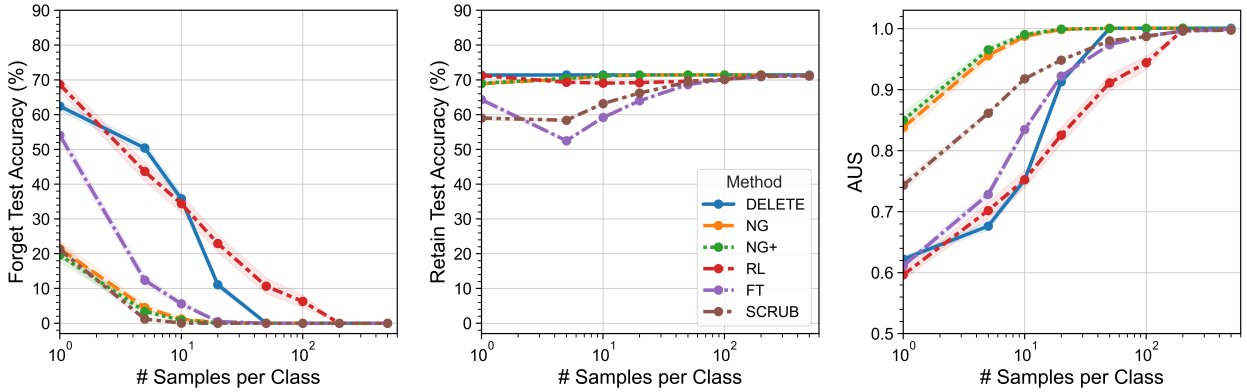

Figure 4: Effect of the number of synthetic embeddings per class on unlearning performance. Results are averaged over three independently trained models, with class-wise unlearning performed separately for each class. Error bars indicate 95% confidence intervals. Experiments use the ResNet-18 architecture on the TinyImageNet dataset.

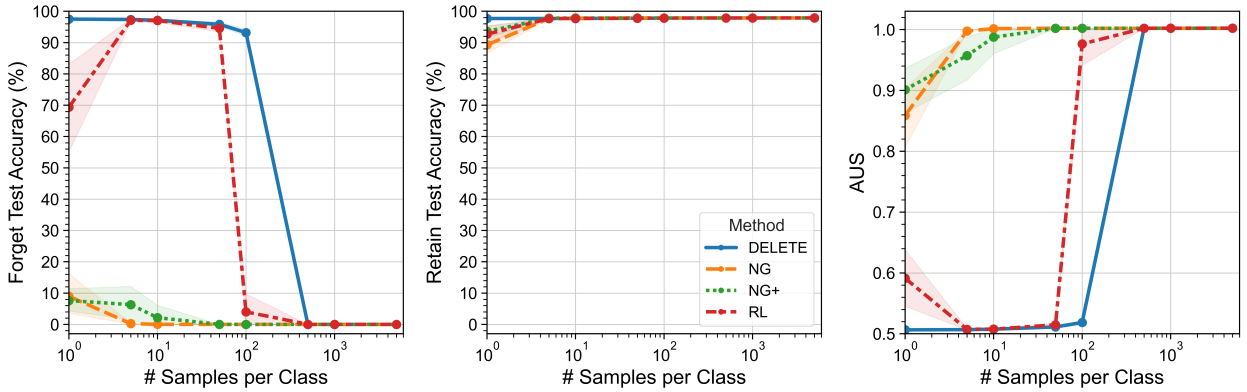

Figure 5: Effect of the number of synthetic embeddings per class on unlearning performance. Results are averaged over three independently trained models, with class-wise unlearning performed separately for each class. Error bars indicate 95% confidence intervals. Experiments use the ViT-B/16 architecture on the CIFAR-10 dataset.

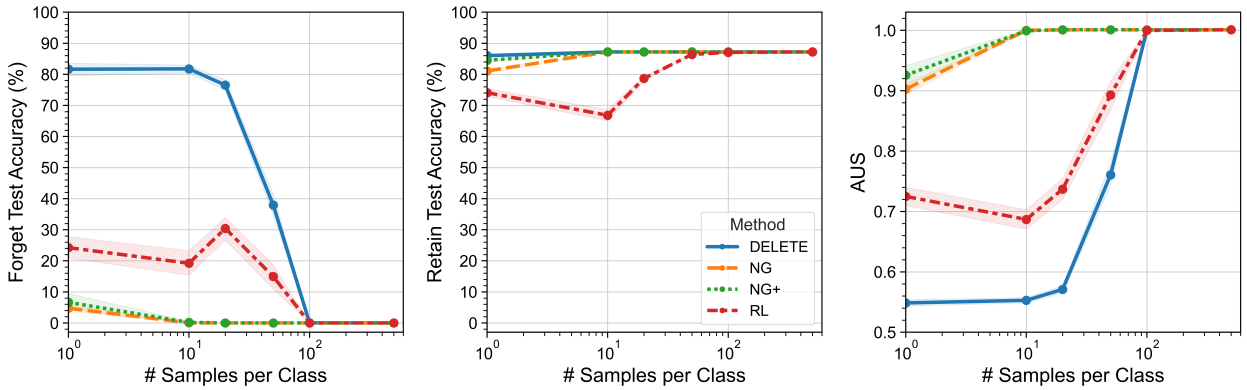

Figure 6: Effect of the number of synthetic embeddings per class on unlearning performance. Results are averaged over three independently trained models, with class-wise unlearning performed separately for each class. Error bars indicate 95% confidence intervals. Experiments use the ViT-B/16 architecture on the CIFAR-100 dataset.

F Per-Class Unlearning Results on CIFAR-10

To supplement the average unlearning performance presented in Table 1 and 2, we provide a detailed per-class evaluation in Table 8 for ResNet-18, Table 9 for ResNet-50, Table 10 for ViT-B/16 and Table 11 for Swin-T. These tables present class-wise unlearning metrics on CIFAR-10 using ResNet-18, ResNet-50, ViT-B/16, and Swin-T backbones, respectively. The results illustrate variability in both unlearning effectiveness and the retain accuracy across target classes, highlighting the impact of semantic complexity and class-specific challenges.

Table 8: Single-class unlearning performance for CIFAR-10 using ResNet-18, averaged over 5 random trials. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method.

