LoTUS: Large-Scale Machine Unlearning with a Taste of Uncertainty

Christoforos N. Spartalis[1,2] Theodoros Semertzidis[2] Efstratios Gavves[1,3] Petros Daras[2]
[1]University of Amsterdam, [2]Centre for Research & Technology Hellas, [3]Archimedes/Athena RC
{c.spartalis,e.gavves}@uva.nl {c.spartalis,theosem,daras}@iti.gr

Abstract

This paper, accepted at CVPR 2025, presents LoTUS, a novel Machine Unlearning (MU) method that eliminates the influence of training samples from pre-trained models, avoiding retraining from scratch. LoTUS smooths the prediction probabilities of the model up to an information-theoretic bound, mitigating its over-confidence stemming from data memorization. We evaluate LoTUS on Transformer and ResNet18 models against eight baselines across five public datasets. Beyond established MU benchmarks, we evaluate unlearning on ImageNet1k, a large-scale dataset, where retraining is impractical, simulating real-world conditions. Moreover, we introduce the novel Retrain-Free Jensen-Shannon Divergence (RF-JSD) metric to enable evaluation under real-world conditions. The experimental results show that LoTUS outperforms state-of-the-art methods in terms of both efficiency and effectiveness. Code: https://github.com/cspartalis/LoTUS.

1. Introduction

Machine Unlearning focuses on removing the influence of training samples from pre-trained models without retraining the model entirely [26]. Its applications include privacy protection in Machine Learning [3, 14, 15]. As an alternative to retraining a model from scratch, Machine Unlearning addresses three principal challenges: ① minimizing the time window during which the model is vulnerable, ② minimizing the cost in terms of time and computational resources, and ③ minimizing the dependency on access to all training data to retain the utility of the pre-trained model, as full data access is often limited due to privacy policies and storage limitations. Therefore, an effective and efficient unlearning algorithm should meet the following requirements [13]: ① Effectively eliminate the impact of specific training samples from the model. ② Retain the model's performance on the remaining training samples, even if access to the training set is limited. ③ Be efficient in terms of both time and computational resources.

Considering only the effectiveness of unlearning, the *gold standard* is to retrain the model from scratch without the samples designated for unlearning (also known as forget samples). To this end, two main taxonomy classes have been developed: *exact unlearning*, which aims to produce a model that is statistically indistinguishable from the gold standard, which is often infeasible for complex algorithms [4] or inefficient [3], and *approximate unlearning*, which relaxes the constraints of exact unlearning and adopts a suite of evaluation metrics that typically measure how well the unlearned model approximates the gold standard in terms of accuracy and resilience to privacy attacks [12]. The scope of this study concerns the following questions:

Q1: Can an unlearning method efficiently eliminate the influence of training samples from a pre-trained model while approximating the effectiveness of the gold standard?

Q2: Can this unlearning method effectively handle large-scale datasets and models under real-world constraints, including limited data access?

To answer these questions we propose the Logits Tempering Unlearning Strategy (*LoTUS* for short, such as the fruit that made Ulysses' comrades forget). LoTUS leverages the known tendency of Deep Neural Networks (DNNs) to memorize sample-specific features from training data and output over-confident predictions [38], a vulnerability exploited by Membership Inference Attacks (MIAs) to assess whether a sample is a member of the training set [33]. To this end, LoTUS smooths the model's output probabilities, as shown in Fig. 1, increasing the entropy to resemble that of unseen (during training) samples. This unseen set, which may include synthetic data, enables LoTUS to calibrate the retained information for forget samples post-unlearning and replicate the decision-making process of the gold standard model. Since the gold standard model was not trained on the forget samples, it naturally avoids over-confident predictions and typically exhibits lower accuracy on them. To better approximate the gold standard's performance, LoTUS also introduces Gumbel noise into the pre-trained model's output distribution. This encourages diverse predictions and helps reduce the pre-trained model's accuracy on forget samples to resemble that of the gold standard.

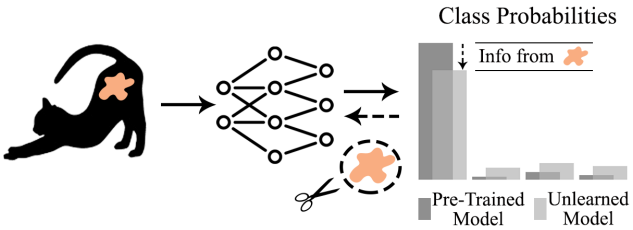


Figure 1. **Machine Unlearning via smoothing prediction probabilities:** LoTUS eliminates sample-specific information (e.g., unique fur patterns in cat images) that the DNN memorized and exposed through overconfident predictions. Then, the DNN responds to unlearned samples as if they were never part of the training set.

In contrast to previous studies that have focused mainly on the input or model space, LoTUS follows an *entropy-based* approach that directly modifies the model's output probabilities, emphasizing an underexplored unlearning approach. The difference from the existing method which indiscriminately maximizes entropy using random labeling [17] is that LoTUS uses an *information-theoretical* bound to control the uncertainty introduced to the model. Our main contributions are as follows:

1. We introduce LoTUS, the first unlearning method that operates directly in the model's output space, while following an information-theoretic approach to determine the amount of entropy increase. This bound enables cautious unlearning that approximates the gold standard.

2. We introduce the *Retrain-Free Jensen-Shannon Divergence (RF-JSD)* unlearning metric, which enables evaluation in real-world scenarios. RF-JSD exhibits a strong Pearson correlation (PCC $= 0.92_{\pm 0.04}$) with the established JSD score while eliminating the need to retrain the model. Compared to the existing retrain-free ZRF score, RF-JSD offers enhanced interpretability and efficiency.

3. We introduce a novel large-scale experimental setup that incorporates a large-scale dataset (ImageNet1k), and limited access to the training set, with the aim of simulating real-world conditions, where model retraining is infeasible. Overall, we evaluated LoTUS on the Vision Tranformer and ResNet18 modes, against eight baseline methods on five public datasets. Extensive experiments demonstrate that LoTUS outperforms state-of-the-art approaches in terms of both unlearning effectiveness and efficiency, in all benchmarks (novel and established).

2. Related Work

Machine Unlearning was first introduced in [4] with an approach that decomposes traditional Machine Learning algorithms into summations, enabling the reduction of the influence of specific data points for exact unlearning. Subsequently, a theoretical framework for approximate unlearning was proposed in [18], suggesting a Hessian-based regularization technique limited to models with convex loss functions to mitigate membership inference risks. Unlearning was subsequently extended to deep neural networks in [15] by introducing a Lagrangian regularization approach that utilizes the Fisher Information Matrix as a Hessian approximation. More recent works have improved unlearning effectiveness and efficiency [14, 23] and expanded machine unlearning applications in diverse areas, including user privacy [30], security defense [29], toxic content removal [12, 20], copyright protection [16], and bias mitigation [7]. Emphasizing on privacy applications, Machine Unlearning has been defined as a privacy game aiming to reduce the accuracy of MIAs [3].

Algorithms are categorized into two classes depending on where the manipulations are applied: model space and data space [37]. In model-space approaches, manipulations include regularizing the loss function to shift model weights far from the pre-trained model and close to the gold standard [9, 15, 23]. Another approach is pruning, which involves identifying and reducing the influence of weights that are most affected by the data to be unlearned [12, 14]. Although model-space approaches can offer theoretical justifications and efficiency, they also present challenges in terms of implementation complexity and interpretability [37].

On the other hand, data-space approaches focus on reorganizing or modifying the data to be unlearned. These methods include data-partitioning techniques that track which partition each data point belongs to and the corresponding model updates they trigger. They enable selective forgetting by isolating specific model updates [4, 17] or by retraining the model from the latest valid checkpoint [3]. These techniques are usually pre-hoc; meaning they must be applied before training, and cannot apply to pre-trained models. Also, they are resource-intensive, trading increased space complexity for reduced time complexity.

Data obfuscation is a data-space approach that can be applied post-hoc. This includes methods such as adversarial attacks [5, 6] or adding noise to the input [9, 13]. Although these techniques primarily focus on modifications in the input space, Random Labeling [17] takes a different approach by altering the output space and reassigning incorrect labels to the forget samples. Despite its simplicity, this approach has been shown to be effective [35]. Data-space adjustments are more conceptually aligned with information theory, although a direct connection was explicitly established only recently in [13], which explores input perturbations.

Information Theory formalizes the quantification of information through mathematical measures such as entropy and mutual information [31]. In the context of DNNs, information is typically defined for random variables such as the input and output of the models.

3. Logits Tempering Unlearning Strategy

3.1. Preliminaries

Let $x \sim P(X)$ be a feature vector representing an image sampled from the sampling space $A(X)$, and $y \sim P(Y)$ be a classification label sampled from the sampling space $A(Y) = \{c_1, c_2, \ldots, c_k\}$, where k is the total number of classes. Let $f_w(X)\colon A(X) \mapsto A(Y)$ be a DNN model parameterized by weights w that maps an image x to a classification label y. Also, let $D = \{(x_i, y_i)\}_{i=1}^{n} = D_f \cup D_r \cup D_u$ be a dataset of images $x_i \in A(X)$ and their corresponding labels $y_i \in A(Y)$, which comprises three pairwise disjoint datasets: ① **Forget set D_f**: Training samples whose influence on the model weights w should be removed. ② **Retain set D_r**: Training samples whose influence on w must be preserved. ③ **Unseen set D_u**: Samples that were not used to train the model f_w. As unseen sets, we use either the validation sets or synthetic data generated from training data. Finally, we denote: f_{orig} as the pre-trained (or original) model, trained on $D_f \cup D_r$, f_{gold} as the gold standard model, retrained from scratch only on D_r, and f_{un} as the model derived from unlearning, which is the process of updating the model weights of f_{orig} so that $f_{\text{un}}(x) \approx f_{\text{gold}}(x),\ \forall x \in D$.

3.2. Upper-bounding Uncertainty

Unlike existing unlearning methods [17], which indiscriminately increase entropy in the output space, we aim to establish an upper bound on the uncertainty introduced by unlearning, removing only the information specific to the forget set D_f which extends beyond the model's general knowledge. To achieve this, we adopt an information-theoretic framework to delineate the information essential for preserving model utility from the information that needs to be removed. Although directly estimating the mutual information between the model's input and output would be ideal, this approach is both challenging and computationally intensive [1]. Therefore, we introduce a relaxed version of the framework that enables the assessment of the appropriate entropy increase required for unlearning.

Proposition. Let X_s be a random variable with any sampling space $A(X_s) \subset A(X)$. In other words, X_s is derived from X by filtering. Then, $X \to X_s \to f_w(X_s)$ is a processing chain where $f_w(X_s)$ depends on X only through X_s. By the Data Processing Inequality [10], this is a Markov chain that implies $f_w(X_s) \to X_s \to X$. Therefore, by the chain rule, we can expand the mutual information in two different ways:

$$I\big(f_w(X_s); X_s, X\big) = I\big(f_w(X_s); X\big) + I\big(f_w(X_s); X_s \,|\, X\big) = I\big(f_w(X_s); X_s\big) + \cancelto{0}{I\big(f_w(X_s); X \,|\, X_s\big)} \tag{1}$$

Since $f_w(X_s)$ is conditionally independent of X given X_s, it follows that $I\big(f_w(X_s); X \mid X_s\big) = 0$. Therefore, from Eq. (1), the mutual information between the input X_s and the output $f_w(X_s)$ of the classifier is:

$$\underbrace{I(f_w(X_s); X_s)}_{\substack{\text{total information captured by} \\ \text{the model from the data subset}}} = \underbrace{I(f_w(X_s); X)}_{\substack{\text{global} \\ \text{information}}} + \underbrace{I(f_w(X_s); X_s \,|\, X)}_{\substack{\text{subset-specific} \\ \text{information}}} \tag{2}$$

We consider $I(f_w(X_s); X)$ as the *global information* a model f_w has captured from the set $A(X)$. In other words, it quantifies the contribution of the shared features among training samples in $A(X)$ (*i.e.*, global features) to the model's decision-making. Respectively, we consider $I(f_w(X_s); X_s \mid X)$ as the additional *subset-specific information* learned exclusively from the subset $A(X_s)$, which refines the model's decision and adds detail beyond what is already captured from $A(X)$.

For example, if there are images of *cats* in both $A(X)$ and its subset $A(X_s)$, then the *total information* captured from the images in $A(X_s)$ can be categorized into two types: The *global information* learned from shared features across all cat images in $A(X)$, *e.g.* body shape of cats; and the additional *subset-specific information* learned exclusively from cat images in $A(X_s)$, *e.g.* unique fur patterns.

