GENERALIZATION GUIDED MACHINE UNLEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep machine unlearning has become increasingly critical and urgent across various applications. Existing unlearning methods typically modify the model's weights to scrub the information associated with the specific forget set. However, they often fail to fulfill the *effective and efficient unlearning goal* while *exposing serious privacy leakage risks* for data to be forgotten.

Rethinking the machine unlearning problem through the lens of model generalization, we provide a fundamental understanding of the essentials of machine unlearning and validate the privacy risks of deep models trained under various generalization-improving policies. Building upon the intrinsic insights and key observations gained, we propose generalization-guided unlearning methods, GENUN, consisting of GENUN-M by leveraging a high-bias checkpoint model and GENUN-F by employing targeted rewinding and unlearning via soft targets derived from contrastive learning. Extensive empirical evaluations demonstrate that GENUN algorithms consistently achieve near-ideal forget quality and utility while maintaining significant speed-ups. They consistently outperform various state-of-the-art baselines and can protect the privacy of the forget data effectively[1].

1 INTRODUCTION

Deep models have become indispensable across various applications, affecting all aspects of daily life. Legislation regulations, such as GDPR (Regulation, 2018) and CCPA (CCP, 2024), have reinforced the *right to be forgotten*, granting individuals the right to erase their data to protect privacy and prevent unauthorized third-party access; there is also a widespread need to update models by removing learned knowledge corresponding to outdated information or outliers. Thus, it raises a fundamental question: *Can a deep model accurately and efficiently 'forget' any part of its training data?*

Machine unlearning (MU) aims to erase the impact of a subset of the training data on the already trained model to fulfill such objectives, and has been attracting significant attention Triantafillou et al. (2023). However, the highly non-convex nature and complexity of deep models make it challenging to isolate and scrub the effect of a specific training subset on the model weights, Therefore, achieving efficient and reliable MU remains a significant challenge.

To mitigate the high computational overhead of exact unlearning—retraining from scratch on the retain data, which is regarded as the gold standard, various approximate MU methods have been proposed (Nguyen et al., 2022; Xu et al., 2023). These approaches attempt to modify the original model's weights to achieve a performance comparable to that of the retrained model, ensuring the model's error on the forget data is as high as on data it has never seen before—*a property closely related to model generalization*. However, existing methods suffer from serious defects in terms of forget efficacy, adaptability to different forget-set sizes, and privacy preservation, while often imposing significant restrictions on model architectures and training procedures. Although some approaches improve efficiency compared to retraining, *no approach has achieved an effective and comprehensive trade-off across all these metrics*, *nor can it ensure complete privacy protection for forget information within the model*, as we validated, which limits their practical application.

We rethink MU problem via the lens of model generalization. Grounded in the strong correlation between privacy risk and generalization gap (Yeom et al., 2018), we establish a fundamental understanding about the essentials of MU by examining its intrinsic connection to model generalization

[1]Our code is anonymously available at: https://anonymous.4open.science/anonymize/GENUN-3623.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

capability, which is supported by exhaustive empirical validation for deep models trained under various generalization-improving policies. Building upon the intrinsic insights and key observations, we propose generalization-guided unlearning for two settings: **GENUN-M**, *utilizing high-bias intermediate checkpoints to prevent the retention of specific knowledge from the forget set while significantly reducing the computational cost of erasure;* and **GENUN-F**, *leveraging the soft and implicit target derived from unsupervised contrastive learning to guide state rewinding and knowledge removal*.

We conduct extensive evaluations for GENUN across diverse baselines, datasets, and model architectures. Our results verify that GENUN consistently outperforms all state-of-the-art (SOTA) methods, achieving superior performance across full-stack metrics. The holistic evaluation with MIA validates GENUN's effectiveness in protecting forget data, highlighting its advantage over existing methods.

2 PROBLEM DEFINITION

Notation and preliminaries. Given a training dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n \sim \mathcal{P}^{\bigotimes n}$ of n samples where $\mathbf{x}_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$ are i.i.d. drawn from a joint distribution $\mathcal{P}$ over $\mathcal{X} \times \mathcal{Y}$; $\mathcal{X} \subseteq \mathbb{R}^F$ with dimension as F and label space $\mathcal{Y} = \{1, \ldots, K\}$ with K classes. X and Y are the corresponding random variables for $\mathbf{x}_i$ and y_i with $(X, Y) \sim \mathcal{P}$. Let a deep neural network (DNN) $\mathcal{F} : \mathcal{X} \mapsto \mathbb{R}^{|\mathcal{Y}|}$ be a function with the trainable parameters $\mathbf{\Theta}$ and trained in a supervised manner that minimizes the empirical risk with optimization algorithms, e.g., SGD. Here, the loss function is cross-entropy loss as $\mathcal{L}_{\mathcal{D}}(\mathbf{\Theta})$. Given an input $\mathbf{x} \in \mathcal{X}$, $\mathcal{F}(\mathbf{x}; \mathbf{\Theta})$ denotes the logits output by the DNN model on $\mathbf{x}$.