Method	Metric	Forget Class 0	1	2	3	4	5	6	7	8	9
Original	$\mathcal{A}_r^t \uparrow$	86.22 ± 0.54	85.91 ± 0.40	86.91 ± 0.47	88.30 ± 0.29	86.50 ± 0.50	87.43 ± 0.42	86.05 ± 0.43	86.29 ± 0.46	86.01 ± 0.38	86.16 ± 0.33
	$\mathcal{A}_f^t \downarrow$	89.8 ± 1.1	92.6 ± 0.7	83.6 ± 0.8	71.0 ± 2.0	87.3 ± 0.9	78.9 ± 0.8	91.4 ± 1.0	89.2 ± 0.7	91.7 ± 0.8	90.3 ± 1.4
	AUS ↑	0.527 ± 0.003	0.519 ± 0.002	0.545 ± 0.002	0.585 ± 0.007	0.534 ± 0.002	0.559 ± 0.002	0.523 ± 0.003	0.529 ± 0.002	0.522 ± 0.002	0.525 ± 0.004
Retrained	$\mathcal{A}_r^t \uparrow$	86.43	86.29	87.38	89.53	86.79	88.66	86.16	86.24	85.92	86.14
	$\mathcal{A}_f^t \downarrow$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AUS ↑	0.997	1.001	1.001	1.009	0.999	1.008	0.997	0.996	0.996	0.997
FT (Golatkar et al., 2020)	$\mathcal{A}_r^t \uparrow$	87.01 ± 0.26	86.58 ± 0.13	87.82 ± 0.17	89.64 ± 0.22	87.38 ± 0.29	88.83 ± 0.27	86.77 ± 0.10	86.85 ± 0.27	86.53 ± 0.25	86.91 ± 0.28
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.008 ± 0.003	1.007 ± 0.003	1.009 ± 0.004	1.013 ± 0.002	1.009 ± 0.003	1.014 ± 0.005	1.007 ± 0.003	1.006 ± 0.003	1.005 ± 0.001	1.007 ± 0.001
	$\mathcal{A}_r^t \uparrow$	86.92 ± 0.43	86.50 ± 0.41	87.79 ± 0.31	89.74 ± 0.30	87.26 ± 0.48	88.78 ± 0.31	86.66 ± 0.35	86.77 ± 0.48	86.40 ± 0.43	86.88 ± 0.44
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.002	1.006 ± 0.001	1.009 ± 0.002	1.014 ± 0.001	1.008 ± 0.000	1.013 ± 0.002	1.006 ± 0.001	1.005 ± 0.001	1.004 ± 0.001	1.007 ± 0.001
NG (Golatkar et al., 2020)	$\mathcal{A}_r^t \uparrow$	86.89 ± 0.54	86.46 ± 0.36	87.71 ± 0.41	89.71 ± 0.34	87.20 ± 0.54	88.68 ± 0.43	86.59 ± 0.41	86.71 ± 0.54	86.37 ± 0.50	86.76 ± 0.40
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.001	1.005 ± 0.001	1.008 ± 0.001	1.014 ± 0.001	1.007 ± 0.001	1.012 ± 0.001	1.005 ± 0.001	1.004 ± 0.002	1.004 ± 0.001	1.006 ± 0.001
	$\mathcal{A}_r^t \uparrow$	86.98 ± 0.46	86.47 ± 0.41	87.79 ± 0.35	89.82 ± 0.42	87.27 ± 0.52	88.89 ± 0.31	86.69 ± 0.33	86.77 ± 0.46	86.46 ± 0.44	86.86 ± 0.41
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.008 ± 0.001	1.006 ± 0.000	1.009 ± 0.002	1.015 ± 0.001	1.008 ± 0.001	1.015 ± 0.001	1.006 ± 0.001	1.005 ± 0.001	1.004 ± 0.001	1.007 ± 0.001
RL (Hayase et al., 2020)	$\mathcal{A}_r^t \uparrow$	86.99 ± 0.51	86.48 ± 0.41	87.83 ± 0.35	89.83 ± 0.45	87.35 ± 0.42	88.99 ± 0.44	86.73 ± 0.33	86.81 ± 0.47	86.43 ± 0.44	86.82 ± 0.44
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.008 ± 0.001	1.006 ± 0.000	1.009 ± 0.001	1.015 ± 0.002	1.008 ± 0.001	1.016 ± 0.002	1.007 ± 0.001	1.005 ± 0.001	1.004 ± 0.001	1.007 ± 0.001
	$\mathcal{A}_r^t \uparrow$	86.93 ± 0.46	86.38 ± 0.38	87.77 ± 0.28	89.65 ± 0.33	87.22 ± 0.48	88.82 ± 0.34	86.64 ± 0.37	86.78 ± 0.45	86.41 ± 0.41	86.72 ± 0.36
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.002	1.005 ± 0.000	1.009 ± 0.002	1.013 ± 0.001	1.007 ± 0.001	1.014 ± 0.002	1.006 ± 0.001	1.005 ± 0.001	1.004 ± 0.000	1.006 ± 0.001
BS (Chen et al., 2023)	$\mathcal{A}_r^t \uparrow$	85.31 ± 1.26	85.83 ± 0.58	86.87 ± 0.57	88.18 ± 0.65	85.98 ± 0.33	87.44 ± 0.94	85.74 ± 0.89	86.08 ± 0.55	85.45 ± 0.52	86.06 ± 0.34
	$\mathcal{A}_f^t \downarrow$	0.5 ± 0.8	0.2 ± 0.2	0.1 ± 0.1	0.0 ± 0.0	0.5 ± 1.0	0.0 ± 0.0	0.0 ± 0.0	0.1 ± 0.2	0.1 ± 0.2	0.0 ± 0.0
	AUS ↑	0.986 ± 0.017	0.997 ± 0.005	0.998 ± 0.005	0.999 ± 0.005	0.990 ± 0.013	1.000 ± 0.007	0.997 ± 0.005	0.997 ± 0.002	0.993 ± 0.006	0.999 ± 0.001
	$\mathcal{A}_r^t \uparrow$	86.83 ± 0.50	86.46 ± 0.37	87.70 ± 0.37	89.81 ± 0.34	87.26 ± 0.50	89.01 ± 0.32	86.62 ± 0.37	86.77 ± 0.48	86.45 ± 0.41	86.81 ± 0.36
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.006 ± 0.001	1.005 ± 0.001	1.008 ± 0.002	1.015 ± 0.001	1.008 ± 0.001	1.016 ± 0.002	1.006 ± 0.002	1.005 ± 0.001	1.004 ± 0.001	1.007 ± 0.001
BE (Chen et al., 2023)	$\mathcal{A}_r^t \uparrow$	82.40 ± 3.28	84.66 ± 1.05	85.63 ± 0.18	85.60 ± 0.64	85.32 ± 0.77	84.51 ± 1.89	84.56 ± 0.64	85.49 ± 0.57	83.78 ± 1.71	85.23 ± 0.61
	$\mathcal{A}_f^t \downarrow$	1.4 ± 1.7	0.0 ± 0.0	0.1 ± 0.2	1.0 ± 2.2	0.5 ± 1.1	0.5 ± 1.0	0.6 ± 0.9	0.0 ± 0.0	0.9 ± 1.7	0.0 ± 0.1
	AUS ↑	0.949 ± 0.040	0.987 ± 0.008	0.986 ± 0.004	0.964 ± 0.020	0.983 ± 0.010	0.966 ± 0.016	0.980 ± 0.005	0.992 ± 0.002	0.969 ± 0.029	0.990 ± 0.004
	$\mathcal{A}_r^t \uparrow$	86.01 ± 0.60	85.92 ± 0.38	86.82 ± 0.47	88.15 ± 0.39	86.50 ± 0.45	87.33 ± 0.54	86.04 ± 0.43	86.25 ± 0.48	85.95 ± 0.30	86.13 ± 0.27
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	0.998 ± 0.002	1.000 ± 0.000	0.999 ± 0.001	0.999 ± 0.001	1.000 ± 0.001	0.999 ± 0.002	1.000 ± 0.000	1.000 ± 0.000	0.999 ± 0.001	1.000 ± 0.001
DELETE (Zhou et al., 2025)	$\mathcal{A}_r^t \uparrow$	86.93 ± 0.44	86.42 ± 0.38	87.74 ± 0.32	89.71 ± 0.38	87.22 ± 0.48	88.80 ± 0.34	86.60 ± 0.32	86.74 ± 0.46	86.41 ± 0.42	86.75 ± 0.39
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.001	1.005 ± 0.000	1.008 ± 0.002	1.014 ± 0.001	1.007 ± 0.001	1.014 ± 0.001	1.006 ± 0.001	1.004 ± 0.001	1.004 ± 0.001	1.006 ± 0.001
	$\mathcal{A}_r^t \uparrow$	86.95 ± 0.46	86.44 ± 0.39	87.76 ± 0.34	89.75 ± 0.40	87.24 ± 0.52	88.83 ± 0.39	86.63 ± 0.34	86.76 ± 0.48	86.43 ± 0.44	86.79 ± 0.40
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.001	1.005 ± 0.000	1.009 ± 0.001	1.014 ± 0.001	1.007 ± 0.001	1.014 ± 0.001	1.006 ± 0.001	1.005 ± 0.001	1.004 ± 0.001	1.006 ± 0.001
NG+ (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	86.31 ± 1.29	86.18 ± 0.52	87.41 ± 0.38	89.23 ± 0.30	86.99 ± 0.50	88.08 ± 0.33	85.58 ± 1.39	83.70 ± 6.71	73.08 ± 29.61	86.60 ± 0.39
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.001 ± 0.008	1.003 ± 0.001	1.005 ± 0.002	1.009 ± 0.003	1.005 ± 0.001	1.006 ± 0.002	0.995 ± 0.011	0.974 ± 0.064	0.871 ± 0.293	1.004 ± 0.002
	$\mathcal{A}_r^t \uparrow$	86.95 ± 0.49	86.45 ± 0.41	87.82 ± 0.34	89.79 ± 0.42	87.27 ± 0.54	88.82 ± 0.32	86.63 ± 0.33	86.77 ± 0.46	86.46 ± 0.48	86.79 ± 0.44
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.001	1.005 ± 0.000	1.009 ± 0.001	1.015 ± 0.001	1.008 ± 0.001	1.014 ± 0.001	1.006 ± 0.001	1.005 ± 0.001	1.005 ± 0.001	1.006 ± 0.001
SCRUB (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	86.48 ± 0.74	86.33 ± 0.44	87.53 ± 0.28	89.32 ± 0.32	86.96 ± 0.42	88.41 ± 0.22	86.44 ± 0.23	86.69 ± 0.38	86.35 ± 0.37	86.61 ± 0.51
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.003 ± 0.002	1.004 ± 0.001	1.006 ± 0.002	1.010 ± 0.002	1.005 ± 0.004	1.010 ± 0.003	1.004 ± 0.002	1.004 ± 0.002	1.003 ± 0.001	1.004 ± 0.003
	$\mathcal{A}_r^t \uparrow$	87.01 ± 0.46	86.54 ± 0.39	87.82 ± 0.30	89.97 ± 0.40	87.28 ± 0.53	88.96 ± 0.35	86.69 ± 0.32	86.82 ± 0.46	86.50 ± 0.45	86.89 ± 0.38
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.008 ± 0.001	1.006 ± 0.000	1.009 ± 0.002	1.017 ± 0.001	1.008 ± 0.001	1.015 ± 0.001	1.006 ± 0.001	1.005 ± 0.001	1.005 ± 0.001	1.007 ± 0.001
SCAR (Bonato et al., 2024)	$\mathcal{A}_r^t \uparrow$	87.03 ± 0.47	86.50 ± 0.37	87.85 ± 0.32	89.87 ± 0.39	87.31 ± 0.52	88.96 ± 0.41	86.73 ± 0.35	86.81 ± 0.46	86.49 ± 0.43	86.88 ± 0.43
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.008 ± 0.001	1.006 ± 0.000	1.009 ± 0.002	1.016 ± 0.001	1.008 ± 0.001	1.015 ± 0.002	1.007 ± 0.001	1.005 ± 0.001	1.005 ± 0.001	1.007 ± 0.001
	$\mathcal{A}_r^t \uparrow$	86.97 ± 0.45	86.46 ± 0.37	87.80 ± 0.31	89.77 ± 0.37	87.27 ± 0.49	88.85 ± 0.34	86.66 ± 0.30	86.78 ± 0.46	86.46 ± 0.42	86.80 ± 0.39
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.007 ± 0.001	1.005 ± 0.001	1.009 ± 0.002	1.015 ± 0.001	1.008 ± 0.001	1.014 ± 0.001	1.006 ± 0.001	1.005 ± 0.001	1.004 ± 0.001	1.006 ± 0.001