To determine the presence of *subset-specific information* and how this is expressed in the model's decision, we refer to the memorization capabilities of DNNs and the derived privacy considerations. DNNs are known to memorize information from individual samples in the training set [38]. Considering a DNN classifier, the memorization of specific patterns is exposed in the model's output probabilities via increased confidence (*i.e.*, lower entropy in the model's output probability distribution), and this is an indicator exploited by privacy attacks to distinguish which samples are members of the training set [32].

Therefore, if $A(X_s)$ is a subset of the training set, then the model can capture the *subset-specific information* leading to over-confident predictions. However, if $A(X_s)$ was unseen during training, then the model had no chance to capture *subset-specific information* and its predictions are based solely on the *global information* captured from training samples in $A(X)$. Defining the sampling space of X as $A(X) = D = D_f \cup D_r \cup D_u$, and the sampling space of X_s as the forget set $A(X_s) = D_f \subset D$, we can assess the *total information* captured by the available pre-trained model f_{orig} and the ideal gold standard model f_{gold} as such:

$$I\big(f_{\text{orig}}(X_s), X_s \in D_f\big) = I\big(f_{\text{orig}}(X_s), X\big) + I\big(f_{\text{orig}}(X_s); X_s \,|\, X\big) \tag{3}$$

$$I\big(f_{\text{gold}}(X_s), X_s \in D_f\big) = I\big(f_{\text{gold}}(X_s), X\big) + \cancelto{0}{I\big(f_{\text{gold}}(X_s); X_s \,|\, X\big)} \tag{4}$$

The gold standard model f_{gold} has not been trained on the

forget set D_f, thus f_{gold} has not captured *subset-specific information* from D_f as shown in Eq. (4).

Based on Eqs. (3) and (4), we define Machine Unlearning as the process of eliminating the *subset-specific information* $I\big(f_{\text{orig}}(X_s);X_s\,|X\big)$ from the pre-trained model (*i.e.*, forgetting objective), while retaining the *global information* $I\big(f_{\text{orig}}(X_s);X\big)$ captured from the training samples in D (*i.e.*, retention objective to preserve model's utility on the remaining training samples). Therefore, the *total information* that the unlearned model f_{un} retains for the samples in the forget set $X_s \in D_f$ is by definition equal to the *global information* the pre-trained model f_{orig} had captured from the training set $X \in D_f \cup D_r$:

$$I\big(f_{\text{un}}(X_s);X_s\big) \triangleq I\big(f_{\text{un}}(X_s);X\big) \triangleq I\big(f_{\text{orig}}(X_s);X\big) \quad (5)$$

Assumption of instance-wise unlearning: Equation (5) holds under the condition that the forget set D_f comprises only a subset of the training samples of a class (*i.e.*, instance-wise unlearning) and not all class samples (*i.e.*, class unlearning). For example, if D_f contains images of cats, then the retain set D_r must also include images of cats to ensure that the global features related to the cat class are still encoded in the model after unlearning. Otherwise, the *global information* will be eliminated during unlearning and $I(f_{\text{orig}}(X_s),X) \neq I\big(f_{\text{un}}(X_s),X\big)=0$.

Subsequently, we focus on instance-wise unlearning and the quantification of the *global information* that should be retained post-unlearning. However, in Sec. 14, we provide details on the class unlearning task and how LoTUS can be easily adapted to this.

Quantifying *global information*. Estimating the *global information* $I\big(f_{\text{orig}}(X_s);X\big)$ is challenging due to the high dimensionality and complex dependencies in data, making it difficult and computationally intensive [1]. To address this, we use the *total information* $I\big(f_{\text{orig}}(X_s);X_s\big)$ as a proxy of the *global information* and conclude on an efficient yet effective weaker approximation.

As previously explained and shown in Eq. (4), a model cannot capture *subset-specific information*, if this subset has not been used during training. Therefore, if D_u consists of unseen (during training) samples, then:

$$I\big(f_{\text{orig}}(X_s),X_s{\in}D_u\big) = I\big(f_{\text{orig}}(X_s),X\big) + \cancelto{0}{I\big(f_{\text{orig}}(X_s),X_s\,|\,X\big)} \quad (6)$$

Assumption of Distributional Similarity: The forget set D_f and unseen set D_u are assumed to follow the same distribution in terms of visual features and class distribution. This leads to the conclusion that their entropies are equal: $H(X_s \in D_f) = H(X_s \in D_u)$. Additionally, the *total information* captured by the unlearned model f_{un} from D_f can be considered equivalent to that captured by the pre-trained model f_{orig} from D_u. Based on Eqs. (5) and (6), we can thus reformulate the unlearning objective as:

$$\begin{aligned} I\big(f_{\text{un}}(X_s),X_s\in D_f\big) = I\big(f_{\text{orig}}(X_s),X_s\in D_u\big) \Rightarrow \\ \cancel{H(X_s\in D_f)} - H\big(X_s\in D_f\,|\,f_{\text{un}}(X_s)\big) = \\ \cancel{H(X_s\in D_u)} - H\big(X_s\in D_u\,|\,f_{\text{orig}}(X_s)\big) \Rightarrow \\ H\big(X_s\in D_f\,|\,f_{\text{un}}(X_s)\big) = H\big(X_s\in D_u\,|\,f_{\text{orig}}(X_s)\big) \end{aligned} \quad (7)$$

which establishes that the uncertainty about whether a sample belongs to the forget or unseen set should be the same when conditioned on the respective model's outputs.

Although this theoretical formulation assumes an identical distribution for D_f and D_u, we show that the assumption can be relaxed in practice. Specifically, the images in both sets only need to share relevant features that contribute to the *global information*. In other words, the forget and unseen sets should contain visually similar images rather than images with exactly the same information. For example, if the forget set contains cat images, the unseen set should also contain cat images –even synthetic ones– rather than entirely different objects such as human portraits. In practice, this ensures sufficient similarity in global features related to the cat class.

Approximating conditional entropy. Given the complexity of the underlying distributions and the computational challenge associated with entropy estimation in Eq. (7), we derive a practical relationship linking the prediction error to the uncertainty in the model's predictions. Let $\widehat{X}_s$ be an estimate of X_s based on the model's output $f_w(X_s)$, following $X_s \rightarrow f_w(X_s) \rightarrow \widehat{X}_s$, and define the prediction error probability as $P_e = P\{X_s \neq \widehat{X}_s\}$. Then Fano's Inequality [10] states:

$$P_e \geq \frac{H\big(X_s \mid f_w(X_s)\big) - 1}{\log|A(X_s)|} \quad (8)$$

This inequality implies that lower prediction error P_e –or equivalently higher accuracy ($\text{Acc}=1-P_e$)– corresponds to reduced uncertainty $H\big(X_s \mid f_w(X_s)\big)$. Moreover, this aligns with the empirical observation that models tend to be more accurate in images for which they make predictions of higher confidence [36].

Since accuracy is straightforward to measure and computationally efficient, we approximate the conditional entropies in Eq. (7) and define a relaxed unlearning objective:

$$\text{Acc}(f_{\text{un}},D_f) = \text{Acc}(f_{\text{orig}},D_u) \quad (9)$$

where $\text{Acc}(f_{\square},D_{\triangle})$ denotes the prediction accuracy of a model $f_{\square}$ on a data subset $D_{\triangle}$. Equation (9) suggests that the unlearning process can be calibrated by aligning the accuracy of the unlearned model on the forget set with the accuracy of the pre-trained model on the unseen set.

3.3. Unlearning with LoTUS

LoTUS leverages the accuracy objective in Eq. (9) to regulate the increase in model's uncertainty. Specifically, LoTUS increases model's uncertainty by smoothing the predicted probabilities of forget samples to tackle memorization –evident in over-confident predictions– ensuring that the accuracy of the unlearned model f_{un} converges toward that of the pre-trained model f_{orig} on the unseen (during training) set. This approach not only eliminates the *subset-specific information*, but also preserves the *global information*, preventing over-unlearning and safeguarding the utility of the model on the remaining training samples.

To achieve this, LoTUS employs a knowledge distillation framework in which both the teacher and student models are initialized with the weights of the original model f_{orig}, as in [23]. The teacher serves as the original model f_{orig} throughout the unlearning process, while the student f_{un} undergoes unlearning by receiving perturbed knowledge from the teacher. This perturbation is applied during the activation of the teacher's logits using the Gumbel-Softmax function $gs(\cdot)$:

$$p_i = gs(\pi, \tau) = \frac{\exp\left(\left(\log \pi_i + g_i\right)/\tau\right)}{\sum_{j=1}^{k} \exp\left(\left(\log \pi_j + g_j\right)/\tau\right)},\ i = 1,\ldots,k \tag{10}$$

where p_i is the probability of class i, π_i is the corresponding logit, g_i is statistical noise sampled from the Gumbel distribution, k is the total number of classes, and $\tau \in \mathbb{R}^+$ is a temperature parameter that controls the sharpness of the output probabilities: smoothing them when $\tau > 1$, sharpening them when $\tau < 1$, and leaving them unchanged when $\tau = 1$.

Temperature τ is the key component in LoTUS, as it controls the uncertainty introduced to the student by adjusting the entropy in the teacher's output probabilities. In each unlearning epoch, the temperature τ is dynamically adjusted based on Eq. (9) as follows:

$$\tau_d = \exp\left(\alpha \cdot \left(\text{Acc}(f_{\text{un}}, D_f) - \text{Acc}(f_{\text{orig}}, D_u)\right)\right) \tag{11}$$

where f_{un} and f_{orig} are the student and teacher models, respectively; D_f is the forget set and D_u the unseen set (*i.e.*, validation set or synthetic data); and $\alpha \in \mathbb{R}^+$ is a positive value that scales the accuracy difference.

This implementation facilitates convergence to the unlearning objective Eq. (9) by dynamically adjusting the entropy in the teacher's output probabilities as follows: ① At the beginning of the unlearning process, when the student model is initialized with the weights of the f_{orig}, the student's accuracy on D_f exceeds the teacher's accuracy on unseen data, since the D_f comprises training data; therefore, $\tau_d > 1$ and the teacher's output probabilities are smoothed to increase the entropy in the output space and induce uncertainty in the student model. ② As the unlearning process continues, LoTUS converges to the unlearning objective Eq. (9), the accuracy difference becomes smaller, and uncertainty is introduced with smaller steps, proportional to this accuracy difference, facilitating smooth convergence. ③ If, during the unlearning process, the entropy in the student's output probabilities exceeds the desired level, then the student's accuracy on the forget set decreases below the teacher's accuracy on unseen data; therefore, $\tau_d < 1$ and the teacher's output probabilities are sharpened to restore the entropy in the student's output probabilities to the desired level. In Fig. 2, we illustrate how the dynamically adjusted temperature τ_d contributes to the convergence of the unlearning objective in Eq. (9).

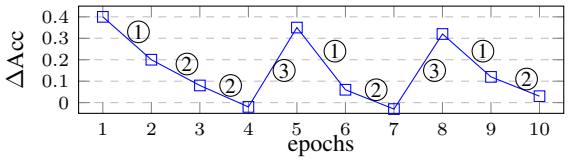


Figure 2. Contribution of the dynamically adjusted temperature τ_d to convergence toward the objective $\Delta\text{Acc} = \text{Acc}(f_{\text{un}}, D_f) - \text{Acc}(f_{\text{orig}}, D_u) = 0$. The steps are denoted as follows: ①: sharp step towards objective, ②: smaller step (proportional to ΔAcc), ③: drastic accuracy restoration when over-unlearning.

Statistical noise $g \sim \text{Gumbel}(0, 1)$ added to the teacher's logits also contributes to the unlearning process. While smoothing the output probabilities does not typically alter the prediction outcome, the stochasticity introduced by g facilitates the student model f_{un} to produce predictions that differ from those of the well-conveged and accurate teacher model f_{orig}. This reduces the student's accuracy on the forget set D_f and drives convergence towards the objective in Eq. (9). This observation aligns with the ablation analysis of Gubmel-Softmax *vs.* Softmax in Sec. 15.

To this end, the loss function in LoTUS, which guides the student model f_{un} to align with the perturbed output probabilities of the teacher model f_{orig}, is defined for a single instance x as follows:

$$\begin{aligned}\ell(x, f_{\text{orig}}, f_{\text{un}}) = &\underbrace{l \cdot gs\big(f_{\text{orig}}(x), \tau_d\big) \odot \log s\big(f_{\text{un}}(x)\big)}_{\text{forget}} \\ &+ \underbrace{(1-l) \cdot gs\big(f_{\text{orig}}(x), \tau \to 0^+\big) \odot \log s\big(f_{\text{un}}(x)\big)}_{\text{retain}}\end{aligned} \tag{12}$$

where $l \in \{0, 1\}$ is an unlearning label, similar to [9], indicating whether the instance belongs to the forget set D_f or the retain set D_r, $gs(\cdot)$ is the Gumbel-Softmax function as in Eq. (15), and $s(\pi) = \exp(\pi_i)/\sum_{j=1}^{k} \exp(\pi_j)$ is the Softmax function for $i = 1, \ldots, k$, where k is the total number of classes. For forget samples, temperature τ_d is dynamically scheduled, as shown in Eq. (11), while for retain samples, τ is assigned a near-zero value to sharpen the teacher's output distribution to the greatest extent, decreasing the entropy, and enhancing retention.