Machine Unlearning. Let $\mathcal{D}_f \subset \mathcal{D}$ be a subset of the training dataset composed of n_f samples to be removed as the *forget set*; $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$ ($\mathcal{D}_r \cap \mathcal{D}_f = \emptyset$), is the *remaining set* of size n_r, where the model is allowed to retain information for. Besides, $\mathcal{D}_v$ and $\mathcal{D}_t$ are the validation and test data sets drawn from $\mathcal{P}$. Under the MU setting, it is usually assumed that $n_f \ll n_r$, i.e., $\mathcal{D}_f$ is just a small portion of $\mathcal{D}$. We mainly focus on *random-subset forgetting* for the classification task: $\mathcal{D}_f$ may either consist of a randomly selected subset of the samples from a single class or multiple classes of $\mathcal{D}$.

Assume that $\mathcal{F}(\cdot; \bar{\mathbf{\Theta}})$ is a well-trained DNN model with parameters $\bar{\mathbf{\Theta}}$ over $\mathcal{D}$ by minimizing the loss $\mathcal{L}_{\mathcal{D}}$: $\bar{\mathbf{\Theta}} = \arg\min_{\mathbf{\Theta}} \sum_{(\mathbf{x},y)\in\mathcal{D}} \ell(\mathcal{F}(\mathbf{x}; \mathbf{\Theta}), y)$ — it is the *original model*, i.e., before performing any unlearning intervention. The *retrained model*, $\mathcal{F}(\cdot; \mathbf{\Theta}^*)$, is trained from scratch (with the same learning process) by exclusively utilizing $\mathcal{D}_r$ while ignoring all forget samples by minimizing $\mathcal{L}_{\mathcal{D}_r}$, which is deemed the optimal unlearned model and regarded as the gold standard for unlearning.

Given $\mathcal{F}(\cdot; \bar{\mathbf{\Theta}})$, $\mathcal{D}_f$, and $\mathcal{D}_r$, MU aims to erase the influence of $\mathcal{D}_f$ on $\bar{\mathbf{\Theta}}$ to produce updated weights $\mathbf{\Theta}^u$ with an unlearning algorithm $\mathcal{U}$, such that the *unlearned model* $\mathcal{F}(\cdot; \mathbf{\Theta}^u)$ has 'forgotten' $\mathcal{D}_f$ and maintains similar generalization ability to unseen data as the original one. Ideally, $\mathcal{F}(\cdot; \mathbf{\Theta}^u)$ is expected to be as similar to $\mathcal{F}(\cdot; \mathbf{\Theta}^*)$ as possible, while retraining is prohibitively costly, especially for DNNs. So, an efficient MU method should also be more economical regarding computational cost than retraining. Therefore, MU methods should satisfy the following requirements:

- **R1** (**Forget**): the influence of $\mathcal{D}_f$ doesn't exist in $\mathcal{F}(\cdot; \mathbf{\Theta}^u)$, i.e., similar to $\mathcal{F}(\cdot; \mathbf{\Theta}^*)$ over $\mathcal{D}_f$;
- **R2** (**Accuracy**): the prediction capability of $\mathcal{F}(\cdot; \mathbf{\Theta}^u)$ for $\mathcal{D}_t$ is comparable to that of $\mathcal{F}(\cdot; \bar{\mathbf{\Theta}})$;
- **R3** (**Efficiency**): the computational costs (e.g., running time) are cheaper than retraining.

3 METHODOLOGY

We establish an essential understanding of the core aspects of machine unlearning via the lens of model generalization. We empirically examine the effect of various strategies designed for improving generalization on membership inference attack (MIA) risks. Based on these insights, we propose generalization-guided unlearning algorithms, tailored for different practical scenarios for DNNs.