Table 9: Single-class unlearning performance for CIFAR-10 using ResNet-50, averaged over 5 random trials. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method.

Method	Metric	Forget Class 0	1	2	3	4	5	6	7	8	9
Original	$\mathcal{A}_r^t \uparrow$	88.18 ± 0.55	87.84 ± 0.51	88.57 ± 0.62	89.58 ± 0.50	88.10 ± 0.69	89.26 ± 0.73	87.77 ± 0.57	87.98 ± 0.71	87.73 ± 0.83	87.74 ± 0.63
	$\mathcal{A}_f^t \downarrow$	89.1 ± 3.1	92.2 ± 2.2	85.6 ± 1.2	76.5 ± 2.4	89.9 ± 0.6	79.4 ± 0.9	92.8 ± 1.3	90.9 ± 0.8	93.2 ± 2.4	93.1 ± 0.8
	AUS $\uparrow$	0.529 ± 0.009	0.520 ± 0.006	0.539 ± 0.004	0.567 ± 0.008	0.527 ± 0.002	0.557 ± 0.003	0.519 ± 0.003	0.524 ± 0.002	0.518 ± 0.006	0.518 ± 0.002
Retrained	$\mathcal{A}_r^t \uparrow$	88.79	88.42	89.40	91.09	89.04	90.66	87.92	88.82	87.92	88.27
	$\mathcal{A}_f^t \downarrow$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AUS $\uparrow$	1.006	1.006	1.008	1.015	1.009	1.014	1.002	1.008	1.002	1.005
FT (Golatkar et al., 2020)	$\mathcal{A}_r^t \uparrow$	89.17 ± 0.37	88.62 ± 0.35	89.73 ± 0.27	91.46 ± 0.49	89.39 ± 0.27	90.64 ± 0.37	88.68 ± 0.42	88.95 ± 0.32	88.55 ± 0.50	88.83 ± 0.34
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.010 ± 0.003	1.008 ± 0.003	1.012 ± 0.004	1.019 ± 0.003	1.013 ± 0.005	1.014 ± 0.005	1.009 ± 0.003	1.010 ± 0.005	1.008 ± 0.004	1.011 ± 0.004
	$\mathcal{A}_r^t \uparrow$	88.80 ± 0.60	88.27 ± 0.57	89.34 ± 0.60	90.92 ± 0.50	88.88 ± 0.57	90.18 ± 0.62	88.31 ± 0.58	88.55 ± 0.64	88.25 ± 0.67	88.30 ± 0.59
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.002	1.004 ± 0.001	1.008 ± 0.001	1.013 ± 0.001	1.008 ± 0.002	1.009 ± 0.001	1.005 ± 0.001	1.006 ± 0.002	1.005 ± 0.002	1.006 ± 0.001
NG (Golatkar et al., 2020)	$\mathcal{A}_r^t \uparrow$	87.20 ± 3.36	88.34 ± 0.44	89.54 ± 0.42	91.24 ± 0.20	89.09 ± 0.40	90.54 ± 0.35	88.53 ± 0.37	88.67 ± 0.41	88.03 ± 0.35	88.43 ± 0.42
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.1 ± 0.1	0.0 ± 0.0
	AUS $\uparrow$	0.990 ± 0.035	1.004 ± 0.000	1.008 ± 0.001	1.015 ± 0.002	1.007 ± 0.001	1.010 ± 0.002	1.006 ± 0.001	1.006 ± 0.002	1.000 ± 0.006	1.005 ± 0.001
	$\mathcal{A}_r^t \uparrow$	88.75 ± 0.62	88.19 ± 0.55	89.35 ± 0.62	91.18 ± 0.58	88.99 ± 0.60	90.41 ± 0.58	88.33 ± 0.60	88.61 ± 0.65	88.29 ± 0.61	88.33 ± 0.46
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.001	1.004 ± 0.001	1.008 ± 0.001	1.016 ± 0.001	1.009 ± 0.001	1.012 ± 0.002	1.006 ± 0.001	1.006 ± 0.003	1.006 ± 0.003	1.006 ± 0.002
RL (Hayase et al., 2020)	$\mathcal{A}_r^t \uparrow$	88.86 ± 0.60	88.25 ± 0.55	89.38 ± 0.60	91.14 ± 0.54	89.02 ± 0.58	90.30 ± 0.59	88.39 ± 0.59	88.59 ± 0.63	88.30 ± 0.66	88.40 ± 0.50
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.007 ± 0.002	1.004 ± 0.001	1.008 ± 0.001	1.016 ± 0.001	1.009 ± 0.002	1.010 ± 0.002	1.006 ± 0.001	1.006 ± 0.003	1.006 ± 0.002	1.007 ± 0.001
	$\mathcal{A}_r^t \uparrow$	88.79 ± 0.60	88.15 ± 0.57	89.28 ± 0.62	90.93 ± 0.46	88.81 ± 0.57	90.14 ± 0.63	88.21 ± 0.50	88.43 ± 0.63	88.20 ± 0.67	88.30 ± 0.56
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.001	1.003 ± 0.001	1.007 ± 0.001	1.013 ± 0.002	1.007 ± 0.001	1.009 ± 0.001	1.004 ± 0.001	1.004 ± 0.002	1.005 ± 0.002	1.006 ± 0.001
BS (Chen et al., 2023)	$\mathcal{A}_r^t \uparrow$	88.15 ± 0.70	87.73 ± 0.51	88.18 ± 0.70	87.62 ± 1.28	87.21 ± 0.91	87.41 ± 2.87	87.32 ± 1.15	87.98 ± 0.89	87.71 ± 0.80	87.70 ± 0.67
	$\mathcal{A}_f^t \downarrow$	3.8 ± 5.2	0.0 ± 0.0	0.3 ± 0.6	0.0 ± 0.0	0.0 ± 0.0	3.7 ± 8.2	0.7 ± 1.3	0.6 ± 1.1	0.4 ± 0.5	0.6 ± 1.1
	AUS $\uparrow$	0.965 ± 0.045	0.999 ± 0.002	0.993 ± 0.008	0.980 ± 0.012	0.991 ± 0.008	0.950 ± 0.066	0.989 ± 0.014	0.995 ± 0.012	0.996 ± 0.004	0.993 ± 0.009
	$\mathcal{A}_r^t \uparrow$	88.68 ± 0.58	88.44 ± 0.46	89.48 ± 0.33	91.14 ± 0.58	89.15 ± 0.29	90.58 ± 0.32	88.66 ± 0.37	88.78 ± 0.42	88.61 ± 0.06	88.71 ± 0.39
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.005 ± 0.001	1.005 ± 0.001	1.007 ± 0.002	1.014 ± 0.002	1.008 ± 0.002	1.011 ± 0.003	1.007 ± 0.003	1.005 ± 0.002	1.003 ± 0.001	1.006 ± 0.001