4. Experimental Setup

We focus on the instance-wise unlearning task, while in Sec. 14, we propose a LoTUS adaptation to the class unlearning task. The forget sets consists of 10% or 50% of the training data, following [12]. LoTUS uses only 30% of the remaining training samples as the retain set to evaluate its robustness in scenarios with limited data access. To emphasize real-world conditions, we also evaluate unlearning a small portion of a large-scale dataset while restricting access to the original training data, making retraining from scratch infeasible. To assess unlearning performance under these constraints, we introduce the novel *Retrain-Free Jensen-Shannon Divergence (RF-JSD)* metric.

Data. Following [7, 9, 23, 34], we use the CIFAR-10/100 datasets [21], which consist of $50,000$ training samples across 10 and 100 classes, respectively. Moreover, we use the domain-specific MUFAC dataset [8] with 8 classes and fixed forget/retain splits. After cleaning MUFAC (see Sec. 12), the forget set consists of $\sim$16% of the training data. Additionally, we test TinyImageNet [24], which contains $100,000$ images of 200 classes and exhibits more complex data statistics than CIFAR-10/100, to further validate –beyond MUFAC– that the assumption of distribution similarity between the forget and unseen sets in Sec. 3.2 can be relaxed. To reinforce this finding, we include an experiment with the CIFAR-10 and CIFAKE [2] datasets, where the unseen set is not the validation set of CIFAR-10 but consists of synthetic AI-generated data from CIFAKE. For large-scale unlearning, we use the ImageNet1k dataset [28] which contains ~ 1.2M training samples of $1,000$ classes. Following [27], we split the training set into forget/retain sets in a stratified manner to ensure robust evaluation.

Evaluation Metrics. Following [9, 12, 14] we evaluate the unlearning methods based on how closely they approximate the gold standard model, in terms of MIA accuracy and accuracy on the forget/retain/test sets, using the Average (Avg) Gap metric [12]:

$$\text{Avg Gap} = \frac{1}{4}(|\Delta\text{Acc}_{\text{MIA}}|+|\Delta\text{Acc}_f|+|\Delta\text{Acc}_r|+|\Delta\text{Acc}_t|)$$

where $|\Delta\text{Acc}|$ is the absolute difference in accuracy between the the unlearned and gold standard models, Acc_{MIA} is the accuracy of the Membership Inference Attack used in [9, 14], and Acc_f, Acc_r, Acc_t are the accuracies on the forget, retain, and test sets, respectively. Small $\Delta\text{Acc}_{\text{MIA}}$ and ΔAcc_f indicate effective unlearning while small ΔAcc_r and ΔAcc_t suggest effective retention. Thus, Avg Gap reflects the balance between forgetting and retention.

Following [9], we use the Jensen-Shannon Divergence (JSD) to further assess unlearning effectiveness and resilience to the Streisand Effect (*i.e.*, when unlearning unintentionally makes forget samples more identifiable to attackers). The JSD provides a more sensitive measure than accuracy, as it captures distributional differences between the outputs of the unlearned and gold standard models:

$$\mathcal{JS}\big(f_{\text{un}}(D_f) \,||\, f_{\text{gold}}(D_f)\big) = \frac{1}{|D_f|}\sum_{x\in D_f}\Big(0.5\cdot\mathcal{KL}\big(f_{\text{un}}(x)\,||\,m\big)+0.5\cdot\mathcal{KL}\big(f_{gold}(x)\,||\,m\big)\Big)$$

where $\mathcal{JS}$ is the Jensen-Shannon divergence [25], $\mathcal{KL}$ is the Kullback-Leibler divergence [22], $|D_f|$ is the number of samples in the forget set, $f_{\text{un}}(x)$ and $f_{\text{gold}}(x)$ are the predicted probability distributions for a sample x, and m is their average, defined as $m = \big(f_{\text{un}}(x) + f_{\text{gold}}(x)\big)/2$.

Also, we introduce the novel Retrain-Free Jensen-Shannon Divergence (RF-JSD) metric, which does not rely on the gold standard model f_{gold}, making it useful in real-world scenarios where model retraining is impractical or infeasible. RF-JSD is computed by first averaging the predicted probability distributions per class from the unlearned model on the forget set and the pre-trained model on the unseen set, then averaging the JSD values between the normalized class-wise mean distributions of these models:

$$\mathcal{JS}\big(f_{\text{un}}(D_f)\,||\,f_{\text{orig}}(D_u)\big) = \frac{1}{k}\sum_{c=1}^{k}\mathcal{JS}(P_i \,||\, Q_i)$$

$$P_i = \frac{1}{Z_P}\sum_{j=1}^{n_i} f_{\text{un}}(x_j\,|\,y_j=i)\,,\; Q_i = \frac{1}{Z_Q}\sum_{j=1}^{n_i} f_{\text{orig}}(x_j\,|\,y_j=i)$$

where P_i and Q_i are the normalized class-wise mean distributions for the class i, k is the total number of classes, n_i is the number of samples in class i, and Z_P, Z_Q are sums of the mean class probabilities used for L1-normalization, ensuring that P, Q are valid probability distributions.

RF-JSD provides greater interpretability than the retrain-free ZRF score [9] by aligning with the well-established JSD and maintaining a consistent optimal value of zero across different models, datasets, and forget sets. Additionally, RF-JSD is more computationally efficient, as it avoids the need for an extra randomly initialized model to establish a reference score, unlike ZRF.

Models and Training. We use Vision Transformer [11] and ResNet18 [19] architectures. Unlearning runs for 3 epochs in ViT models and 10 epochs in ResNet18 models, as in [9]. We use the AdamW optimizer with a weight decay of 5×10^{-4}. Learning rates are set to 10^{-6} for ViT and 10^{-4} for ResNet18. We perform minimal hyperparameter tuning, only on α in Eq. (11) via a search over $\{2, 4, 8, 16\}$ to minimize the Avg Gap score without using the test set, as in [14]; the optimal value is $\alpha = 2$. For baselines, we use the hyperparameters specified in the original papers. Baseline and hyperparameters descriptions are provided in the Supp. Material. Batch sizes remain consistent across all methods. Each experiment is evaluated using three seeds, which are also used to sample various forget sets.

		Metric (↓)	Gold Std	Finetuning	NegGrad+ [23]	RndLbl [17]	BadT [9]	SCRUB [23]	SSD [14]	UNSIR [34]	SalUn [12]	**LoTUS**
Vision Transformer (ViT)	TinyIN	Avg Gap	0.0000	0.0175	0.0400	0.2925	0.0775	0.0225	0.0225	0.0225	0.0925	**0.0150**
		JSD $\times 1e4$	$0.00_{\pm 0.00}$	$0.05_{\pm 0.00}$	$0.10_{\pm 0.00}$	$0.64_{\pm 1.03}$	$0.18_{\pm 0.01}$	$0.04_{\pm 0.00}$	$0.04_{\pm 0.00}$	$0.06_{\pm 0.00}$	$0.25_{\pm 0.59}$	$\mathbf{0.03}_{\pm \mathbf{0.00}}$
		Time (min.)	$228.9_{\pm 6.49}$	$22.64_{\pm 0.02}$	$25.20_{\pm 0.02}$	$25.19_{\pm 0.02}$	$16.91_{\pm 0.05}$	$33.25_{\pm 0.01}$	$27.27_{\pm 0.06}$	$21.17_{\pm 0.22}$	$76.97_{\pm 1.72}$	$\mathbf{13.41}_{\pm \mathbf{0.04}}$
	C-100	Avg Gap	0.0000	0.0275	0.0325	0.0175	0.0375	0.0200	0.0175	0.0250	0.0200	**0.0125**
		JSD $\times 1e4$	$0.00_{\pm 0.00}$	$0.07_{\pm 0.00}$	$0.13_{\pm 0.01}$	$0.06_{\pm 0.00}$	$0.17_{\pm 0.01}$	$\mathbf{0.04}_{\pm \mathbf{0.00}}$	$\mathbf{0.04}_{\pm \mathbf{0.00}}$	$0.08_{\pm 0.01}$	$0.06_{\pm 0.01}$	$\mathbf{0.04}_{\pm \mathbf{0.02}}$
		Time (min.)	$112.25_{\pm 0.13}$	$11.35_{\pm 0.00}$	$12.63_{\pm 0.01}$	$12.79_{\pm 0.02}$	$9.18_{\pm 0.27}$	$16.74_{\pm 0.03}$	$13.67_{\pm 0.02}$	$10.69_{\pm 0.01}$	$38.15_{\pm 0.04}$	$\mathbf{7.02}_{\pm \mathbf{0.01}}$
	MUFAC	Avg Gap	0.0000	0.0400	0.0475	**0.0200**	0.1750	**0.0200**	**0.0200**	0.0475	**0.0200**	**0.0200**
		JSD $\times 1e{-4}$	$0.00_{\pm 0.00}$	$0.27_{\pm 0.02}$	$0.39_{\pm 0.04}$	$0.35_{\pm 0.09}$	$1.89_{\pm 1.01}$	$\mathbf{0.05}_{\pm \mathbf{0.02}}$	$0.17_{\pm 0.17}$	$0.85_{\pm 0.06}$	$0.29_{\pm 0.01}$	$\mathbf{0.05}_{\pm \mathbf{0.01}}$
		Time (min.)	$13.83_{\pm 0.01}$	$1.40_{\pm 0.01}$	$1.67_{\pm 0.00}$	$1.76_{\pm 0.01}$	$2.09_{\pm 0.25}$	$2.21_{\pm 0.01}$	$1.91_{\pm 0.00}$	$3.21_{\pm 0.01}$	$8.15_{\pm 0.01}$	$\mathbf{1.09}_{\pm \mathbf{0.00}}$
ResNet18 (RN18)	TinyIN	Avg Gap	0.0000	0.2200	0.2250	0.1925	0.2850	0.2725	0.2700	0.2375	0.2025	**0.1675**
		JSD $\times 1e4$	$0.00_{\pm 0.00}$	$1.80_{\pm 0.04}$	$1.82_{\pm 0.07}$	$1.71_{\pm 0.11}$	$1.81_{\pm 0.04}$	$0.98_{\pm 0.00}$	$0.96_{\pm 0.02}$	$1.76_{\pm 0.05}$	$1.93_{\pm 0.09}$	$\mathbf{0.62}_{\pm \mathbf{0.01}}$
		Time (min.)	$46.81_{\pm 0.57}$	$2.85_{\pm 0.00}$	$3.17_{\pm 0.00}$	$3.44_{\pm 0.01}$	$1.91_{\pm 0.01}$	$3.97_{\pm 0.00}$	$3.47_{\pm 0.00}$	$5.00_{\pm 0.01}$	$6.06_{\pm 0.06}$	$\mathbf{1.62}_{\pm \mathbf{0.00}}$
	C-100	Avg Gap	0.0000	0.3600	0.3575	0.4025	0.3675	0.1650	0.2125	0.3625	0.3650	**0.1200**
		JSD $\times 1e4$	$0.00_{\pm 0.00}$	$6.88_{\pm 0.59}$	$6.87_{\pm 0.62}$	$5.84_{\pm 0.98}$	$4.30_{\pm 0.49}$	$1.87_{\pm 0.08}$	$3.04_{\pm 1.55}$	$3.05_{\pm 0.32}$	$6.50_{\pm 0.60}$	$\mathbf{1.67}_{\pm \mathbf{0.37}}$
		Time (min.)	$3.39_{\pm 0.30}$	$0.43_{\pm 0.00}$	$0.49_{\pm 0.00}$	$0.57_{\pm 0.00}$	$0.34_{\pm 0.01}$	$0.58_{\pm 0.00}$	$0.54_{\pm 0.00}$	$0.45_{\pm 0.00}$	$1.55_{\pm 0.01}$	$\mathbf{0.30}_{\pm \mathbf{0.01}}$
	MUFAC	Avg Gap	0.0000	0.1525	0.1550	0.1300	**0.1025**	0.1625	0.1600	0.1450	0.1400	0.1250
		JSD $\times 1e4$	$0.00_{\pm 0.00}$	$19.52_{\pm 6.23}$	$19.16_{\pm 5.31}$	$9.51_{\pm 2.39}$	$9.41_{\pm 0.04}$	$10.53_{\pm 2.31}$	$10.30_{\pm 2.28}$	$16.32_{\pm 4.82}$	$15.25_{\pm 5.20}$	$\mathbf{6.90}_{\pm \mathbf{1.49}}$
		Time (min.)	$7.34_{\pm 0.77}$	$0.76_{\pm 0.00}$	$0.91_{\pm 0.00}$	$1.06_{\pm 0.00}$	$0.66_{\pm 0.00}$	$1.20_{\pm 0.00}$	$1.07_{\pm 0.00}$	$1.68_{\pm 0.02}$	$2.72_{\pm 0.02}$	$\mathbf{0.62}_{\pm \mathbf{0.00}}$

Table 1. **Performance Summary of unlearning 10% of Tiny-ImageNet (TinyIN), CIFAR-100 (C-100), and MUFAC training sets:** LoTUS outperforms state-of-the-art approaches in **balancing forgetting and retention** (measured by Avg Gap), **unlearning effectiveness and resilience to the Streisand effect** (indicated by JSD), and **efficiency** (reflected in Time, measured in minutes).