3.1 UNLEARNING FROM THE ML PRINCIPLE PERSPECTIVE.

Given a trained model $\mathcal{F}(\cdot; \bar{\mathbf{\Theta}})$, its parameters $\bar{\mathbf{\Theta}}$ encapsulate the task-related knowledge acquired from $\mathcal{D}$. However, due to memorization (Feldman, 2020; Feldman & Zhang, 2020), the model retains information specific to both $\mathcal{D}_r$ and $\mathcal{D}_f$ more or less in addition to learning inherently generalizable

shared knowledge. It raises elementary questions: *How does a model's generalization capability impact $\mathcal{U}$? Can generalization insights be leveraged to achieve effective MU?*

Theorem 1 (**Informal**). *Let $\mathcal{A}$ be a learning algorithm producing parameters $\Theta = \mathcal{A}(\mathcal{D})$ from $\mathcal{D} \sim \mathcal{P}^{\otimes n}$ with bounded loss. Assume that at least one of the following conditions holds, uniform stability, smoothness & local density, and representation prior, if the learned model $\mathcal{F}(\cdot; \widehat{\mathbf{\Theta}})$ has ϵ-bounded expected generalization gap, i.e., being a well-generalized model*[2]*, then it exhibits only weak dependence on any individual training sample, and $\widehat{\mathbf{\Theta}}$ are dominated by* distribution-level (shared) information *(i.e., $\mathcal{P}$-related) rather than sample-specific memorization.*

Consider that $\mathcal{D}_f$ is *a uniformly selected small-size subset* from $\mathcal{D}$; it almost surely consists of typical data points[3]. Under standard algorithm stability assumptions for $\mathcal{A}$, thus the contribution (influence) of $\mathcal{D}_f$ on the learned parameters is negligible, and removing $\mathcal{D}_f$ typically has little or no effect on the learned shared information. In this regime, for the original model $\mathcal{F}(\cdot; \bar{\mathbf{\Theta}})$, *the practical objective of machine unlearning is to remove only residual, sample-specific influence* attributable to $\mathcal{D}_f$, provided that $\mathcal{D}_f$ does not exert a dominant influence on the learned model.

MIA measures privacy risk by determining whether a given sample belongs to the training set by designing adversarial attacks, i.e., assessing whether the influence of $\mathcal{D}_f$ has been completely removed in $\mathbf{\Theta}^u$ for MU setting. *Perfect unlearning* ensures that *forget samples are indistinguishable from non-membership samples (i.e. test set)* for the adversary, effectively reducing MIA success to random guessing, where the Attack Success Rate (ASR) as a metric drops to 50%.

Overfitting and memorization in DNNs are primary contributors to the success of MIAs, as they induce systematic differences between the model's behavior on training and non-training samples. For models with strong generalization—i.e., where the expected loss on training and test data is nearly identical—the advantage of a broad class of loss-based MIAs is provably bounded by the generalization error and vanishes as the gap goes to zero, consistent with the analysis of (Yeom et al., 2018). Consequently, improving generalization helps to reduce privacy risk (MIA can be a proxy to measure),although perfect generalization alone does not preclude all possible inference attacks.

Ultimately, what determines a model's practical utility is its ability to generalize to unseen data, with performance on a test set serving as a standard proxy. Accordingly, the unlearning requirement **R**2 for prediction capability should emphasize $\mathcal{D}_t$, i.e., ensuring $Acc(\mathcal{D}_t; \mathbf{\Theta}^u) \approx Acc(\mathcal{D}_t; \bar{\mathbf{\Theta}})$, rather than for $\mathcal{D}_r$, which matches with (Hayes et al., 2025), especially when unlearning comparison.

Proposition 1 (Model Generalization and Unlearning)**.** *Under standard stability assumption and in the limit of a strong generalization, the dependence of the learned parameters on any small subset vanishes. In this regime, the model predominantly encodes shared, distribution-level information, and the sample-specific contribution of $\mathcal{D}_f$ is negligible. Consequently, for MU objective, such a well-generalized model approximately satisfies unlearning requirements **R**1-**R**2, without requiring substantial parameter modification.*

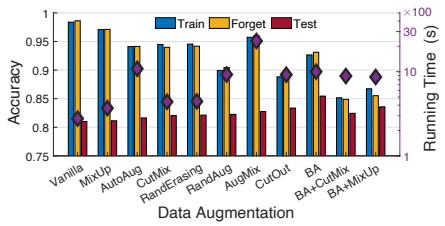
3.2 Empirical Validation & Insights for Principles.