BE (Chen et al., 2023)	$\mathcal{A}_r^t \uparrow$	87.68 ± 0.52	87.24 ± 0.53	88.27 ± 0.64	86.68 ± 4.42	87.86 ± 0.57	89.21 ± 0.68	86.89 ± 0.69	87.39 ± 0.78	87.14 ± 0.80	86.79 ± 0.60
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	1.2 ± 1.6	0.5 ± 1.1	10.1 ± 14.9	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	0.995 ± 0.002	0.994 ± 0.004	0.997 ± 0.001	0.959 ± 0.033	0.993 ± 0.010	0.919 ± 0.107	0.991 ± 0.003	0.994 ± 0.001	0.994 ± 0.002	0.990 ± 0.002
	$\mathcal{A}_r^t \uparrow$	88.14 ± 0.58	87.81 ± 0.50	88.51 ± 0.61	89.47 ± 0.55	88.08 ± 0.70	89.19 ± 0.76	87.70 ± 0.61	87.97 ± 0.71	87.67 ± 0.85	87.69 ± 0.65
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.000 ± 0.000	1.000 ± 0.000	0.999 ± 0.000	0.999 ± 0.001	1.000 ± 0.000	0.999 ± 0.000	0.999 ± 0.001	1.000 ± 0.000	0.999 ± 0.001	1.000 ± 0.000
DELETE (Zhou et al., 2025)	$\mathcal{A}_r^t \uparrow$	88.78 ± 0.60	88.17 ± 0.56	89.33 ± 0.62	91.04 ± 0.51	88.90 ± 0.59	90.25 ± 0.59	88.33 ± 0.53	88.52 ± 0.63	88.24 ± 0.66	88.32 ± 0.50
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.001	1.003 ± 0.001	1.008 ± 0.001	1.014 ± 0.002	1.008 ± 0.001	1.010 ± 0.001	1.006 ± 0.001	1.005 ± 0.002	1.005 ± 0.002	1.006 ± 0.001
	$\mathcal{A}_r^t \uparrow$	88.76 ± 0.62	88.16 ± 0.55	89.33 ± 0.63	91.04 ± 0.51	88.91 ± 0.60	90.26 ± 0.61	88.30 ± 0.62	88.50 ± 0.66	88.23 ± 0.64	88.32 ± 0.52
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.001	1.003 ± 0.001	1.008 ± 0.001	1.014 ± 0.002	1.008 ± 0.001	1.010 ± 0.001	1.005 ± 0.001	1.005 ± 0.002	1.005 ± 0.002	1.006 ± 0.001
NG+ (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	88.91 ± 0.60	88.47 ± 0.57	89.54 ± 0.54	90.96 ± 0.49	89.09 ± 0.57	90.33 ± 0.62	88.43 ± 0.71	88.64 ± 0.62	88.24 ± 0.51	88.59 ± 0.48
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.007 ± 0.003	1.006 ± 0.003	1.010 ± 0.003	1.014 ± 0.003	1.010 ± 0.003	1.011 ± 0.003	1.007 ± 0.003	1.007 ± 0.003	1.005 ± 0.004	1.009 ± 0.003
	$\mathcal{A}_r^t \uparrow$	88.77 ± 0.66	88.22 ± 0.57	89.35 ± 0.64	90.96 ± 0.52	88.97 ± 0.62	90.24 ± 0.63	88.32 ± 0.62	88.51 ± 0.64	88.25 ± 0.65	88.33 ± 0.55
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.002	1.004 ± 0.001	1.008 ± 0.001	1.014 ± 0.001	1.009 ± 0.001	1.010 ± 0.002	1.006 ± 0.001	1.005 ± 0.001	1.005 ± 0.002	1.006 ± 0.001
SCRUB (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	88.87 ± 0.57	88.33 ± 0.47	89.33 ± 0.44	90.70 ± 0.35	88.92 ± 0.53	90.16 ± 0.62	88.28 ± 0.64	88.47 ± 0.51	88.16 ± 0.64	88.39 ± 0.59
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.007 ± 0.002	1.006 ± 0.003	1.009 ± 0.003	1.013 ± 0.002	1.010 ± 0.003	1.011 ± 0.003	1.007 ± 0.003	1.006 ± 0.003	1.006 ± 0.004	1.008 ± 0.002
	$\mathcal{A}_r^t \uparrow$	88.82 ± 0.56	88.32 ± 0.55	89.37 ± 0.65	91.25 ± 0.59	89.02 ± 0.57	90.46 ± 0.67	88.37 ± 0.60	88.66 ± 0.64	88.32 ± 0.59	88.46 ± 0.53
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.002	1.005 ± 0.001	1.008 ± 0.001	1.017 ± 0.001	1.009 ± 0.002	1.012 ± 0.002	1.006 ± 0.002	1.007 ± 0.003	1.006 ± 0.003	1.007 ± 0.001
SCAR (Bonato et al., 2024)	$\mathcal{A}_r^t \uparrow$	88.87 ± 0.58	88.31 ± 0.54	89.39 ± 0.55	91.25 ± 0.52	89.02 ± 0.61	90.43 ± 0.50	88.37 ± 0.58	88.64 ± 0.66	88.37 ± 0.60	88.44 ± 0.48
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.007 ± 0.002	1.005 ± 0.001	1.008 ± 0.001	1.017 ± 0.002	1.009 ± 0.001	1.012 ± 0.003	1.006 ± 0.001	1.007 ± 0.003	1.006 ± 0.002	1.007 ± 0.002
	$\mathcal{A}_r^t \uparrow$	88.81 ± 0.63	88.21 ± 0.56	89.36 ± 0.64	91.06 ± 0.51	88.95 ± 0.60	90.29 ± 0.63	88.34 ± 0.61	88.56 ± 0.65	88.26 ± 0.64	88.36 ± 0.54
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS $\uparrow$	1.006 ± 0.001	1.004 ± 0.001	1.008 ± 0.001	1.015 ± 0.002	1.008 ± 0.001	1.010 ± 0.001	1.006 ± 0.001	1.006 ± 0.002	1.005 ± 0.002	1.006 ± 0.001

Table 10: Single-class unlearning performance for CIFAR-10 using ViT-B/16, averaged over 5 random trials. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method.