	Metric (↓)	Gold Std	Finetuning	NegGrad+	RndLbl	BadT	SCRUB	SSD	UNSIR	SalUn	**LoTUS**	**LoTUS synthetic D_u**
ViT	Avg Gap	0.0000	0.0075	0.0125	0.0125	0.0375	**0.0050**	0.0075	0.0100	0.0125	**0.0050**	0.0075
	JSD $\times 1e4$	$0.00_{\pm 0.00}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$	$0.03_{\pm 0.00}$	$0.02_{\pm 0.01}$	$0.12_{\pm 0.03}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$	$0.02_{\pm 0.01}$	$\mathbf{0.01}_{\pm \mathbf{0.01}}$	$\mathbf{0.01}_{\pm \mathbf{0.01}}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$
	Time (min.)	$111.00_{\pm 1.99}$	$11.33_{\pm 0.03}$	$12.61_{\pm 0.03}$	$12.78_{\pm 0.01}$	$8.97_{\pm 0.02}$	$16.66_{\pm 0.02}$	$13.65_{\pm 0.02}$	$10.68_{\pm 0.02}$	$37.97_{\pm 0.19}$	$7.34_{\pm 0.19}$	$\mathbf{7.25}_{\pm \mathbf{0.06}}$
RN18	Avg Gap	0.0000	0.1375	0.0975	0.0925	0.2650	0.0750	0.0825	0.1075	0.1800	0.0350	**0.0300**
	JSD $\times 1e4$	$0.00_{\pm 0.00}$	$1.03_{\pm 0.24}$	$1.06_{\pm 0.21}$	$1.00_{\pm 0.26}$	$2.39_{\pm 2.03}$	$0.41_{\pm 0.09}$	$0.82_{\pm 0.57}$	$0.65_{\pm 0.05}$	$1.09_{\pm 0.05}$	$\mathbf{0.32}_{\pm \mathbf{0.04}}$	$\mathbf{0.32}_{\pm \mathbf{0.03}}$
	Time (min.)	$5.32_{\pm 1.18}$	$0.43_{\pm 0.00}$	$0.49_{\pm 0.00}$	$0.57_{\pm 0.00}$	$0.33_{\pm 0.00}$	$0.58_{\pm 0.00}$	$0.54_{\pm 0.00}$	$0.45_{\pm 0.00}$	$1.56_{\pm 0.01}$	$\mathbf{0.29}_{\pm \mathbf{0.00}}$	$0.30_{\pm 0.01}$

Table 2. **Unlearning 10% of CIFAR-10**. LoTUS outperforms state-of-the-art approaches, both when the calibration/unseen set D_u consists of **real data** (●) and when it consists of **synthetic data** (●) from the CIFAKE dataset. We highlight the **best** and second-best scores.

5. Results & Discussion

Unlearning Effectiveness was assessed using the Avg Gap and JSD scores. Avg Gap incorporates knowledge from the MIA accuracy and the model accuracies on the forget, retain and test sets; thus it indicates the balance of forgetting/retention. JSD evaluates the unlearning effectiveness and the resilience to the Streisand effect. As shown in Tabs. 1 to 3, LoTUS outperforms state-of-the-art methods in balancing forgetting/retention, unlearning effectiveness, and resilience to the Streisand effect. As shown in Tab. 1, MUFAC & ResNet18 is the only benchmark where LoTUS succeeds the second-best and not the best Avg Gap, however MUFAC is a challenging dataset as seen by the increased JSD scores accross all methods compared to other datasets. This may derive from the increased similarity of images in the retain and forget sets, as presented in Sec. 13. Regarding the assumption of distributional similarity between the forget and unseen sets, in Tab. 2, we demonstrate that it can be relaxed by showing that LoTUS is still the best-performing method even when the unseen set consists of AI-generated synthetic data from CIFAKE [2]. Another intriguing finding is that across all datasets and models, LoTUS consistently achieves the highest JSD score.

The JSD metric provides a more sensitive measure of unlearning effectiveness than model's accuracy on the forget set Acc_f, enabling it to capture unlearning misconceptions that may lead to the Streisand effect. Specifically, JSD evaluates shifts in output distributions, while Acc_f considers only the predicted class. In Machine Unlearning applications, the accuracy of the pre-trained model on the forget set is typically higher than that of the gold standard model. Thus, Acc_f is commonly used to assess whether unlearning reduces the pre-trained model's accuracy to align with the gold standard. However, as emphasized by Chundawat *et al*. [9], misclassification alone does not imply successful unlearning. They highlight a strawman unlearning solution where predictions on the forget set are maximally incorrect (*e.g*., a *cat* is classified into the *airplane* class with increased confidence), arguing that this undermines the generalization capacity of the model and increases the risk of the Streisand effect –making the forget samples more noticeable to attackers. The JSD score penalizes these maximally wrong predictions, while accuracy on the forget set Acc_f does

		Metric (↓)	BadT	SCRUB	SSD	UNSIR	SalUn	**LoTUS**
Vision Transformer	C-100	Avg. Gap	0.0575	0.0350	0.0350	0.0375	0.0375	**0.0225**
		JSD $\times 1e4$	$0.06_{\pm 0.01}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$	$0.02_{\pm 0.00}$	$0.10_{\pm 0.01}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$
		Time (min)	$15.04_{\pm 0.03}$	$16.82_{\pm 0.03}$	$18.69_{\pm 0.06}$	$18.33_{\pm 0.02}$	$38.08_{\pm 0.02}$	$\mathbf{13.79}_{\pm \mathbf{0.02}}$
	C-10	Avg. Gap	0.0600	0.0125	0.0150	0.0150	**0.0050**	**0.0050**
		JSD $\times 1e4$	$0.04_{\pm 0.01}$	$\mathbf{0.00}_{\pm \mathbf{0.00}}$	$\mathbf{0.00}_{\pm \mathbf{0.00}}$	$\mathbf{0.00}_{\pm \mathbf{0.00}}$	$0.02_{\pm 0.00}$	$\mathbf{0.00}_{\pm \mathbf{0.00}}$
		Time (min)	$15.10_{\pm 0.20}$	$16.99_{\pm 0.35}$	$19.03_{\pm 0.54}$	$18.33_{\pm 0.02}$	$37.93_{\pm 0.18}$	$\mathbf{14.09}_{\pm \mathbf{0.53}}$
ResNet18	C-100	Avg. Gap	0.3050	0.2225	0.2225	0.2925	0.3300	**0.1725**
		JSD $\times 1e4$	$0.55_{\pm 0.04}$	$0.44_{\pm 0.02}$	$0.44_{\pm 0.02}$	$0.65_{\pm 0.23}$	$6.25_{\pm 0.45}$	$\mathbf{0.28}_{\pm \mathbf{0.00}}$
		Time (min)	$0.58_{\pm 0.01}$	$0.62_{\pm 0.00}$	$1.29_{\pm 0.03}$	$0.72_{\pm 0.01}$	$1.49_{\pm 0.01}$	$\mathbf{0.57}_{\pm \mathbf{0.01}}$
	C-10	Avg. Gap	**0.0625**	0.1075	0.1025	0.1025	0.1300	0.0650
		JSD $\times 1e4$	$0.18_{\pm 0.02}$	$0.14_{\pm 0.00}$	$0.13_{\pm 0.00}$	$0.21_{\pm 0.05}$	$1.40_{\pm 0.02}$	$\mathbf{0.09}_{\pm \mathbf{0.01}}$
		Time (min)	$\mathbf{0.57}_{\pm \mathbf{0.02}}$	$0.61_{\pm 0.02}$	$1.27_{\pm 0.02}$	$0.73_{\pm 0.00}$	$1.49_{\pm 0.01}$	$\mathbf{0.57}_{\pm \mathbf{0.00}}$

Table 3. **Scaling up the Forget set to 50% of the training sets:** LoTUS outperforms state-of-the-art approaches in all metrics. Basic unlearning methods (Finetuning, NegGrad+, Rnd Labeling) are more efficient, but less effective than LoTUS.

Method	RF-JSD $\times 1e4$ (↓)	Time (↓)	Retain Acc.	MIA Acc.
Original	$1.22_{\pm 0.01}$	(pre-trained)	$0.94_{\pm 0.00}$	$0.71_{\pm 0.00}$
Finetuning	$2.22_{\pm 0.02}$	$16.24_{\pm 0.03}$	$0.97_{\pm 0.00}$	$0.78_{\pm 0.00}$
NegGrad+	$2.17_{\pm 0.02}$	$18.10_{\pm 0.03}$	$0.97_{\pm 0.00}$	$0.80_{\pm 0.00}$
Rnd Labeling	$1.80_{\pm 0.09}$	$19.37_{\pm 0.03}$	$0.95_{\pm 0.01}$	$0.74_{\pm 0.01}$
Bad Teacher	$3.16_{\pm 3.25}$	$11.66_{\pm 0.03}$	$0.77_{\pm 0.21}$	$0.52_{\pm 0.18}$
SCRUB	$1.24_{\pm 0.01}$	$24.49_{\pm 0.03}$	$0.94_{\pm 0.00}$	$0.71_{\pm 0.00}$
SSD	$1.23_{\pm 0.01}$	$22.61_{\pm 0.10}$	$0.94_{\pm 0.00}$	$0.71_{\pm 0.00}$
UNSIR	$2.54_{\pm 0.03}$	$33.12_{\pm 0.03}$	$0.99_{\pm 0.00}$	$0.77_{\pm 0.01}$
SalUn	$1.83_{\pm 0.03}$	$59.27_{\pm 0.37}$	$0.95_{\pm 0.00}$	$0.74_{\pm 0.01}$
LoTUS	$\mathbf{1.11}_{\pm \mathbf{0.01}}$	$\mathbf{10.72}_{\pm \mathbf{0.01}}$	$0.94_{\pm 0.00}$	$0.61_{\pm 0.01}$

Table 4. **Large-Scale Unlearning with ImageNet1k:** LoTUS outperforms state-of-the-art approaches in both unlearning effectiveness (RF-JSD) and efficiency (Time). While other metrics lack concrete validation due to the absence of a Gold Standard, they provide additional insights: LoTUS uniquely preserves the Retain Accuracy of the pre-trained model while reducing MIA Accuracy.

not. Therefore, JSD captures both unlearning effectiveness and the vulnerability to the Streisand effect, while Acc_f may present misleading results. In Sec. 9, we provide an extended analysis on how LoTUS succeeds effective unlearning on the JSD, while maintaining high accuracy even on the forget set. Also in Sec. 16, we examine the Streisand effect using an entropy-based approach as in [15].

Unlearning Efficiency was assessed based on the execution time of each algorithm. As shown in Tabs. 1 to 4, LoTUS consistently outperforms the state-of-the-art approaches in terms of unlearning efficiency. The time complexity of unlearning methods can be analyzed in terms of two factors: the complexity of model updates and the complexity of the auxiliary computations (such as τ_d in Eq. (11)). With respect to the time complexity of model updates, the main advantage of LoTUS over Finetuning, NegGrad+, Rnd Labeling, and SCRUB is that LoTUS can preserve the model's utility using only a small percentage of retain samples, while others cannot. Considering the remaining approaches, LoTUS is more efficient mainly because it is the only one with auxiliary computations of linear complexity. A detailed analysis is presented in Sec. 11.