Taking ResNet-18 trained on CIFAR-10 as a representative example, we empirically examine how commonly used strategies for improving the generalization capability of DNN models—including various data augmentation techniques and training policies—affect privacy risk. Our results provide evidence supporting the relationship between generalization and membership inference risk established in Yeom et al. (2018), and offer practical insights for the design of MU algorithms. In terms of generalization measure, we use the accuracy gap as a metric: $G_{Acc}(\mathbf{\Theta}) = Acc(\mathcal{D}; \mathbf{\Theta}) - Acc(\mathcal{D}_t; \mathbf{\Theta}) \geq 0$. A smaller G_{Acc} indicates superior generalization, $G_{Acc}(\mathbf{\Theta}) = 0$ for a well-generalized model with $\mathbf{\Theta}$. A good model means that it has high test accuracy and a small G_{Acc}.

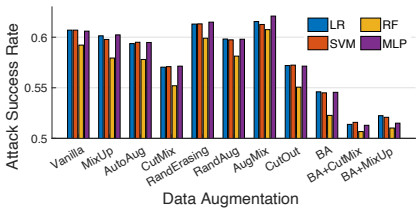
Applying various data augmentation techniques (Shorten & Khoshgoftaar, 2019) to $\mathcal{D}$, such as *MixUp, AutoAug, CutMix, RandErasing, AugMix, CutOut, Random Crop & Flip (BA)*, and their composition,

[2]Its accuracy remains consistent across different subsets drawn from $\mathcal{P}$, i.e., $Acc(\mathcal{D}; \widehat{\mathbf{\Theta}}) \approx Acc(\mathcal{D}_t; \widehat{\mathbf{\Theta}})$ for any $\mathcal{D}_t \sim \mathcal{P}$, where $Acc(\mathcal{D}; \widehat{\mathbf{\Theta}})$ denotes the accuracy of the model $\mathcal{F}(\cdot; \widehat{\mathbf{\Theta}})$ over $\mathcal{D}$.

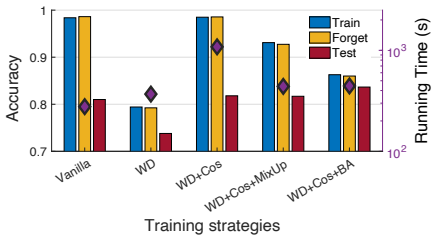
[3]It follows that atypical or rare samples occur with probability $\mathcal{O}(1/n)$, which might contribute to atypical subpattern learning for the model to achieve good generalization Feldman (2020), which implies that the fraction of atypical samples in randomly chosen $\mathcal{D}_f$ concentrates at zero with probability $1 - \mathcal{O}(n_f/n)$.



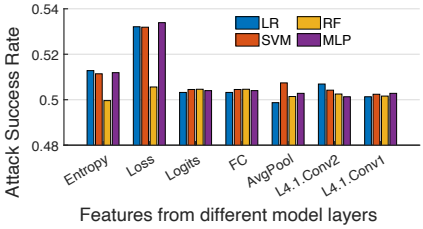
(a) Accuracy and Running Time.



(b) ASRs of MIA.



(c) Accuracy and Running time.



(d) ASR of MIA.

Figure 1: Performance of ResNet-18 trained with various data augmentation techniques and adopting various training policies to improve generalization. (a) shows Accuracy and Training runtime (Diamond); (b) is ASR in MIA for attackers (*'LR', 'SVM', 'RF', 'MLP'*) over a balanced audit set constructed with model's 'Loss' value. 'Aug' and '*BA*' denote *'Augmentation'* and Basic Augmentation ('*Random Crop & Flip*'); (c) shows Accuracy and Training runtime (Diamond); (d) shows ASR in MIA for the model trained with *'WD+Cos+BA'*, attackers use different layer output as features. Here, *'WD'* is *'weight decay'* and *'Cos'* is *'CosineAnnealingWarmRestarts'* learning rate scheduler.

we train ResNet-18 models and report the performance in Fig. 1(a), and ASR of MIA across four attack models in Fig. 1(b) for a random selected $\mathcal{D}_f$ which corresponds to any part of $\mathcal{D}$.

Compared to the *Vanilla* model (without any augmentation, $G_{Acc} = 0.174$), all these techniques reduce G_{Acc} to varying degrees ($G_{Acc} \in [0.027, 0.159]$) albeit at the cost of longer training time. For the composition augmentations, which achieve the best performance and outperform others, *'BA+CutMix'* is the best with the smallest G_{Acc} and lowest ASR, followed closely by *'BA+MixUp'*, which has slightly higher accuracy but similar G_{Acc} and ASR. All single augmentations have a larger generalization gap and higher ASR across different MIA attackers. Notably, *'CutOut'* achieves smaller G_{Acc}, followed by *'BA'* and *'RandAug'* with comparable results; *'BA'* obtains a better ASR, followed by *'CutOut'* and *'CutMix'* with similar values, *'RandErasing'* and *'AugMix'* are even worse than *Vanilla* model for ASR.