Method	Metric	Forget Class 0	1	2	3	4	5	6	7	8	9
Original	$\mathcal{A}_r^t \uparrow$	97.65 ± 0.07	97.60 ± 0.11	97.71 ± 0.13	97.98 ± 0.11	97.68 ± 0.18	97.88 ± 0.11	97.55 ± 0.13	97.63 ± 0.08	97.55 ± 0.13	97.65 ± 0.15
	$\mathcal{A}_f^t \downarrow$	98.0 ± 0.6	98.5 ± 0.5	97.5 ± 0.2	95.1 ± 0.8	97.8 ± 0.7	95.9 ± 0.2	98.9 ± 0.3	98.2 ± 0.9	98.9 ± 0.2	98.0 ± 0.4
	AUS ↑	0.505 ± 0.002	0.504 ± 0.001	0.506 ± 0.001	0.513 ± 0.002	0.506 ± 0.002	0.510 ± 0.000	0.503 ± 0.001	0.505 ± 0.002	0.503 ± 0.000	0.505 ± 0.001
Retrained	$\mathcal{A}_r^t \uparrow$	98.39	98.38	98.21	98.86	98.38	98.67	98.17	98.28	98.20	98.31
	$\mathcal{A}_f^t \downarrow$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AUS ↑	1.007	1.008	1.005	1.009	1.007	1.008	1.006	1.006	1.006	1.007
NG (Golatkar et al., 2020)	$\mathcal{A}_r^t \uparrow$	97.80 ± 0.10	97.78 ± 0.10	97.85 ± 0.12	98.34 ± 0.09	97.92 ± 0.12	98.27 ± 0.14	97.64 ± 0.14	97.77 ± 0.07	97.71 ± 0.13	97.82 ± 0.12
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.001 ± 0.000	1.004 ± 0.001	1.002 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.001 ± 0.001	1.002 ± 0.000	1.002 ± 0.001
	$\mathcal{A}_r^t \uparrow$	97.81 ± 0.10	97.78 ± 0.10	97.86 ± 0.12	98.34 ± 0.09	97.93 ± 0.12	98.28 ± 0.13	97.65 ± 0.14	97.77 ± 0.06	97.71 ± 0.14	97.82 ± 0.12
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.002 ± 0.000	1.004 ± 0.001	1.003 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.001 ± 0.001	1.002 ± 0.000	1.002 ± 0.001
RL (Hayase et al., 2020)	$\mathcal{A}_r^t \uparrow$	97.81 ± 0.10	97.79 ± 0.10	97.85 ± 0.12	98.37 ± 0.10	97.94 ± 0.12	98.28 ± 0.12	97.68 ± 0.13	97.78 ± 0.07	97.72 ± 0.14	97.83 ± 0.12
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.001 ± 0.000	1.004 ± 0.000	1.003 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.001 ± 0.001	1.002 ± 0.000	1.002 ± 0.001
	$\mathcal{A}_r^t \uparrow$	97.85 ± 0.07	97.84 ± 0.12	97.90 ± 0.13	98.38 ± 0.11	97.96 ± 0.13	98.30 ± 0.13	97.70 ± 0.12	97.83 ± 0.06	97.75 ± 0.15	97.83 ± 0.12
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.001	1.002 ± 0.000	1.004 ± 0.000	1.003 ± 0.001	1.004 ± 0.000	1.001 ± 0.000	1.002 ± 0.001	1.002 ± 0.000	1.002 ± 0.000
BS (Chen et al., 2023)	$\mathcal{A}_r^t \uparrow$	97.67 ± 0.12	97.75 ± 0.08	97.79 ± 0.19	98.13 ± 0.24	97.74 ± 0.19	97.97 ± 0.19	97.61 ± 0.15	97.68 ± 0.07	97.60 ± 0.14	97.72 ± 0.22
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.000 ± 0.001	1.002 ± 0.001	1.001 ± 0.001	1.002 ± 0.002	1.001 ± 0.000	1.001 ± 0.001	1.001 ± 0.000	1.000 ± 0.000	1.000 ± 0.001	1.001 ± 0.001
	$\mathcal{A}_r^t \uparrow$	97.80 ± 0.10	97.76 ± 0.07	97.86 ± 0.11	98.29 ± 0.08	97.92 ± 0.13	98.25 ± 0.14	97.67 ± 0.14	97.79 ± 0.05	97.69 ± 0.17	97.82 ± 0.11
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.001	1.001 ± 0.000	1.003 ± 0.001	1.002 ± 0.001	1.004 ± 0.000	1.001 ± 0.000	1.002 ± 0.001	1.001 ± 0.000	1.002 ± 0.001
DELETE (Zhou et al., 2025)	$\mathcal{A}_r^t \uparrow$	97.81 ± 0.10	97.78 ± 0.09	97.85 ± 0.12	98.34 ± 0.10	97.93 ± 0.12	98.28 ± 0.13	97.64 ± 0.13	97.77 ± 0.07	97.71 ± 0.13	97.82 ± 0.12
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.001 ± 0.000	1.004 ± 0.001	1.003 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.001 ± 0.001	1.002 ± 0.000	1.002 ± 0.001
	$\mathcal{A}_r^t \uparrow$	97.81 ± 0.09	97.79 ± 0.10	97.87 ± 0.13	98.35 ± 0.10	97.95 ± 0.12	98.29 ± 0.13	97.66 ± 0.15	97.79 ± 0.07	97.72 ± 0.14	97.83 ± 0.13
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.002 ± 0.000	1.004 ± 0.000	1.003 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.002 ± 0.001	1.002 ± 0.000	1.002 ± 0.000
NG+ (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	97.79 ± 0.10	97.77 ± 0.13	97.83 ± 0.13	98.34 ± 0.09	97.91 ± 0.13	98.26 ± 0.15	97.64 ± 0.14	97.75 ± 0.08	97.69 ± 0.14	97.81 ± 0.13
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.001 ± 0.000	1.002 ± 0.000	1.001 ± 0.000	1.004 ± 0.001	1.002 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.001 ± 0.001	1.001 ± 0.000	1.002 ± 0.001
	$\mathcal{A}_r^t \uparrow$	97.82 ± 0.10	97.80 ± 0.10	97.88 ± 0.13	98.37 ± 0.10	97.96 ± 0.12	98.31 ± 0.14	97.67 ± 0.14	97.79 ± 0.07	97.74 ± 0.14	97.85 ± 0.13
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.002 ± 0.000	1.004 ± 0.000	1.003 ± 0.001	1.004 ± 0.001	1.001 ± 0.000	1.002 ± 0.001	1.002 ± 0.000	1.002 ± 0.000

Table 11: Single-class unlearning performance for CIFAR-10 using Swin-T, averaged over 5 random trials. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method.

Method	Metric	Forget Class 0	1	2	3	4	5	6	7	8	9
Original	$\mathcal{A}_r^t \uparrow$	97.58 ± 0.08	97.65 ± 0.05	97.78 ± 0.08	97.96 ± 0.15	97.74 ± 0.03	98.03 ± 0.10	97.55 ± 0.05	97.63 ± 0.08	97.60 ± 0.07	97.74 ± 0.09
	$\mathcal{A}_f^t \downarrow$	99.0 ± 0.3	98.4 ± 0.5	97.3 ± 0.6	95.6 ± 0.9	97.6 ± 0.7	95.0 ± 0.9	99.3 ± 0.3	98.6 ± 0.3	98.8 ± 0.1	97.6 ± 0.3
	AUS ↑	0.502 ± 0.001	0.504 ± 0.001	0.507 ± 0.002	0.511 ± 0.002	0.506 ± 0.002	0.513 ± 0.002	0.502 ± 0.001	0.504 ± 0.001	0.503 ± 0.000	0.506 ± 0.001
Retrained	$\mathcal{A}_r^t \uparrow$	98.22	98.30	98.31	98.80	98.30	98.73	98.14	98.14	98.17	98.43
	$\mathcal{A}_f^t \downarrow$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AUS ↑	1.006	1.006	1.005	1.008	1.006	1.007	1.006	1.005	1.006	1.007
NG (Golatkar et al., 2020)	$\mathcal{A}_r^t \uparrow$	97.73 ± 0.05	97.86 ± 0.05	97.88 ± 0.07	98.46 ± 0.10	97.91 ± 0.04	98.37 ± 0.12	97.65 ± 0.06	97.76 ± 0.06	97.74 ± 0.07	97.88 ± 0.08
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.002 ± 0.000	1.002 ± 0.000	1.001 ± 0.000	1.005 ± 0.001	1.002 ± 0.000	1.003 ± 0.001	1.001 ± 0.000	1.001 ± 0.000	1.001 ± 0.000	1.001 ± 0.000
	$\mathcal{A}_r^t \uparrow$	97.71 ± 0.10	97.65 ± 0.28	97.74 ± 0.08	97.31 ± 2.61	97.90 ± 0.07	97.81 ± 0.60	97.28 ± 0.72	97.66 ± 0.19	97.71 ± 0.08	97.71 ± 0.11
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	1.1 ± 1.6	0.3 ± 0.3	0.9 ± 1.9	0.1 ± 0.2	1.1 ± 1.6	0.2 ± 0.3	0.2 ± 0.2	0.1 ± 0.1	0.7 ± 0.6
	AUS ↑	1.001 ± 0.001	0.990 ± 0.017	0.997 ± 0.003	0.985 ± 0.044	1.001 ± 0.002	0.987 ± 0.021	0.995 ± 0.010	0.998 ± 0.004	1.001 ± 0.001	0.993 ± 0.007
NG+ (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	97.67 ± 0.07	97.67 ± 0.06	97.85 ± 0.06	98.32 ± 0.13	97.81 ± 0.02	98.32 ± 0.14	97.59 ± 0.05	97.66 ± 0.06	97.64 ± 0.04	97.76 ± 0.07
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.001 ± 0.000	1.000 ± 0.000	1.001 ± 0.000	1.004 ± 0.001	1.001 ± 0.000	1.003 ± 0.001	1.000 ± 0.000	1.000 ± 0.001	1.000 ± 0.001	1.000 ± 0.000
	$\mathcal{A}_r^t \uparrow$	97.45 ± 0.30	90.38 ± 11.22	92.22 ± 6.15	95.45 ± 3.03	97.15 ± 0.70	95.16 ± 0.77	85.57 ± 16.19	94.36 ± 5.09	94.61 ± 4.95	92.80 ± 4.31
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.1	2.1 ± 1.5	1.2 ± 0.7	0.9 ± 1.4	0.1 ± 0.2	3.0 ± 0.5	1.2 ± 1.8	0.8 ± 0.5	0.5 ± 0.5	2.2 ± 1.6
	AUS ↑	0.998 ± 0.003	0.908 ± 0.113	0.933 ± 0.060	0.967 ± 0.041	0.993 ± 0.007	0.943 ± 0.009	0.872 ± 0.172	0.960 ± 0.052	0.966 ± 0.053	0.931 ± 0.054
SCRUB (Kurmanji et al., 2023)	$\mathcal{A}_r^t \uparrow$	97.63 ± 0.09	97.68 ± 0.06	97.84 ± 0.07	98.38 ± 0.09	97.87 ± 0.03	98.22 ± 0.10	97.65 ± 0.06	97.73 ± 0.04	97.69 ± 0.04	97.79 ± 0.04
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.000 ± 0.001	1.000 ± 0.001	1.001 ± 0.000	1.004 ± 0.001	1.001 ± 0.000	1.002 ± 0.001	1.001 ± 0.000	1.001 ± 0.001	1.001 ± 0.001	1.001 ± 0.001
	$\mathcal{A}_r^t \uparrow$	97.59 ± 0.07	97.49 ± 0.35	97.57 ± 0.18	97.93 ± 0.38	96.70 ± 2.09	97.94 ± 0.37	97.23 ± 0.53	96.36 ± 2.62	97.38 ± 0.29	97.71 ± 0.10
	$\mathcal{A}_f^t \downarrow$	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	AUS ↑	1.000 ± 0.001	0.998 ± 0.003	0.998 ± 0.002	1.000 ± 0.005	0.990 ± 0.021	0.999 ± 0.003	0.997 ± 0.005	0.987 ± 0.026	0.998 ± 0.003	1.000 ± 0.001