Large-scale unlearning on ImageNet1k. We consider an experimental setup that includes a ViT trained on ImageNet1k ($\sim$1.2M training samples) and data access constraints that define a retain set of $45,000$ samples and forget/validation/test sets of $5,000$ samples each. The size of the ImageNet1k dataset deters retraining the model entirely to effectively unlearn the forget samples. Furthermore, when the original training dataset is not fully accessible, retraining is infeasible. This leaves Machine Unlearning as the only viable solution for removing the influence of the forget samples from the pre-trained model. Moreover, since the gold standard model is not available, the established Avg Gap and JSD metrics cannot be used. To address this, we use the RF-JSD evaluation metric, which does not require the retrained model, and has been proved to have a strong correlation with the established JSD metric. As shown in Tab. 7, the mean Pearson correlation coefficient (PCC) of JSD and RF-JSD is $0.92_{\pm 0.04}$ (p-value: 0.001). As shown in Tab. 4, LoTUS outperforms state-of-the-art approaches in terms of both unlearning effectiveness and efficiency.

6. Conclusions

We introduced an information-theoretic framework for unlearning and proposed LoTUS, a novel method that removes the influence of specific training samples from a pre-trained model while preserving its utility on the remaining data. We demonstrated how the dynamic temperature parameter and the introduction of Gumbel noise in the activation function enable LoTUS to smooth output probabilities for forget samples, mitigating over-confident predictions that stem from data memorization.

We introduced the RF-JSD metric, which strongly correlates with the established JSD metric but eliminates the need for a retrained model, making it particularly valuable for unlearning in large-scale datasets, where retraining is impractical, or in settings with restricted data access. We compared it with the existing ZRF score, showing that RF-JSD offers greater interpretability and efficiency. Moreover, we highlighted that the established Avg Gap metric can produce misleading results and emphasized the increased sensitivity of JSD, which enables it to capture unlearning misconceptions that Avg Gap fails to detect.

We demonstrated that LoTUS surpasses state-of-the-art methods in both effectiveness and efficiency, demonstrating its scalability and adaptability to large-scale unlearning challenges and stringent data constraints.

Limitations. Both our theoretical framework and extensive experiments demonstrate that LoTUS surpasses state-of-the-art performance in instance-wise unlearning. While Sec. 14 shows that our theoretical framework extends to class unlearning and LoTUS can be adapted for this task, our experimental setup in class unlearning is less extensive.

Acknowledgments

This work was partially supported by the EU funded project ATLANTIS (Grant Agreement Number 101073909).

References

[1] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR, 2018. 3, 4

[2] Jordan J Bird and Ahmad Lotfi. Cifake: Image classification and explainable identification of ai-generated synthetic images. *IEEE Access*, 12:15642–15650, 2024. 6, 7

[3] 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)*, pages 141–159. IEEE, 2021. 1, 2

[4] Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pages 463–480. IEEE, 2015. 1, 2

[5] 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*, pages 11186–11194, 2024. 2, 5

[6] 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*, pages 7766–7775, 2023. 2

[7] Ruizhe Chen, Jianfei Yang, Huimin Xiong, Jianhong Bai, Tianxiang Hu, Jin Hao, Yang Feng, Joey Tianyi Zhou, Jian Wu, and Zuozhu Liu. Fast model debias with machine unlearning. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 6

[8] Dasol Choi and Dongbin Na. Towards machine unlearning benchmarks: Forgetting the personal identities in facial recognition systems. *arXiv preprint arXiv:2311.02240*, 2023. 6

[9] Vikram S Chundawat, Ayush K Tarun, Murari Mandal, and Mohan Kankanhalli. Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7210–7217, 2023. 2, 5, 6, 7, 1, 3

[10] Thomas Cover and Joy Thomas. *Elements of information theory*. Wiley-Interscience, 2nd edition, 2012. 3, 4

[11] 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. In *International Conference on Learning Representations*, 2020. 6

[12] Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 6, 7, 3

[13] Jack Foster, Kyle Fogarty, Stefan Schoepf, Cengiz Öztireli, and Alexandra Brintrup. An information theoretic approach to machine unlearning, 2024. 1, 2

[14] Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12043–12051, 2024. 1, 2, 6, 7, 3

[15] 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*, pages 9304–9312, 2020. 1, 2, 8, 5

[16] Aditya Golatkar, Alessandro Achille, Luca Zancato, Yu-Xiang Wang, Ashwin Swaminathan, and Stefano Soatto. Cpr: Retrieval augmented generation for copyright protection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12374–12384, 2024. 2

[17] Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11516–11524, 2021. 2, 3, 7, 1, 5

[18] Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. Certified data removal from machine learning models. In *International Conference on Machine Learning*, pages 3832–3842. PMLR, 2020. 2

[19] 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*, pages 770–778, 2016. 6

[20] Alvin Heng and Harold Soh. Selective amnesia: A continual learning approach to forgetting in deep generative models. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical Report*, 2009. 6

[22] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1): 79–86, 1951. 6

[23] Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36, 2024. 2, 5, 6, 7, 1

[24] Yann Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 6

[25] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. 6

[26] Sijia Liu, Yang Liu, Nathalie Baracaldo Angel, and Eleni Triantafillou. Machine unlearning in computer vision: Foundations and applications. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[27] Youyang Qu, Xin Yuan, Ming Ding, Wei Ni, Thierry Rakotoarivelo, and David Smith. Learn to unlearn: Insights into machine unlearning. *Computer*, 57(3):79–90, 2024. 6

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. 6

[29] Stefan Schoepf, Jack Foster, and Alexandra Brintrup. Potion: Towards poison unlearning. *arXiv preprint arXiv:2406.09173*, 2024. 2

[30] Juwon Seo, Sung-Hoon Lee, Tae-Young Lee, Seungjun Moon, and Gyeong-Moon Park. Generative unlearning for any identity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9151–9161, 2024. 2

[31] Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948. 2

[32] 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)*, pages 3–18. IEEE, 2017. 3

[33] Reza Shokri, Martin Strobel, and Yair Zick. On the privacy risks of model explanations. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 231–241, 2021. 1

[34] Ayush K Tarun, Vikram S Chundawat, Murari Mandal, and Mohan Kankanhalli. Fast yet effective machine unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, 2023. 6, 7, 1, 2, 3

[35] Eleni Triantafillou, Peter Kairouz, Fabian Pedregosa, Jamie Hayes, Meghdad Kurmanji, Kairan Zhao, Vincent Dumoulin, Julio Jacques Junior, Ioannis Mitliagkas, Jun Wan, et al. Are we making progress in unlearning? findings from the first neurips unlearning competition. *arXiv preprint arXiv:2406.09073*, 2024. 2

[36] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2020. 4

[37] Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S. Yu. Machine unlearning: A survey. *ACM Comput. Surv.*, 56(1), 2023. 2

[38] Jiayuan Ye, Anastasia Borovykh, Soufiane Hayou, and Reza Shokri. Leave-one-out distinguishability in machine learning. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 3

LoTUS: Large-Scale Machine Unlearning with a Taste of Uncertainty

Supplementary Material

7. Baselines

The **Gold Stadnard (Gold Std)** model is retrained entirely only on the retain set (D_r), achieving ideal unlearning –when access to the full training set is guaranteed– but at the cost of increased computational complexity. **Fine-tuning**: The pre-trained model is further trained only on the retain samples (D_r). **NegGrad+** [23]: The pre-trained model continues training on the full training set, but the gradient sign is reversed during backpropagation for the forget samples. **Random Labeling (RndLbl)** [17]: The pre-trained model continues training on the full training set, but the forget samples are randomly reassigned to incorrect classes. **Bad Teacher (BadT)** [9]: A knowledge distillation framework where the student model follows the pre-trained model for retain samples and a randomly initialized model for forget samples. **SCRUB** [23]: A knowledge distillation framework where student model selectively aligns with the pre-trained model by minimizing the KL divergence of their outputs on retain samples while maximizing it for forget samples. **SSD** [14]: Weights that are disproportionately important for forget samples are identified using the Fisher Information Matrix and subsequently dampened. **UNSIR** [34]: A noise matrix, generated based on the forget samples, is fed to the pre-trained model to maximize its error on these samples. **SalUn** [12]: A gradient-based approach that identifies weights to be unlearned and those to keep unchanged, followed by a downstream unlearning method such as Random Labeling. Finetuning, NegGrad+ and Random Labeling are considered simple yet widely used unlearning baselines, whereas the latter five are state-of-the-art approaches.

LoTUS can be integrated with SalUn, with SalUn used to obtain the weight saliency mask for pruning, and LoTUS applied for unlearning. This integration can enhance the unlearning effectiveness of LoTUS. For instance, on ResNet18 with TinyImageNet, it reduces the Avg Gap of LoTUS to 0.1250 (a 25.37% decrease) and the JSD to 0.55 (an 11.29% decrease). However, this comes at the cost of efficiency, with unlearning time increasing to 4.62 minutes (a 162% increase).

8. Reproducibility and Transparency

The code to reproduce the results presented in this paper is publicly available at https://github.com/cspartalis/LoTUS. In addition, all tables and figures have been documented in Jupyter notebooks to enhance transparency. We conducted the experiments using Python 3.11 and CUDA 12.1. For ImageNet1k experiments, we used an NVIDIA RTX A6000 48GB GPU. The remaining experiments were performed on an NVIDIA RTX 4080 16GB GPU. We also used an Intel i7-12700K CPU and 32GB RAM. The hyperparameters used for the baselines are listed in Tab. 5

Baseline	Learning Rate	Weight Decay	Optimizer
Finetune	1×10^{-3}	5×10^{-4}	SGD
Negrad+	1×10^{-3}	5×10^{-4}	SGD
RndLbl	1×10^{-3}	5×10^{-4}	SGD
BadT	1×10^{-4}	0	Adam
SCRUB	5×10^{-4}	5×10^{-4}	Adam
SSD	0.1	0	SGD
UNSIR	1×10^{-3}	0	SGD
SalUn	0.1	5×10^{-4}	SGD

Table 5. Hyperparameters used for baselines. For state-of-the-art methods, they are taken from their respective papers.

9. Extended Analysis on the Accuracy Metrics

Table 6 presents the accuracy scores that define the Avg Gap metric. Beyond outperforming state-of-the-art methods in terms of Avg Gap, LoTUS achieves the best scores in individual accuracy metrics, including MIA accuracy, and accuracy on the retain and test sets. Specifically, it consistently ranks either first or second in these metrics, with first place being the most frequent.

Regarding retention performance (*i.e.*, preserving the utility of the pre-trained model), LoTUS clearly outperforms state-of-the-art, as evidenced by its superior accuracy on the retain and test sets.

However, evaluating unlearning effectiveness, requires a more nuanced analysis. Although LoTUS consistently ranks among the top two methods in MIA accuracy, its accuracy on the forget set exceeds that of the gold standard model (*i.e.*, the model retrained solely on the retain set). This apparent discrepancy may lead to misleading evaluation, suggesting that LoTUS exhibits poor unlearning performance.