Observation 1. *Common data augmentation can improve generalization to a certain extent at the cost of a certain amount of training time, and there is some specific compositional augmentation that can help train a better generalizable model, although it may not be perfect.*

Adopting different policies in the model training process, including weight decay, a learning rate scheduler, and data augmentation, we train ResNet-18 models and report the results in Fig. 1(c). All strategies reduce G_{Acc} compared to the Vanilla model, except for *'WD'* alone. Among them, *'WD+Cos+BA'* achieves the highest test accuracy and the smallest G_{Acc} (≈ 0.025). However, applying weight decay alone significantly degrades model performance. Although adding *'Cos'* can slightly improve the test accuracy, it increases overfitting compared to *'WD'* alone and requires longer training time. Choosing *'WD+Cos+BA'* as the best-performing model as the target, we show the ASR of MIA across four attackers in Fig. 1(d), who launch attack based on balanced audit sets derived from different features of varying model layers. Their ASRs is consistently close to perfect results (about 0.5) across different layers' output; *'Loss'*-based ASR result is slightly higher, but still no more than 0.53. The results for the impact of different optimizers, e.g., SGD and AdamW, are similar to the above findings of *Adam*.

Observation 2. *Optimization-level regularization (e.g., 'WD' and 'Cos') combined with stochastic data augmentation mechanisms (e.g., 'Mixup' or 'BA' etc.) reshapes the training dynamics toward*

reduced memorization. Such training regimes also reduce the associated generalization gap, leading to a near-indistinguishable loss and confidence distributions between forget and non-member samples, and consequently driving MIA attack success rates toward a random chance.

Lemma 1. *These empirical observations substantiate the essential analysis presented in Proposition 1. Consequently, similar techniques, like compositional data augmentation and elaborate training policy, can be leveraged to obtain a highly generalizable model $\mathcal{F}(\cdot;\bar{\mathbf{\Theta}})$ which can accommodate forget requests for small subsets $\mathcal{D}_f$ in a cost-efficient manner. In this sense, training models with strong generalization capability provides a principled and practical foundation for machine unlearning.*

Lemma 2. *For a generally trained $\mathcal{F}(\cdot;\bar{\mathbf{\Theta}})$, reducing the generalization gap using any technique, similar to the aforementioned ones, does help to reduce the privacy risk associated with MIA by enhancing its generalization capability.*

3.3 Generalization-guided Unlearning Algorithms: GenUn.

3.3.1 GenUn-M Algorithm

Building upon the preceding analysis and under a mild assumption, we posit that *a model without severely overfitting provides a favorable initialization for machine unlearning*. In this regime, the influence of $\mathcal{D}_f$-specific information on the learned parameters is minor or negligible, thereby reducing the effort for effective unlearning. The overparameterized deep networks tend to first learn simple and generalizable patterns, and only later fit noisy labels or sample-specific details, which has been studied both empirically and theoretically in (Arpit et al., 2017). Such transition is frequently accompanied by a change in the generalization gap. This observation motivates identifying a critical training point corresponding to the epoch at which the generalization gap G_{Acc} is minimized.

So when the training trajectory is traceable or intermediate model checkpoints are available, such a critical point model can be directly selected at *essentially no additional cost*. For training regimes that fine-tune pre-trained models, a similar dynamic is typically observed, although the minimum achievable G_{Acc} may be slightly larger due to inherited representations and optimization constraints.

We denote the resulting model at epoch k as $\mathcal{F}(\cdot;\widehat{\mathbf{\Theta}}_k)$, which represents a generalized checkpoint (Chen et al., 2024), that is, potentially exhibiting lower peak accuracy but limited overfitting, and satisfies $0 \leq G_{Acc}(\widehat{\mathbf{\Theta}}_k) \ll G_{Acc}(\bar{\mathbf{\Theta}})$ while $Acc(\mathcal{D}_t;\widehat{\mathbf{\Theta}}_k) \leq Acc(\mathcal{D}_t;\bar{\mathbf{\Theta}})$.