G Membership Inference Attacks (MIAs)

In the machine unlearning literature, several variants of Membership Inference Attacks (MIAs) have been proposed to assess whether a model has truly forgotten the requested data. In this work, we adopt two commonly used MIAs. MIA_I(MIA-Efficacy) (Song & Mittal, 2021; Jia et al., 2023; Fan et al., 2023; Zhang et al., 2025; Chen et al., 2024; Sepahvand et al., 2025) trains an attack classifier to distinguish training samples from test samples using the model's prediction confidence. This attacker is then applied to the forget set at inference time and the metric reports the proportion of forget samples classified as "test". Higher values indicate stronger forgetting; a score of 100% means the attacker labels all forget samples as "not seen," suggesting the model has fully removed the forget set from its training signal. MIA_II (Kurmanji et al., 2023; Sepahvand et al., 2025) trains a binary attacker to distinguish forget samples from test samples using the per-sample loss produced by the unlearned model. The attacker is evaluated on a held-out set of losses from both groups. Ideally, effective unlearning method should lead the attacker to an accuracy of 50%, meaning it cannot distinguish between the forget set and test set, which indicates successful unlearning. Therefore, for this MIA, accuracy values closer to 50% indicate stronger unlearning. In Table 12, MIA_I indicates strong forgetting, while MIA_II remains moderately above chance for ResNet-18 and ResNet50 and is closest to chance for ViT-B/16 and Swin-T backbones.

Table 12: MIA performance of single-class unlearning on CIFAR10 using ResNet-18, ResNet-50, ViT-B/16 and Swin-T, averaged over 5 random trials. Rows highlighted in gray represent our results using synthetic embeddings, while the corresponding non-shaded rows use original embeddings with the same method. Columns $\mathcal{D}_r$-free and $\mathcal{D}_f$-free indicate whether the method operates without access to the retain or forget set, respectively, with (✓) denoting true and (✗) denoting false.

Method	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	**ResNet-18**		**ResNet-50**		**ViT-B/16**		**Swin-T**	
			$\text{MIA}_I \uparrow$	$\text{MIA}_{II} \downarrow$	$\text{MIA}_I \uparrow$	$\text{MIA}_{II} \downarrow$	$\text{MIA}_I \uparrow$	$\text{MIA}_{II} \downarrow$	$\text{MIA}_I \uparrow$	$\text{MIA}_{II} \downarrow$
Original	–	–	0.28 ± 0.27	60.12 ± 4.27	0.45 ± 0.79	57.98 ± 3.07	0.04 ± 0.08	52.90 ± 2.03	0.00 ± 0.00	52.51 ± 2.14
Retrained	–	–	100.00 ± 0.00	54.16 ± 4.83	100.00 ± 0.00	55.72 ± 4.31	100.00 ± 0.00	52.86 ± 2.71	100.00 ± 0.00	51.01 ± 2.38
FT (Golatkar et al., 2020)	✗	✓	100.00 ± 0.00	55.30 ± 4.24	100.00 ± 0.00	58.89 ± 3.30	-	-	-	-
	✓	✓	100.00 ± 0.00	60.58 ± 4.57	100.00 ± 0.00	59.11 ± 3.58	-	-	-	-
NG (Golatkar et al., 2020)	✓	✗	100.00 ± 0.00	58.85 ± 5.60	100.00 ± 0.00	53.92 ± 4.40	100.00 ± 0.00	51.50 ± 2.03	100.00 ± 0.00	51.52 ± 1.58
	✓	✓	100.00 ± 0.00	59.13 ± 4.77	100.00 ± 0.00	57.04 ± 3.85	100.00 ± 0.00	52.08 ± 1.88	99.52 ± 1.30	50.58 ± 2.09
RL (Hayase et al., 2020)	✓	✗	100.00 ± 0.00	58.45 ± 5.60	99.99 ± 0.03	54.33 ± 4.87	100.00 ± 0.00	51.40 ± 2.40	-	-
	✓	✓	100.00 ± 0.00	58.12 ± 4.18	100.00 ± 0.00	57.03 ± 4.19	99.99 ± 0.00	52.24 ± 2.13	-	-
BS (Chen et al., 2023)	✓	✗	99.66 ± 1.16	52.91 ± 3.16	99.02 ± 6.77	55.94 ± 3.09	100.00 ± 0.00	51.35 ± 1.95	-	-
	✓	✓	100.00 ± 0.00	58.31 ± 4.12	100.00 ± 0.00	58.06 ± 3.20	100.00 ± 0.00	51.52 ± 1.75	-	-
DELETE (Zhou et al., 2025)	✓	✗	100.00 ± 0.00	58.26 ± 5.63	100.00 ± 0.00	54.99 ± 5.46	100.00 ± 0.00	51.30 ± 2.19	-	-
	✓	✓	100.00 ± 0.00	58.25 ± 4.74	100.00 ± 0.00	56.69 ± 4.34	100.00 ± 0.00	51.25 ± 2.05	-	-
NG+ (Kurmanji et al., 2023)	✗	✗	100.00 ± 0.00	55.65 ± 5.10	100.00 ± 0.00	53.08 ± 3.38	100.00 ± 0.00	51.76 ± 1.96	100.00 ± 0.00	51.18 ± 1.58
	✓	✓	100.00 ± 0.00	57.84 ± 4.51	100.00 ± 0.00	56.64 ± 3.90	100.00 ± 0.00	51.71 ± 1.80	92.96 ± 9.98	50.69 ± 2.08
SCRUB (Kurmanji et al., 2023)	✗	✗	99.99 ± 0.01	54.40 ± 4.25	100.00 ± 0.00	56.94 ± 4.56	-	-	99.99 ± 0.00	50.47 ± 1.68
	✓	✓	100.00 ± 0.00	56.25 ± 5.26	100.00 ± 0.00	55.54 ± 3.74	-	-	100.00 ± 0.00	50.47 ± 1.85
SCAR (Bonato et al., 2024)	✗	✗	100.00 ± 0.00	53.06 ± 3.25	97.99 ± 14.14	55.99 ± 4.34	-	-	-	-
	✓	✓	100.00 ± 0.00	58.27 ± 4.74	100.00 ± 0.00	58.06 ± 3.66	-	-	-	-

H Verification of Assumptions 3 and 5

Assumption 3 is formulated as a condition for the theoretical result, rather than as a universal property of all unlearning methods. Several methods considered in this work optimize a composite objective that includes a forget term on forget samples and, in some cases, an additional retain or regularization term on retain samples. Here, Assumption 3 and 5 is analyzed only for the forget component $\mathcal{L}_f$, since the monotonicity condition is meant to characterize the suppressive action of the forgetting objective on synthetic forget embeddings. We empirically verify Assumptions 3 and 5 on sampled pseudo-forget embeddings. All empirical checks are performed in the head-only setting, where synthetic embeddings are used and only the classifier head h is updated. Accordingly, Assumptions 3 and 5 are evaluated at the embedding level and only with respect to the forget component $\mathcal{L}_f$. In particular, FT optimizes only a retain objective, while SCAR also reduces to a retain-only objective in the head-only setting considered here; therefore, since neither method includes an explicit forget component, Assumptions 3 and 5 are not evaluated for them.