However, by incorporating the more sensitive JSD metric –a measure that captures distributional-level differences and provides a more robust evaluation, as detailed in Sec. 5– we conclude that LoTUS achieves effective unlearning. Given this, the increased accuracy on the forget set does not indicate poor unlearning, but rather suggests that LoTUS preserves the utility of the pre-trained model even for the forget samples. The fact that LoTUS achieves the best Avg Gap scores despite the disproportionate penalty imposed by the gap between the accuracy of the unlearned and gold standard models on forget samples

		Metric (↓)	Gold Std	Finetuning	NegGrad+ [23]	RndLbl [17]	Bad Teacher [9]	SCRUB [23]	SSD [14]	UNSIR [34]	SalUn [12]	**LoTUS**
Vision Transformer (ViT)	TinyImageNet	MIA Acc.	$0.76_{\pm 0.00}$	$0.78_{\pm 0.00}$(0.02)	$0.83_{\pm 0.00}$(0.07)	$0.50_{\pm 0.43}$(0.26)	$0.67_{\pm 0.00}$(0.09)	$0.79_{\pm 0.00}$(0.03)	$0.79_{\pm 0.00}$(0.03)	$0.80_{\pm 0.00}$(0.04)	$0.67_{\pm 0.25}$(0.09)	$0.76_{\pm 0.00}$**(0.00)**
		Forget Acc.	$0.90_{\pm 0.00}$	$0.93_{\pm 0.00}$(0.03)	$0.97_{\pm 0.00}$(0.07)	$0.61_{\pm 0.52}$(0.29)	$0.84_{\pm 0.01}$(0.06)	$0.96_{\pm 0.00}$(0.06)	$0.96_{\pm 0.00}$(0.06)	$0.92_{\pm 0.00}$**(0.02)**	$0.82_{\pm 0.30}$(0.08)	$0.96_{\pm 0.00}$(0.06)
		Retain Acc.	$0.96_{\pm 0.00}$	$0.98_{\pm 0.00}$(0.02)	$0.98_{\pm 0.00}$(0.02)	$0.64_{\pm 0.55}$(0.32)	$0.87_{\pm 0.01}$(0.09)	$0.96_{\pm 0.00}$**(0.00)**	$0.96_{\pm 0.00}$**(0.00)**	$0.94_{\pm 0.00}$(0.02)	$0.86_{\pm 0.32}$(0.10)	$0.96_{\pm 0.00}$**(0.00)**
		Test Acc.	$0.90_{\pm 0.00}$	$0.90_{\pm 0.00}$**(0.00)**	$0.90_{\pm 0.00}$**(0.00)**	$0.60_{\pm 0.51}$(0.30)	$0.83_{\pm 0.01}$(0.07)	$0.90_{\pm 0.00}$**(0.00)**	$0.90_{\pm 0.00}$**(0.00)**	$0.89_{\pm 0.00}$(0.01)	$0.80_{\pm 0.30}$(0.10)	$0.90_{\pm 0.00}$**(0.00)**
		Avg Gap	0.0000	0.0175	0.0400	0.2925	0.0775	0.0225	0.0225	0.0225	0.0925	**0.0150**
	CIFAR-100	MIA Acc.	$0.72_{\pm 0.00}$	$0.77_{\pm 0.00}$(0.05)	$0.79_{\pm 0.02}$(0.07)	$0.74_{\pm 0.01}$(0.02)	$0.66_{\pm 0.01}$(0.06)	$0.75_{\pm 0.00}$(0.03)	$0.75_{\pm 0.01}$(0.03)	$0.78_{\pm 0.01}$(0.06)	$0.75_{\pm 0.01}$(0.03)	$0.71_{\pm 0.02}$**(0.01)**
		Forget Acc.	$0.92_{\pm 0.00}$	$0.95_{\pm 0.01}$(0.03)	$0.97_{\pm 0.02}$(0.05)	$0.94_{\pm 0.00}$**(0.02)**	$0.90_{\pm 0.01}$**(0.02)**	$0.97_{\pm 0.00}$(0.05)	$0.96_{\pm 0.01}$(0.04)	$0.94_{\pm 0.00}$**(0.02)**	$0.94_{\pm 0.01}$(0.02)	$0.96_{\pm 0.01}$(0.04)
		Retain Acc.	$0.96_{\pm 0.00}$	$0.98_{\pm 0.00}$(0.02)	$0.97_{\pm 0.02}$(0.01)	$0.98_{\pm 0.00}$(0.02)	$0.91_{\pm 0.00}$(0.05)	$0.96_{\pm 0.00}$**(0.00)**	$0.96_{\pm 0.00}$**(0.00)**	$0.95_{\pm 0.01}$(0.01)	$0.98_{\pm 0.01}$(0.02)	$0.96_{\pm 0.00}$**(0.00)**
		Test Acc.	$0.91_{\pm 0.01}$	$0.92_{\pm 0.01}$(0.01)	$0.91_{\pm 0.01}$**(0.00)**	$0.92_{\pm 0.00}$(0.01)	$0.89_{\pm 0.01}$(0.02)	$0.91_{\pm 0.01}$**(0.00)**	$0.91_{\pm 0.00}$**(0.00)**	$0.90_{\pm 0.01}$(0.01)	$0.92_{\pm 0.00}$(0.01)	$0.91_{\pm 0.00}$**(0.00)**
		Avg Gap	0.0000	0.0275	0.0325	0.0175	0.0375	0.0200	0.0175	0.0250	0.0200	**0.0125**
	CIFAR-10	MIA Acc.	$0.88_{\pm 0.00}$	$0.90_{\pm 0.00}$(0.02)	$0.91_{\pm 0.00}$(0.03)	$0.84_{\pm 0.02}$(0.04)	$0.81_{\pm 0.02}$(0.07)	$0.88_{\pm 0.00}$**(0.00)**	$0.89_{\pm 0.01}$(0.01)	$0.90_{\pm 0.00}$(0.02)	$0.84_{\pm 0.02}$(0.04)	$0.87_{\pm 0.00}$(0.01)
		Forget Acc.	$0.99_{\pm 0.00}$	$0.99_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.00}$(0.01)	$0.99_{\pm 0.00}$**(0.00)**	$0.96_{\pm 0.01}$(0.03)	$1.00_{\pm 0.00}$(0.01)	$1.00_{\pm 0.01}$(0.01)	$0.99_{\pm 0.00}$**(0.00)**	$0.99_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.00}$(0.01)
		Retain Acc.	$1.00_{\pm 0.00}$	$1.00_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.00}$**(0.00)**	$0.97_{\pm 0.01}$(0.03)	$1.00_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.01}$**(0.00)**	$0.99_{\pm 0.00}$(0.01)	$1.00_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.00}$**(0.00)**
		Test Acc.	$0.98_{\pm 0.01}$	$0.99_{\pm 0.01}$(0.01)	$0.99_{\pm 0.01}$(0.01)	$0.99_{\pm 0.00}$(0.01)	$0.96_{\pm 0.01}$(0.02)	$0.99_{\pm 0.01}$(0.01)	$0.99_{\pm 0.01}$(0.01)	$0.99_{\pm 0.00}$(0.01)	$0.99_{\pm 0.01}$(0.01)	$0.98_{\pm 0.01}$**(0.00)**
		Avg Gap	0.0000	0.0075	0.0125	0.0125	0.0375	**0.0050**	0.0075	0.0100	0.0125	**0.0050**
	MUFAC	MIA Acc.	$0.57_{\pm 0.00}$	$0.52_{\pm 0.08}$(0.05)	$0.52_{\pm 0.07}$(0.05)	$0.52_{\pm 0.10}$(0.05)	$0.35_{\pm 0.05}$(0.22)	$0.59_{\pm 0.01}$**(0.02)**	$0.59_{\pm 0.01}$**(0.02)**	$0.47_{\pm 0.08}$(0.10)	$0.53_{\pm 0.12}$(0.04)	$0.59_{\pm 0.01}$**(0.02)**
		Forget Acc.	$0.57_{\pm 0.01}$	$0.61_{\pm 0.01}$(0.04)	$0.66_{\pm 0.02}$(0.09)	$0.58_{\pm 0.01}$**(0.01)**	$0.43_{\pm 0.06}$(0.14)	$0.62_{\pm 0.01}$(0.05)	$0.59_{\pm 0.04}$(0.02)	$0.58_{\pm 0.01}$**(0.01)**	$0.58_{\pm 0.01}$**(0.01)**	$0.63_{\pm 0.00}$(0.06)
		Retain Acc.	$0.66_{\pm 0.01}$	$0.72_{\pm 0.01}$(0.06)	$0.71_{\pm 0.01}$(0.05)	$0.67_{\pm 0.02}$(0.01)	$0.47_{\pm 0.07}$(0.19)	$0.66_{\pm 0.01}$**(0.00)**	$0.63_{\pm 0.04}$(0.03)	$0.72_{\pm 0.01}$(0.06)	$0.68_{\pm 0.01}$(0.02)	$0.66_{\pm 0.01}$**(0.00)**
		Test Acc.	$0.65_{\pm 0.01}$	$0.66_{\pm 0.01}$(0.01)	$0.65_{\pm 0.03}$**(0.00)**	$0.64_{\pm 0.01}$(0.01)	$0.50_{\pm 0.08}$(0.15)	$0.66_{\pm 0.01}$(0.01)	$0.64_{\pm 0.01}$(0.01)	$0.63_{\pm 0.02}$(0.02)	$0.64_{\pm 0.02}$(0.01)	$0.65_{\pm 0.01}$**(0.00)**
		Avg Gap	0.0000	0.0400	0.0475	**0.0200**	0.1750	**0.0200**	**0.0200**	0.0475	**0.0200**	**0.0200**
ResNet18 (RN18)	TinyImageNet	MIA Acc.	$0.30_{\pm 0.01}$	$0.00_{\pm 0.00}$(0.30)	$0.00_{\pm 0.00}$(0.30)	$0.00_{\pm 0.00}$(0.30)	$0.67_{\pm 0.52}$(0.37)	$0.96_{\pm 0.01}$(0.66)	$0.95_{\pm 0.01}$(0.65)	$0.67_{\pm 0.58}$(0.37)	$0.00_{\pm 0.00}$(0.30)	$0.53_{\pm 0.01}$**(0.23)**
		Forget Acc.	$0.58_{\pm 0.00}$	$0.70_{\pm 0.02}$(0.12)	$0.73_{\pm 0.02}$(0.15)	$0.56_{\pm 0.02}$**(0.02)**	$0.49_{\pm 0.04}$(0.09)	$1.00_{\pm 0.00}$(0.42)	$1.00_{\pm 0.00}$(0.42)	$0.68_{\pm 0.03}$(0.10)	$0.62_{\pm 0.01}$(0.04)	$0.91_{\pm 0.01}$(0.33)
		Retain Acc.	$1.00_{\pm 0.00}$	$0.73_{\pm 0.02}$(0.27)	$0.73_{\pm 0.02}$(0.27)	$0.73_{\pm 0.02}$(0.27)	$0.55_{\pm 0.04}$(0.45)	$1.00_{\pm 0.00}$**(0.00)**	$1.00_{\pm 0.00}$**(0.00)**	$0.71_{\pm 0.02}$(0.29)	$0.71_{\pm 0.02}$(0.29)	$0.93_{\pm 0.01}$(0.07)
		Test Acc.	$0.89_{\pm 0.01}$	$0.40_{\pm 0.01}$(0.19)	$0.41_{\pm 0.01}$(0.18)	$0.41_{\pm 0.02}$(0.18)	$0.36_{\pm 0.03}$(0.23)	$0.60_{\pm 0.01}$**(0.01)**	$0.60_{\pm 0.00}$**(0.01)**	$0.40_{\pm 0.02}$(0.19)	$0.41_{\pm 0.01}$(0.18)	$0.55_{\pm 0.00}$(0.04)
		Avg Gap	0.0000	0.2200	0.2250	0.1925	0.2850	0.2725	0.2700	0.2375	0.2025	**0.1675**
	CIFAR-100	MIA Acc.	$0.49_{\pm 0.01}$	$0.00_{\pm 0.00}$(0.49)	$0.00_{\pm 0.00}$(0.49)	$0.00_{\pm 0.00}$(0.49)	$0.33_{\pm 0.58}$(0.16)	$0.78_{\pm 0.05}$(0.29)	$0.59_{\pm 0.05}$(0.10)	$0.00_{\pm 0.00}$(0.49)	$0.00_{\pm 0.00}$(0.49)	$0.28_{\pm 0.22}$**(0.21)**
		Forget Acc.	$0.57_{\pm 0.02}$	$0.40_{\pm 0.06}$(0.17)	$0.41_{\pm 0.06}$(0.16)	$0.31_{\pm 0.06}$(0.26)	$0.27_{\pm 0.03}$(0.30)	$0.93_{\pm 0.03}$(0.36)	$0.50_{\pm 0.32}$**(0.07)**	$0.40_{\pm 0.07}$(0.17)	$0.38_{\pm 0.04}$(0.19)	$0.81_{\pm 0.08}$(0.24)
		Retain Acc.	$0.94_{\pm 0.03}$	$0.41_{\pm 0.06}$(0.53)	$0.41_{\pm 0.06}$(0.53)	$0.37_{\pm 0.07}$(0.57)	$0.28_{\pm 0.03}$(0.66)	$0.93_{\pm 0.03}$**(0.01)**	$0.50_{\pm 0.32}$(0.44)	$0.41_{\pm 0.07}$(0.53)	$0.41_{\pm 0.04}$(0.53)	$0.92_{\pm 0.02}$**(0.02)**
		Test Acc.	$0.60_{\pm 0.02}$	$0.35_{\pm 0.05}$(0.25)	$0.35_{\pm 0.05}$(0.25)	$0.31_{\pm 0.06}$(0.29)	$0.25_{\pm 0.03}$(0.35)	$0.60_{\pm 0.02}$**(0.00)**	$0.36_{\pm 0.20}$(0.24)	$0.34_{\pm 0.04}$(0.26)	$0.35_{\pm 0.03}$(0.25)	$0.61_{\pm 0.01}$(0.01)
		Avg Gap	0.0000	0.3600	0.3575	0.4025	0.3675	0.1650	0.2125	0.3625	0.3650	**0.1200**
	CIFAR-10	MIA Acc.	$0.76_{\pm 0.03}$	$0.30_{\pm 0.26}$(0.46)	$0.48_{\pm 0.50}$(0.28)	$0.48_{\pm 0.50}$(0.28)	$0.43_{\pm 0.37}$(0.33)	$0.94_{\pm 0.01}$(0.18)	$0.81_{\pm 0.11}$**(0.05)**	$0.46_{\pm 0.03}$(0.30)	$0.16_{\pm 0.28}$(0.60)	$0.82_{\pm 0.10}$(0.06)
		Forget Acc.	$0.91_{\pm 0.02}$	$0.97_{\pm 0.01}$(0.06)	$0.97_{\pm 0.01}$(0.06)	$0.96_{\pm 0.01}$(0.05)	$0.71_{\pm 0.18}$(0.20)	$1.00_{\pm 0.00}$(0.09)	$0.86_{\pm 0.16}$(0.05)	$0.93_{\pm 0.01}$**(0.02)**	$0.94_{\pm 0.02}$**(0.02)**	$0.99_{\pm 0.00}$(0.08)
		Retain Acc.	$0.99_{\pm 0.02}$	$0.98_{\pm 0.01}$(0.01)	$0.97_{\pm 0.01}$(0.02)	$0.97_{\pm 0.01}$(0.02)	$0.71_{\pm 0.18}$(0.28)	$1.00_{\pm 0.00}$(0.01)	$0.87_{\pm 0.16}$(0.12)	$0.93_{\pm 0.01}$(0.06)	$0.95_{\pm 0.02}$(0.04)	$0.99_{\pm 0.00}$**(0.00)**
		Test Acc.	$0.91_{\pm 0.02}$	$0.89_{\pm 0.02}$(0.02)	$0.88_{\pm 0.02}$(0.03)	$0.89_{\pm 0.02}$(0.02)	$0.66_{\pm 0.16}$(0.25)	$0.93_{\pm 0.01}$(0.02)	$0.80_{\pm 0.15}$(0.11)	$0.86_{\pm 0.01}$(0.05)	$0.86_{\pm 0.03}$(0.05)	$0.91_{\pm 0.01}$**(0.00)**
		Avg Gap	0.0000	0.1375	0.0975	0.0925	0.2650	0.0750	0.0825	0.1075	0.1800	**0.0350**
	MUFAC	MIA Acc.	$0.48_{\pm 0.04}$	$0.54_{\pm 0.09}$(0.06)	$0.53_{\pm 0.08}$**(0.05)**	$0.33_{\pm 0.31}$(0.15)	$0.34_{\pm 0.01}$(0.14)	$0.70_{\pm 0.05}$(0.22)	$0.70_{\pm 0.06}$(0.22)	$0.40_{\pm 0.35}$(0.08)	$0.53_{\pm 0.08}$**(0.05)**	$0.53_{\pm 0.04}$**(0.05)**
		Forget Acc.	$0.47_{\pm 0.04}$	$0.64_{\pm 0.04}$(0.17)	$0.68_{\pm 0.04}$(0.21)	$0.66_{\pm 0.04}$(0.19)	$0.53_{\pm 0.07}$**(0.06)**	$0.88_{\pm 0.06}$(0.41)	$0.87_{\pm 0.06}$(0.40)	$0.71_{\pm 0.03}$(0.24)	$0.63_{\pm 0.05}$(0.16)	$0.86_{\pm 0.04}$(0.39)
		Retain Acc.	$0.89_{\pm 0.04}$	$0.64_{\pm 0.04}$(0.25)	$0.66_{\pm 0.03}$(0.23)	$0.80_{\pm 0.03}$(0.09)	$0.76_{\pm 0.04}$(0.13)	$0.89_{\pm 0.04}$**(0.00)**	$0.89_{\pm 0.05}$**(0.00)**	$0.73_{\pm 0.03}$(0.16)	$0.67_{\pm 0.04}$(0.22)	$0.85_{\pm 0.08}$(0.04)
		Test Acc.	$0.56_{\pm 0.02}$	$0.43_{\pm 0.01}$(0.13)	$0.43_{\pm 0.01}$(0.13)	$0.47_{\pm 0.02}$(0.09)	$0.48_{\pm 0.03}$(0.08)	$0.54_{\pm 0.03}$**(0.02)**	$0.54_{\pm 0.03}$**(0.02)**	$0.46_{\pm 0.01}$(0.10)	$0.43_{\pm 0.02}$(0.13)	$0.54_{\pm 0.05}$**(0.02)**
		Avg Gap	0.0000	0.1525	0.1550	0.1300	**0.1025**	0.1625	0.1600	0.1450	0.1400	0.1250