Hence, we propose **GenUn-M** for the MU task. We assume that $\widehat{\mathbf{\Theta}}_k$ is available for the original model w.r.t. $\bar{\mathbf{\Theta}}$. Given $\mathcal{D}_f$ and $\mathcal{D}_r$, GenUn-M finetunes $\widehat{\mathbf{\Theta}}_k$ to satisfy the requirements **R**1-**R**2 by minimizing the combined objective:

$$\min_{\mathbf{\Theta};\, \mathbf{\Theta} \xleftarrow{\text{init}} \widehat{\mathbf{\Theta}}_k} \quad \alpha\mathcal{L}_r - (1-\alpha)\mathcal{L}_f \tag{1}$$

where $\mathcal{L}_r = \frac{1}{n_r}\sum_{(\mathbf{x},y)\in\mathcal{D}_r} \ell(\mathcal{F}(\mathbf{x};\mathbf{\Theta}),y)$ improves the prediction capability for **R**2 by fitting $\mathcal{D}_r$, $\mathcal{L}_f = \frac{1}{n_f}\sum_{(\mathbf{x},y)\in\mathcal{D}_f} \ell(\mathcal{F}(\mathbf{x};\mathbf{\Theta}),y)$ is to *eliminate the possible residual influence* of $\mathcal{D}_f$, especially when $G_{Acc}(\widehat{\mathbf{\Theta}}_k) > 0$, for **R**1. The coefficient $\alpha \in [0,1]$ controls the trade-off of the two components and $\alpha = 0$ if $G_{Acc}(\widehat{\mathbf{\Theta}}_k) \approx 1$. Also, weight regularizers can be introduced in Eq. 1, e.g., ℓ_1 or ℓ_2 norm, to obtain potential gains as (Jia et al., 2023) did.

The key distinction of GenUn-M lies in its *initialization strategy*: instead of starting from the fully trained model $\bar{\mathbf{\Theta}}$, it initializes the unlearning process from the earlier, well-generalized model state $\widehat{\mathbf{\Theta}}_k$. This choice substantially reduces the optimization effort required to suppress $\mathcal{D}_f$-specific influence, as $\widehat{\mathbf{\Theta}}_k$ contained limited sample information by construction, i.e., less contaminated.

The loss formulation in GenUn-M, shown in Eq. (1), is structurally similar to those adopted by existing unlearning approaches, such as fine-tuning, gradient ascent-based forgetting, and their variants. However, GenUn-M differs fundamentally in that it operates in a parameter regime with reduced memorization, making the removal of $\mathcal{D}_f$-specific information more efficient and stable.

3.3.2 GenUn-F Algorithm

When only the fully trained model w.r.t. $\bar{\mathbf{\Theta}}$ is available, unlearning becomes substantially more challenging. In such a model, information associated with $\mathcal{D}_f$ is often deeply and implicitly encoded

in the parameter space, making it difficult to remove in a cost-effective manner (Zhao et al., 2024), thus, inducing a direct model-state phase transition (e.g., $\bar{\mathbf{\Theta}} \rightarrow \widehat{\mathbf{\Theta}}$) becomes impossible.

Moreover, unlike $\mathcal{D}_r$ for which the desired predictive behavior of the unlearned model $\mathbf{\Theta}^u$ is well defined, the expected output of $\mathbf{\Theta}^u$ on samples in $\mathcal{D}_f$ is inherently ambiguous. Hence, the unlearning process cannot be explicitly or precisely controlled by prescribing target labels or outputs for $\mathcal{D}_f$.

To this end, we propose **GENUN-F**, a target rewinding algorithm for unlearning. GENUN-F *constructs a soft and implicit target for $s \in \mathcal{D}_f$* using an *unsupervised contrastive learning* mechanism. Rather than enforcing explicit predictions, this implicit target guides by specifying what representations or behaviors $\mathbf{\Theta}^u$ should resemble, as well as what it should be discouraged from encoding.

The regularities induced through this unsupervised contrastive objective promote improved generalization and effectively reduce the generalization gap, thereby mitigating memorization-driven privacy risks. Combined with supervision on $\mathcal{D}_r$ to preserve predictive utility, GENUN-F performs unlearning via fine-tuning to minimize the objective:

$$\min_{\mathbf{\Theta};\, \mathbf{\Theta} \xleftarrow{\text{init}} \bar{\mathbf{\Theta}}} \alpha \mathcal{L}_r + (1-\alpha)\mathcal{L}_f^{cont}, \tag{2}$$

where $\mathcal{L}_r$ is same as in Eq. (1), $\mathcal{L}_f^{cont}$ is the InfoNCE (Oord et al., 2018) contrastive loss for $\mathcal{D}_f$ defined in Eq. 3, and $\alpha \in [0, 1]$ is a coefficient to balance the two components.