Checking Assumption 3. We sample synthetic embeddings $\mathbf{z} \sim p_{\mathbf{z}}$, pseudo-label them, and keep those assigned to the forget class to form the synthetic forget set $\mathcal{E}_f = \{\mathbf{z} : \hat{y}(\mathbf{z}) = c_f\}$, with $c_f = 0$. Let $\mathcal{E}_f = \{\mathbf{z}_i\}_{i=1}^{N_f}$ denote the resulting set of N_f pseudo-forget embeddings. For each $\mathbf{z}_i \in \mathcal{E}_f$, we compute the logits $h(\mathbf{z}_i) \in \mathbb{R}^C$ and the logit-gradient vector $s(\mathbf{z}_i) = \nabla_h \mathcal{L}_f(\mathbf{z}_i)$, where $s_{c_f}(\mathbf{z}_i) = \frac{\partial \mathcal{L}_f(\mathbf{z}_i)}{\partial h_{c_f}}$ denotes the forget-class component. Since Assumption 3 is stated in expectation, we assess it through its empirical counterpart on the sampled pseudo-forget set. Specifically, Table 13 reports $\frac{1}{N_f}\sum_{i=1}^{N_f} s_{c_f}(\mathbf{z}_i)$, which estimates $\mathbb{E}_{\mathbf{z}\sim p_f}[s_{c_f}(\mathbf{z})]$. A positive value indicates that the forget loss induces, on average, a positive forget-class logit gradient on the synthetic forget distribution, consistent with Assumption 3.

Checking Assumption 5. We empirically verify Assumption 5 using samples from p_f. For a sampled embedding $\mathbf{z} \in \mathbb{R}^D$, let $s(\mathbf{z}) = \nabla_h \mathcal{L}_f(\mathbf{z}) \in \mathbb{R}^C$ be the logit-gradient of the forget loss. Independence of $\mathbf{z}$ and $\mathbf{z}'$ yields the factorization

$$\mathbb{E}_{\mathbf{z},\mathbf{z}'\sim p_f}\left[s_k(\mathbf{z})\,\mathbf{z}^\top \mathbf{z}'\right] = \left(\mathbb{E}_{\mathbf{z}\sim p_f}[s_k(\mathbf{z})\,\mathbf{z}]\right)^\top \left(\mathbb{E}_{\mathbf{z}'\sim p_f}[\mathbf{z}']\right), \tag{25}$$

which motivates estimating the alignment score using two empirical means:

$$\mu \;=\; \frac{1}{N}\sum_{i=1}^{N} \mathbf{z}_i, \qquad v_k \;=\; \frac{1}{N}\sum_{i=1}^{N} s_k(\mathbf{z}_i)\,\mathbf{z}_i, \qquad A_k \;=\; v_k^\top \mu. \tag{26}$$

Assumption 5 predicts the sign pattern: the forget-class score should be positive ($A_{c_f} > 0$), while retain-class scores should be negative ($A_k < 0$ for all $k \in \mathcal{Y}_r$). Accordingly, Table 13 reports A_{c_f} as the forget-class alignment score and summarizes retain-class alignments by the range $\left[\min_{k\in\mathcal{Y}_r} A_k,\ \max_{k\in\mathcal{Y}_r} A_k\right]$.

Table 13: Empirical checks of Assumptions 3 and 5 on N=5000 synthetic forget embeddings with forget class c_f=0 for CIFAR-10 using ResNet-18 and ViT-B/16 as the base architecture.

Backbone	Method	**Assumption 3**	**Assumption 5**	
		$\frac{1}{N_f}\sum_{i=1}^{N_f} s_{c_f}(\mathbf{z}_i)$	A_{c_f}	A_k $(k \in \mathcal{Y}_r)$ [min, max]
ResNet-18	NG (Golatkar et al., 2020)	0.753	2.026	[−0.250, −0.212]
	NG+ (Kurmanji et al., 2023)	0.753	2.026	[−0.250, −0.212]
	RL (Hayase et al., 2020)	0.247	0.743	[−0.101, −0.057]
	BS (Chen et al., 2023)	0.247	0.743	[−0.176, −0.052]
	DELETE (Zhou et al., 2025)	0.087	0.024	[−0.027, −0.026]
	SCRUB (Kurmanji et al., 2023)	0.220	0.677	[−0.111, −0.031]
ViT-B/16	NG (Golatkar et al., 2020)	0.829	2.255	[−0.257, −0.246]
	NG+ (Kurmanji et al., 2023)	0.829	2.255	[−0.257, −0.246]
	RL (Hayase et al., 2020)	0.171	0.495	[−0.085, −0.045]
	BS (Chen et al., 2023)	0.171	0.495	[−0.085, −0.042]
	DELETE (Zhou et al., 2025)	0.077	0.210	[−0.024, −0.022]
	SCRUB (Kurmanji et al., 2023)	0.168	0.489	[−0.085, −0.033]

I Average Raw Draws per Accepted Embeddings

This appendix quantifies the sampling overhead incurred when constructing a synthetic embedding set with *balanced per-class pseudo-label quotas*. We repeatedly draw synthetic embeddings $\mathbf{z} \sim p_{\mathbf{z}}$ in an intermediate embedding space and assign pseudo-labels using the frozen classifier head, $\hat{y} = \arg\max_{k \in \mathcal{Y}} [h(g(\mathbf{z}))]_k$. Given a target quota of K samples per class for C classes, we accept a draw only if its pseudo-label corresponds to a class whose quota is not yet filled; otherwise the draw is discarded. We measure the total number of raw draws T required to reach the balanced target size CK.

Why overhead occurs: Let $q_k := \Pr(\hat{y} = k)$ denote the pseudo-label probability induced by the frozen classifier under the chosen noise distribution. In particular, for a designated forget class c_f, constructing a non-empty synthetic forget subset requires $q_f := \Pr(\hat{y} = c_f) > 0$. When balanced quotas are enforced across all classes, as quotas fill up, a growing fraction of new draws are pseudo-labeled into already-saturated classes and thus discarded, leading to $T > CK$. Intuitively, the stopping time is dominated by the rarest pseudo-label classes: if $q_{\min} = \min_k q_k$, then filling the last remaining class requires on the order of $K/q_{\min}$ draws. Therefore, (i) T grows approximately linearly with the sampling depth K, and (ii) the overhead ratio $T/(CK)$ is largely governed by pseudo-label imbalance (small $q_{\min}$), which tends to be more pronounced in higher-class settings.

Empirical overhead results: Table 14 reports T for CIFAR-10, CIFAR-100, and TinyImageNet under three embedding-space noise distributions (Gaussian, Laplace, Uniform), across four backbones and five random seeds. We use the following balanced quotas: CIFAR-10 (C=10, K=5000, total 50,000), CIFAR-100 (C=100, K=500, total 50,000), and TinyImageNet (C=200, K=500, total 100,000). Across settings, the overhead is stable across seeds and increases for datasets with more classes, consistent with the rare-class domination effect discussed above.

Table 14: Total number of raw synthetic draws required to reach balanced pseudo-label quotas per class. Quotas: CIFAR-10 (C=10, K=5000, total 50,000), CIFAR-100 (C=100, K=500, total 50,000), TinyImageNet (C=200, K=500, total 100,000). We report 5 random seeds per backbone. Noise is sampled in the embedding space and pseudo-labeled by the frozen classifier head (batch size = 2000).