Table 6. **Accuracy Metrics used to compute Average (Avg) Gap**. Mean performance and standard deviation ($\mu \pm \sigma$) are reported across three trials with different forget and retain sets. Performance gaps relative to the Gold Standard are noted as (•), with smaller gaps indicating stronger performance. Avg Gap serves as a key indicator, summarizing performance across MIA, Forget, Retain, and Test Accuracy. LoTUS achieves state-of-the-art results in MIA, retain and test accuracies, ranking as the best in most cases and second-best in the remaining.

further reinforces its capacity to balance forgetting and retention, as evidenced by Avg Gap.

This also raises concerns about the widely used Avg Gap metric, as it may lead to misleading evaluation of unlearning. However, incorporating both Avg Gap and JSD metrics in the evaluation helps mitigate these concerns.

10. Detailed Comparison of RF-JSD and ZRF

The ZRF metric [9] assesses the unlearning effectiveness by computing the JSD score twice: once between the unlearned and a randomly initialized model, and again between the pre-trained and the same randomly initialized model. The latter serves as a reference point for the optimal value.

By contrast, RF-JSD simplifies the evaluation by requiring only a single JSD computation –between the unlearned model and the original model– where the optimal value is fixed at zero. This direct alignment with the JSD metric (which also has an optimal value fixed at zero) facilitates a more comprehensive evaluation of the unlearning effectiveness.

Beyond the obvious efficiency gain from RF-JSD not requiring inference on an additional randomly initialized model to obtain a reference score –unlike ZRF– its use of normalized class-wise mean distributions further enhances computational efficiency. Specifically, this reduces the complexity from $O(n_f \cdot n_u \cdot k)$ to $O\big((n_f + n_u) \cdot k\big)$, where n_f and n_u denote the number of samples in the forget and test sets, respectively, and k is the number of classes. This optimization significantly reduces the computational overhead, particularly for large datasets. In this analysis, we exclude the complexity of the feed-forward process, which remains unchanged.

Finally, Table 7 presents a detailed correlation between RF-JSD and JSD as measured by the Pearson correlation coefficient (PCC) for all benchmarks. PCC results exhibit a strong correlation between these two metrics, with RF-JSD offering the additional advantage of not requiring a retrained model (*i.e.*, gold standard).

	Dataset $\left(\frac{\text{num. of forget samples}}{\text{num. of training samples}} \times 100\%\right)$	PCC (↑)	p-value (↓)
ViT	CIFAR-100 (10%)	0.84	0.0043
	CIFAR-10 (10%)	0.92	0.0005
	MUFAC	0.93	0.0003
	CIFAR-100 (50%)	0.94	0.0001
	CIFAR-10 (50%)	0.99	0.0000
ResNet18	CIFAR-100 (10%)	0.97	0.0000
	CIFAR-10 (10%)	0.90	0.0011
	MUFAC	0.88	0.0018
	CIFAR-100 (50%)	0.91	0.0006
	CIFAR-10 (50%)	0.89	0.0013
	Mean ± Std	$0.92_{\pm 0.04}$	$0.0010_{\pm 0.0016}$

Table 7. **Retrain Free-JSD (RF-JSD) and JSD Correlation** measured with the Pearson correlation coefficient (PCC). A high PCC (closer to 1) indicates a strong correlation, while a low p-value reflects high confidence in the measurement. The table shows that RF-JSD strongly correlates with the well-established JSD metric across datasets and architectures, demonstrating its reliability as unlearning metric that is particularly useful when the gold standard model is not available (*e.g.*, it is impractical due to high computational complexity or it is infeasible due to not access to the original training set) .

	Metric (↓)	Finetuning	NegGrad+	RndLbl	**LoTUS**
ViT C-100	Avg. Gap	0.0400	0.0600	0.0250	**0.0225**
	JSD ×1e4	$0.02_{\pm 0.00}$	$0.03_{\pm 0.01}$	$\mathbf{0.01}_{\pm \mathbf{0.01}}$	$\mathbf{0.01}_{\pm \mathbf{0.00}}$
	Time (min)	$\mathbf{6.34}_{\pm \mathbf{0.01}}$	$12.68_{\pm 0.02}$	$12.63_{\pm 0.02}$	$13.79_{\pm 0.02}$
ViT C-10	Avg. Gap	0.0125	0.0200	**0.0050**	**0.0050**
	JSD ×1e4	$\mathbf{0.00}_{\pm \mathbf{0.00}}$	$0.01_{\pm 0.00}$	$\mathbf{0.00}_{\pm \mathbf{0.00}}$	$\mathbf{0.00}_{\pm \mathbf{0.00}}$
	Time (min)	$\mathbf{6.48}_{\pm \mathbf{0.27}}$	$12.97_{\pm 0.50}$	$12.60_{\pm 0.03}$	$14.09_{\pm 0.53}$
RN18 C-100	Avg. Gap	0.3200	0.3150	0.3875	**0.1725**
	JSD ×1e4	$1.39_{\pm 0.10}$	$1.38_{\pm 0.08}$	$1.03_{\pm 0.23}$	$\mathbf{0.28}_{\pm \mathbf{0.00}}$
	Time (min)	$\mathbf{0.26}_{\pm \mathbf{0.01}}$	$0.52_{\pm 0.00}$	$0.48_{\pm 0.00}$	$0.57_{\pm 0.01}$
RN18 C-10	Avg. Gap	0.1100	0.1475	0.2100	**0.0650**
	JSD ×1e4	$0.31_{\pm 0.00}$	$0.31_{\pm 0.01}$	$0.73_{\pm 0.22}$	$\mathbf{0.09}_{\pm \mathbf{0.01}}$
	Time (min)	$\mathbf{0.26}_{\pm \mathbf{0.01}}$	$0.51_{\pm 0.02}$	$0.48_{\pm 0.00}$	$0.57_{\pm 0.00}$

Table 8. **Scaling up the Forget set to 50% of the training sets:** LoTUS outperforms basic unlearning methods in unlearning effectiveness, but not in efficiency.

11. Detailed Analysis on the Time Complexity

This section provides an in-depth analysis that demonstrates why LoTUS achieves superior efficiency compared to state-of-the-art approaches, as observed in Tabs. 1, 3 and 4 and discussed in Sec. 5. We define the time complexity of model updates in DNNs, generalized across architectures like ResNet18 and ViT, as follows:

$$O(E \cdot \frac{n_f + n_r}{B} \cdot N_p \cdot N_i) \tag{13}$$

where E represents the total number of epochs, n_f and n_r are the number of instances in D_f (forget set) and D_r (retain set) used during unlearning, respectively, B is the batch size, N_p is the total number of model parameters, and N_i is the input dimensionality. While this definition abstracts away architectural-specific details and optimizations, it provides a meaningful framework for comparing methods on shared benchmarks.

The main advantage of LoTUS over Finetuning, NegGrad+, Random Labeling, and SCRUB is that it requires significantly fewer instances n_r from the retain set D_r. Specifically, LoTUS can use only 30% of the instances in D_r to preserve the utility of the model. All other factors (E, n_f, B, N_p, N_i) are the same for all unlearning baselines in our benchmarks. As shown in Tab. 1, LoTUS achieves superior efficiency.

As the number of instances n_f in the forget set increases, the execution time of LoTUS increases, in alignment with Eq. (13). Thus, in the extreme scenario where 50% of the forget set is designated for unlearning, we observe that the efficiency of Finetuning, NegGrad+, and Random Labeling may exceed that of LoTUS, as shown in Tab. 3. In Tab. 8 we present the scores of these basic unlearning methods that are not presented in Tab. 3, and show that they may be better in terms of efficiency, but LoTUS remains the best in terms of effectiveness.

Next, we compare the time complexity of the auxiliary computations between LoTUS and other unlearning baselines that use equal or fewer samples from the retain set D_r:

- LoTUS: $O(n_f + n_v)$, where n_v is the total number of instances in the validation set, for computing τ_d.
- Bad Teacher [9]: $O\big((n_f + n_r) \cdot k\big)$, where k is the total number of classes, for calculating the $\mathcal{KL}$ divergences between the student and the teacher.
- UNSIR [34]: $O(E_{noise} \cdot n_f \cdot N_i)$, where E_{noise} are the epochs for noise optimization, and N_i represents the total input dimensionality (product of channels, width and height of the images).
- SSD [14]: $O(n_f \cdot N_p^2)$ for computing the Fisher Information Matrix.