For each $s = (\mathbf{x}_f, y_f) \in \mathcal{D}_f$, although the ideal post-unlearning prediction $\mathcal{F}(\mathbf{x}_f; \mathbf{\Theta}^u)$ is in general unknowable, we can still impose implicit behavioral constraints on the unlearned model through its latent representations. In particular, within the representation space of a generalizable (or robust) model $\widehat{\mathbf{\Theta}}$, we expect the representation of $\mathbf{x}_f$ to remain close to that of its noise-perturbed or augmented variant $\tilde{\mathbf{x}}_f$, while being dissimilar to or far away from those of samples from $\mathcal{D}_r$, especially those in its local neighborhood $\mathcal{N}_{\mathcal{D}_r}(\mathbf{x}_f)$.

Let $\mathcal{F}_{enc}(\mathbf{x}; \mathbf{\Theta})$ be the representation of $\mathbf{x}$ extracted from an intermediate layer of the model of $\mathbf{\Theta}$. Under the contrastive learning framework (Chopra et al., 2005; Wang & Isola, 2020a), for a forget sample $\mathbf{x}_f \in \mathcal{D}_f$, we construct an encoded query $\mathbf{q} = \mathcal{F}_{enc}^q(\mathbf{x}_f; \mathbf{\Theta}^q)$ where $\mathcal{F}_{enc}^q$ is the quey encoder with parameter $\mathbf{\Theta}^q$. We further define a set of encoded keys $\{\mathbf{k}_1, \cdots, \mathbf{k}_K\}$, $\mathbf{k}_i = \mathcal{F}_{enc}^k(\mathbf{x}_i; \mathbf{\Theta}^k)$, where each $\mathbf{x}_i \in \mathcal{N}_{\mathcal{D}_r}(\mathbf{x}_f)$ is sampled from $\mathcal{D}_r$; $\mathcal{F}_{enc}^k$ is the key encoder with parameters $\mathbf{\Theta}^k$. Among these keys, we assume a single positive key $\mathbf{k}^+ = \mathcal{F}_{enc}(\tilde{\mathbf{x}}_f; \mathbf{\Theta})$ corresponding to an augmented (noise per-turbed) view $\tilde{\mathbf{x}}_f$ of the sample $\mathbf{x}_f$, while the remaining $K-1$ keys are treated as negatives. Using dot-product similarity and a temperature τ, we define an InfoNCE-style contrastive forget loss:

$$\mathcal{L}_f^{cont} = -\log \frac{\exp\left(\mathbf{q} \cdot \mathbf{k}^+ / \tau\right)}{\sum_{i=1}^{K} \exp\left(\mathbf{q} \cdot \mathbf{k}_i / \tau\right)}. \tag{3}$$

To efficiently maintain a large and diverse set of negative samples, we adopt the momentum construct (MoCo) (He et al., 2020) framework. Specifically, a large FIFO queue is used as a dynamic dictionary D where the encoded representations from the current mini-batch are enqueued and the oldest are dequeued. This design decouples the effective dictionary size from the mini-batch size. Furthermore, the key encoder $\mathcal{F}_{enc}^k$ is updated as a momentum-based moving average of the query encoder to maintain consistency: $\mathbf{\Theta}^k \leftarrow m\mathbf{\Theta}^k + (1-m)\mathbf{\Theta}^q$, where $m \in [0, 1]$ is the momentum coefficient. In our implementation, the main encoder $\mathcal{F}_{enc}$ servers as the query encoder $\mathcal{F}_{enc}^q$ (immediate $\mathbf{\Theta}^u$).

Note that the augmented sample $\tilde{\mathbf{x}}_f$ will be unseen for the model of $\bar{\mathbf{\Theta}}$; Eq. (3) aligns with Lemma 1-2 since that the model of $\widehat{\mathbf{\Theta}}$ will robust to augmentation transformations, or in turn, the model that leads to $\mathcal{F}_{enc}(\mathbf{x}; \mathbf{\Theta}) \approx \mathcal{F}_{enc}(\tilde{\mathbf{x}}; \mathbf{\Theta})$ will be at least locally generalizable. Therefore, we measure the similarity between data samples in latent feature space considering the broad nature of the definition.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Datasets. We select 4 commonly used datasets for MU: CIFAR-10 (Krizhevsky et al., 2009), CIFAR-100 (Krizhevsky et al., 2009), SVHN (Netzer et al., 2018), and Tiny-ImageNet (Kurmanji et al., 2024), to measure the unlearning performance on image classification tasks.