Backbone	Seed	CIFAR-10			CIFAR-100			TinyImageNet		
		Gaussian	Laplace	Uniform	Gaussian	Laplace	Uniform	Gaussian	Laplace	Uniform
ResNet-18	1	64000	62000	70000	116000	94000	176000	286000	274000	268000
	2	62000	60000	68000	116000	112000	170000	202000	194000	244000
	3	64000	62000	70000	130000	118000	176000	292000	260000	316000
	4	62000	62000	66000	120000	102000	154000	202000	214000	230000
	5	68000	66000	74000	162000	94000	160000	128000	190000	222000
ResNet-50	1	62000	60000	68000	76000	70000	80000	214000	214000	212000
	2	62000	60000	70000	70000	66000	78000	202000	206000	204000
	3	62000	60000	72000	70000	64000	78000	190000	188000	204000
	4	66000	62000	72000	72000	64000	82000	222000	210000	230000
	5	66000	58000	64000	76000	74000	86000	186000	196000	196000
ViT-B/16	1	56000	54000	58000	158000	166000	202000	194000	198000	222000
	2	60000	58000	66000	194000	204000	230000	232000	228000	250000
	3	54000	54000	56000	188000	196000	204000	194000	186000	222000
	4	56000	56000	62000	176000	190000	192000	230000	224000	264000
	5	58000	56000	64000	244000	246000	296000	204000	198000	236000
Swin-T	1	58000	58000	56000	76000	76000	74000	192000	190000	204000
	2	60000	58000	58000	86000	78000	82000	180000	198000	182000
	3	54000	54000	54000	76000	72000	74000	176000	180000	166000
	4	54000	54000	54000	86000	76000	88000	178000	170000	176000
	5	56000	56000	56000	80000	82000	80000	156000	160000	150000

J Additional Multi-Class Unlearning Experiments

To further analyze the behavior of our method in larger multi-class unlearning settings, we extend the experiments beyond the settings reported in the main paper (2, 5, and 10 forget classes) and evaluate additional experiments with 20, 40, and 60 forget classes on CIFAR-100. These settings follow commonly used protocols in the literature. Specifically, Zhou et al. (2025) evaluate multi-class unlearning with 2, 5, 10, and 20 forget classes, while Tarun et al. (2023) study larger settings with 20, 40, and 60 forget classes. To maintain consistency with prior work, we adopt the same class selection used in Zhou et al. (2025) for the 20-class setting. For larger settings (40 and 60 classes), we extend this list following the class ordering provided in the public implementation of Zhou et al. (2025). The corresponding forget label sets are:

$$\begin{aligned}
\mathcal{Y}_f^{20} &= \{25, 58, 38, 23, 96, 54, 51, 49, 98, 66, 16, 52, 40, 71, 63, 79, 53, 12, 46, 55\},\\
\mathcal{Y}_f^{40} &= \{25, 58, 38, 23, 96, 54, 51, 49, 98, 66, 16, 52, 40, 71, 63, 79, 53, 12, 46, 55,\\
&\quad\ 83, 27, 41, 20, 30, 14, 70, 45, 61, 29, 4, 39, 21, 87, 60, 68, 75, 2, 92, 5\},\\
\mathcal{Y}_f^{60} &= \{25, 58, 38, 23, 96, 54, 51, 49, 98, 66, 16, 52, 40, 71, 63, 79, 53, 12, 46, 55,\\
&\quad\ 83, 27, 41, 20, 30, 14, 70, 45, 61, 29, 4, 39, 21, 87, 60, 68, 75, 2, 92, 5,\\
&\quad\ 57, 42, 0, 8, 97, 31, 50, 47, 13, 80, 34, 91, 17, 69, 85, 76, 94, 73, 99, 74\}.
\end{aligned}$$

All experiments follow the same evaluation protocol used in the main paper. The corresponding results for these larger multi-class settings are reported in the table 15.

Table 15: Multi-class unlearning performance for CIFAR-100 using ResNet-18 as the base architecture. Rows highlighted in gray correspond to methods applied on synthetic embeddings, while the non-shaded rows use original embeddings. Columns $\mathcal{D}_r$-free and $\mathcal{D}_f$-free indicate whether the method operates without access to the retain or forget set, respectively, with (✓) denoting true and (✗) denoting false.

Method	$\mathcal{D}_r$ free	$\mathcal{D}_f$ free	20-Classes $\mathcal{A}_r^t \uparrow$	20-Classes $\mathcal{A}_f^t \downarrow$	20-Classes AUS ↑	40-Classes $\mathcal{A}_r^t \uparrow$	40-Classes $\mathcal{A}_f^t \downarrow$	40-Classes AUS ↑	60-Classes $\mathcal{A}_r^t \uparrow$	60-Classes $\mathcal{A}_f^t \downarrow$	60-Classes AUS ↑
Original	–	–	78.30 ± 1.19	77.61 ± 1.36	0.563 ± 0.004	77.67 ± 1.44	78.91 ± 0.86	0.559 ± 0.003	76.49 ± 1.34	79.28 ± 1.13	0.558 ± 0.004
Retrained	–	–	82.83	0.00	1.032	81.67	0.00	1.048	85.47	0.00	1.076
FT (Golatkar et al., 2020)	✗	✓	78.99 ± 1.43	1.94 ± 3.70	0.989 ± 0.032	81.12 ± 0.89	0.60 ± 0.76	1.028 ± 0.013	83.73 ± 0.69	0.00 ± 0.00	1.072 ± 0.007
	✓	✓	80.29 ± 0.98	0.00 ± 0.00	1.020 ± 0.002	81.69 ± 0.95	0.01 ± 0.01	1.040 ± 0.006	83.50 ± 0.84	0.05 ± 0.06	1.070 ± 0.008
NG (Golatkar et al., 2020)	✓	✗	80.27 ± 0.98	0.06 ± 0.13	1.019 ± 0.003	79.38 ± 3.22	0.00 ± 0.00	1.017 ± 0.025	80.06 ± 4.77	1.12 ± 1.45	1.024 ± 0.035
	✓	✓	80.38 ± 0.99	0.01 ± 0.02	1.021 ± 0.002	81.86 ± 0.86	0.04 ± 0.05	1.042 ± 0.007	83.89 ± 0.68	0.06 ± 0.08	1.073 ± 0.009
RL (Hayase et al., 2020)	✓	✗	79.34 ± 1.00	0.00 ± 0.00	1.010 ± 0.005	79.57 ± 1.19	0.00 ± 0.00	1.019 ± 0.007	79.38 ± 1.29	0.00 ± 0.00	1.029 ± 0.007
	✓	✓	79.37 ± 1.37	0.08 ± 0.10	1.010 ± 0.006	76.77 ± 2.41	0.95 ± 0.81	0.982 ± 0.019	76.51 ± 1.09	2.07 ± 0.70	0.980 ± 0.020
NG+ (Kurmanji et al., 2023)	✗	✗	80.50 ± 0.94	0.00 ± 0.00	1.022 ± 0.003	81.92 ± 0.97	0.01 ± 0.02	1.042 ± 0.006	83.03 ± 0.80	0.00 ± 0.00	1.065 ± 0.006
	✓	✓	80.36 ± 0.98	0.00 ± 0.00	1.021 ± 0.002	81.79 ± 0.95	0.00 ± 0.00	1.041 ± 0.005	83.74 ± 0.79	0.00 ± 0.01	1.072 ± 0.006
SCRUB (Kurmanji et al., 2023)	✗	✗	80.41 ± 0.90	0.00 ± 0.00	1.021 ± 0.004	82.08 ± 1.06	0.00 ± 0.00	1.044 ± 0.005	83.70 ± 0.85	0.00 ± 0.00	1.072 ± 0.007
	✓	✓	79.77 ± 1.08	0.03 ± 0.04	1.014 ± 0.004	80.77 ± 1.25	0.01 ± 0.01	1.031 ± 0.008	82.05 ± 1.15	0.03 ± 0.04	1.055 ± 0.012
SCAR (Bonato et al., 2024)	✗	✗	80.50 ± 1.01	0.00 ± 0.00	1.022 ± 0.002	82.12 ± 0.99	0.00 ± 0.00	1.045 ± 0.005	84.34 ± 0.87	0.00 ± 0.00	1.079 ± 0.006
	✓	✓	80.51 ± 1.01	0.00 ± 0.00	1.022 ± 0.002	82.11 ± 0.97	0.00 ± 0.00	1.044 ± 0.005	84.39 ± 0.82	0.00 ± 0.00	1.079 ± 0.006