In this analysis, we exempt the complexity of the feed-forward process which is the same for all the unlearning methods in our benchmarks. Also, SalUn [12] introduces a computational overhead prior to unlearning due to the computations of the saliency mask for weight pruning. The complexity of this auxiliary computation contributes to the overall complexity of the downstream method used for unlearning (*e.g.*, Random Labeling and LoTUS in our case). Among the unlearning methods, LoTUS is the only one with auxiliary computations of linear complexity.

12. Cleaning the MUFAC Dataset

We identified duplicates within the forget, retain, validation, and test splits of the MUFAC dataset. More critically,

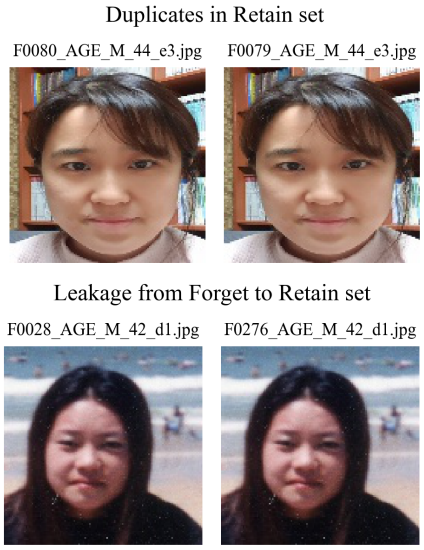


Figure 3. **Duplicates in MUFAC:** An example of a duplicate within the retain set (top) and a critical duplicate shared between the retain and forget set (bottom), which introduces information leakage.

we discovered instances of information leakage across these splits. To address this, we used image hashing to detect identical images with different filenames in these splits, as shown in Fig. 3.

After cleaning MUFAC, the retain set contains $5,513$ samples, and the forget set contains $1,062$ samples. We provide the code for identifying duplicate images and cleaning MUFAC in https://github.com/cspartalis/LoTUS.

Moreover, Figure 4 presents the class distribution of samples in the clean version of MUFAC, showing that the forget set and the unseen set (*i.e.*, the validation set in our case) follow different class distributions. The strong performance of LoTUS in MUFAC further suggest that the assumption of distributional similarity between the forget and unseen sets, discussed in Sec. 3.2, can be relaxed.

13. Failure Analysis

Unlearning samples from MUFAC (the clean version) presents greater challenges for all unlearning methods, as reflected in significantly higher JSD scores in Tab. 1. In addition, MUFAC & ResNet18 is the only benchmark where LoTUS achieves the second-best Avg Gap rather than the best. To explore the particularities of this dataset, we investigated the orthogonality of the forget and retain sets (*i.e.*, how much they differ). Figure 5 presents that the images in the forget and retain sets of MUFAC are more similar, making unlearning more challenging.

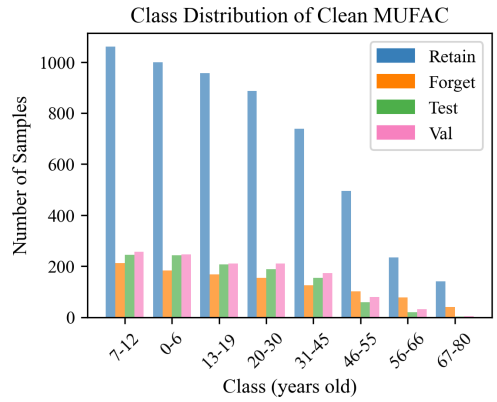


Figure 4. **Number of MUFAC Samples per Class & Split.** Unlike the balanced CIFAR-10/100 splits, MUFAC exhibits imbalanced class distributions of that varies across the retain, forget, test, and validation splits.

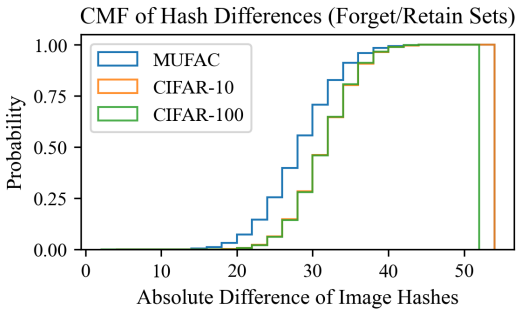
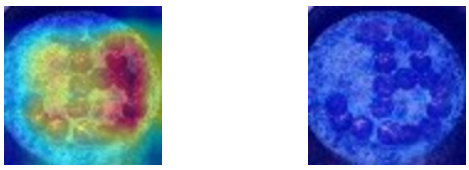


Figure 5. **Orthogonality of Forget/Retain Sets.** We measure the similarity between samples in the forget and retain sets using the absolute difference between their image hashes. MUFAC exhibits significantly higher similarity between forget and retain sets, complicating the unlearning process.



(a) prediction: *pizza* (b) prediction: *potpie*

Figure 6. **Class Activation Maps and Model Predictions:** (a) before and (b) after class unlearning.

14. Class Unlearning with LoTUS

After retraining the model excluding a single *pizza* image from the training set, the model preserves *global information* that stems from the remaining *pizzas* in the training set, being able to correctly classify many of them (see Forget Acc. in Tab. 6). In instance-wise unleanring, LoTUS prevents performance degradation by preventing the elimination of *global information*. To do so, it uses

	Metric (↓)	Gold Std	Finetuning	NegGrad+	RndLbl	BadT	SCRUB	SSD	UNSIR	SalUn	**LoTUS**
TinyIN Pizza	Avg Gap	0.0000	0.2975	0.3250	0.2925	0.3125	0.4200	0.3650	0.5075	0.2925	**0.0925**
	JSD ×1e4	$0.00_{\pm 0.00}$	$94.96_{\pm 7.24}$	$86.36_{\pm 9.66}$	$92.27_{\pm 6.43}$	$72.62_{\pm 22.07}$	$73.10_{\pm 0.82}$	$34.96_{\pm 14.21}$	$102.29_{\pm 9.33}$	$91.01_{\pm 8.59}$	$\mathbf{37.02}_{\pm \mathbf{18.68}}$
	Time (min.)	$42.15_{\pm 16.05}$	$3.23_{\pm 0.01}$	$3.24_{\pm 0.03}$	$3.27_{\pm 0.03}$	$1.59_{\pm 0.01}$	$4.05_{\pm 0.03}$	$3.19_{\pm 0.03}$	$\mathbf{1.01}_{\pm \mathbf{0.01}}$	$3.98_{\pm 0.01}$	$1.30_{\pm 0.02}$
C-100 Beaver	Avg Gap	0.0000	0.2825	0.3725	0.2925	0.3000	0.3225	0.4325	0.4050	0.2850	**0.1200**
	JSD ×1e4	$0.00_{\pm 0.00}$	$101.48_{\pm 2.87}$	$108.50_{\pm 2.59}$	$102.66_{\pm 3.11}$	$78.65_{\pm 3.12}$	$64.09_{\pm 8.71}$	$45.19_{\pm 9.19}$	$76.28_{\pm 6.88}$	$100.93_{\pm 2.44}$	$\mathbf{25.46}_{\pm \mathbf{1.41}}$
	Time (min.)	$4.00_{\pm 0.11}$	$0.43_{\pm 0.00}$	$0.44_{\pm 0.01}$	$0.45_{\pm 0.00}$	$0.26_{\pm 0.01}$	$0.55_{\pm 0.00}$	$0.83_{\pm 0.03}$	$\mathbf{0.20}_{\pm \mathbf{0.01}}$	$1.16_{\pm 0.01}$	$0.23_{\pm 0.01}$

Table 9. **Class Unlearning** with ResNet18 models and the TinyImageNet (TinyIN) and CIFAR-100 (C-100) datasets. We highlight the **best** and second-best scores.

accuracy on labeled unseen *pizzas*, $\mathrm{Acc}(f_{\mathrm{orig}}, D_u)$ in Eq. (11) as an estimator of *global information*.

Framing class unlearning as sequential instance-wise unlearning applied to all class samples, *global information* is ultimately eliminated (see Class Activation Maps of *pizza* class in Fig. 6). Since there is no global information to estimate, we also do not need the unseen set. To adapt LoTUS to class unlearning, we set as objective the accuracy on the forget set to become zero (an empirical observation by retaining the model without the specific class):

$$\tau_d = \exp\left(\alpha(\mathrm{Acc}(f_{\mathrm{un}}, D_f) - \cancelto{0}{\mathrm{Acc}(f_{\mathrm{orig}}, D_u)})\right) \tag{14}$$

Table 9 shows that LoTUS can be adapted to the class unlearning task, outperforming state-of-the-art methods, combining unlearning effectiveness and efficiency.

15. Contribution of Gumbel noise

In Tab. 10, we demonstrate the contribution of the introduction of Gumbel noise in the Softmax activation function. To do so, we perform an ablation analysis using the Gumbel Softmax and the Softmax with Temperature as activation functions in LoTUS. Softmax with Temperature is defined similarly with Eq. (15) as:

$$p_i = s_t(\pi, \tau) = \frac{\exp\left((\log \pi_i)/\tau\right)}{\sum_{j=1}^{k} \exp\left((\log \pi_j)/\tau\right)}, \; i = 1, \ldots, k \tag{15}$$

16. Entropy-based Analysis of the Streisand Effect

Further evaluation of the Streisand effect includes investigating the model's uncertainty, as in [15]. In Fig. 7, it is shown that LoTUS prevents an adversary from readily inferring whether an instance is a member of the training set, or whether it belongs to the forget or retain set, since the entropy distributions of the forget/retain/test sets are similar. In contrast, the existing unlearning method [17] that also performs in the output space, but indiscriminately increases the entropy, clearly presents a significant vulnerability to the Streisand effect.

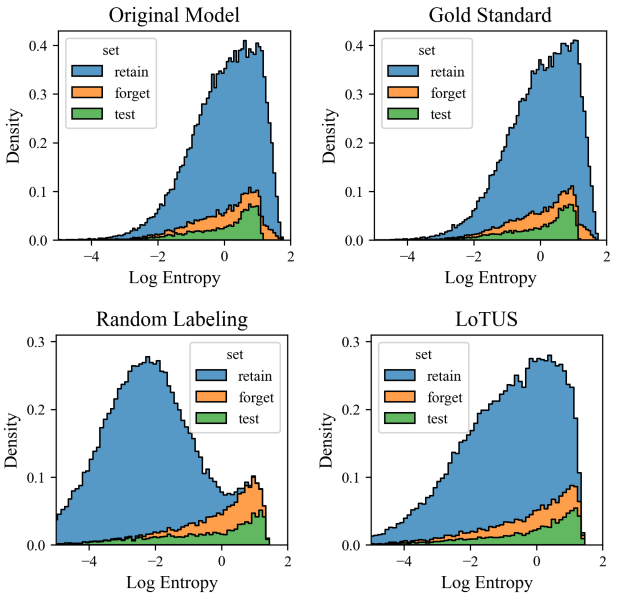


Figure 7. **Privacy Evaluation via entropy comparison:** LoTUS achieves indistinguishable entropy distributions between forget and retain sets, similar to the orignal and gold standard models. In contrast, Random Labeling produces disproportionately lower entropy in the retain set, making it easier for adversaries to distinguish retain from forget and unseen samples.

17. Social Impact

LoTUS can address privacy-related concerns, such as opt-out requests, where users request their data to be deleted not only from the databases, but also from the DNN models. From a security perspective, LoTUS can be applied to unlearn training samples modified by adversaries, which may otherwise compromise the model's performance. In such scenarios, where privacy or security issues arise for specific data points and need to be removed, instance-wise unlearning is more consistent with real-world conditions than class unlearning [5].

		Vision Transformer				ResNet18			
		TinyImageNet	CIFAR-100	CIFAR-10	MUFAC	TinyImageNet	CIFAR-100	CIFAR-10	MUFAC
Avg Gap	Gumbel-Softmax	**0.0150**	**0.125**	**0.0050**	**0.0200**	**0.1675**	0.1200	**0.0350**	0.1250
	Softmax with Temperature	0.0675	0.0225	**0.0050**	**0.0200**	0.1850	**0.1075**	0.0675	**0.1175**
JSD$\times 1e4$	Gumbel-Softmax	**0.03**	**0.04**	**0.01**	**0.05**	**0.62**	1.67	**0.32**	**6.90**
	Softmax with Temperature	0.15	**0.04**	**0.01**	0.08	0.65	**1.36**	0.41	7.33

Table 10. **Contribution of Gumbel noise into the activation function.** Ablation analysis using Gumbel-Softmax and Softmax with Temperature as activation functions. LoTUS performs better with Gumbel-Softmax in the majority of the benchmarks.