We choose to forget about $\approx$ 10% of the whole training data $\mathcal{D}$, i.e., $n_f = |\mathcal{D}_v| = 5k$ for CIFAR-10 and CIFAR-100, $7k$ for SVHN, and $10k$ for Tiny-ImageNet. We construct the audit dataset of size n_f, which consists of the same number of samples from the test set $\mathcal{D}_t$ and forget set $\mathcal{D}_f$, for MIA, considering the limited access resource of an attacker.

Models. For CIFAR-10, we fine-tuned a pre-trained ResNet-18 (He et al., 2016) model using Adam with a fixed learning rate of 2e-3, momentum of 0.9, and batch size of 256. For SVHN, we trained an All-CNN (Springenberg et al., 2014) model from scratch. For CIFAR-100, we finetuned the pre-trained VGG-16 (Simonyan & Zisserman, 2014) and ViT (Dosovitskiy et al., 2021) models using Adam with a fixed learning rate of 1e-4 and batch-size of 256.

Baselines. We covered 11 SOTA baselines: **Original**: the model trained on $\mathcal{D}$ without any unlearning. **Retrain**: included as a reference but inviable in practice. **Fine-tuning (FT)**, **Gradient Ascent (GA)** (Golatkar et al., 2020b), **Random Labels (RL)** (Graves et al., 2021), **Fisher Forgetting (Fisher)** (Golatkar et al., 2020b), **Influence Unlearning (IU)** (Izzo et al., 2021), **EU-k** (Goel et al., 2022a), ℓ_1**-sparse**(Jia et al., 2023), **SalUn**(Fan et al., 2023), and **SCRUB**(Kurmanji et al., 2024).

Metrics. We adhere to **6 empirical evaluation metrics**, including utility (unlearning efficacy, classifier's fidelity, generalization ability, computation efficiency) and privacy guarantees as

- *Accuracy:* accuracies for different datasets: Retain ($\mathcal{D}_r$), Forget ($\mathcal{D}_f$), and Test ($\mathcal{D}_t$);
- *Acc-Index:* an integrated accuracy index defined as $F_{acc}(\mathcal{D}_f, \mathcal{D}_t) = Acc(\mathcal{D}_f) - |\text{ Diff }|$ with $|\text{ Diff }| = |Acc(\mathcal{D}_t) - Acc(\mathcal{D}_f)|$.
- *Time:* running time for performing unlearning method;
- *Attack Successful Rate (ASR):* accuracy of MIA of various attackers for privacy guarantees.

Acc-Index $F_{acc} \in [0, 1]$ considers both the *generalization capability* and *unlearning efficacy* of the unlearned model, *a larger* F_{acc} denotes a better performance. We include the *Speedup* w.r.t. *Retrain* to illustrate the efficiency of unlearning approaches. We use 4 types of attacker models, including Logistic Regression, SVM, Random Forest, and MLP with a single hidden layer, for MIA. The ASRs of MIA are the average result of 5 trials. More results and detailed info. are listed in the Appendix.

4.2 Model Generalization Verification.

We train ResNet-18 on CIFAR-10 using various techniques designed to enhance generalization, including different data augmentation strategies and model training policies (as explained in Sec. 3.2). The results are summarized in Fig. 1, with key insights captured in Obv. 1-2.

Fig. 2 presents the training dynamics of two models: All-CNN on SVHN and ResNet-18 on CIFAR-10, both trained without data augmentation (Vanilla). As training progresses, the accuracy on both training and test sets steadily increases until saturation, while G_{Acc} also grows before stabilizing. The pattern confirms that we can always select a 'high-bias' model state at an intermediate training checkpoint, which serves as an optimal initialization $\widehat{\boldsymbol{\Theta}}_k$ for GenUn-M to perform unlearning optimization. Here, $k = 1$ for both models, corresponding to G_{Acc} being 0.98% for All-CNN on SVHN and 4.92% for ResNet-18 on CIFAR10. More results about using other checkpoints with larger G_{Acc} and their effect on α in GenUn-M are shown in the Appendix.

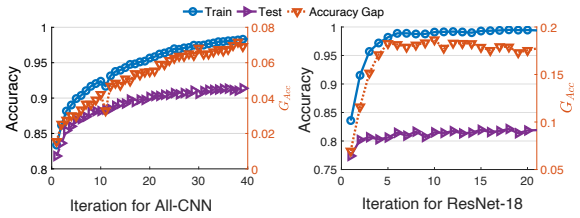


Figure 2: Training dynamics and performance of models. All-CNN is trained from scratch, and ResNet-18 is fine-tuned on a pre-trained model.

4.3 Unlearning: Random-subset Forgetting

We adopt the commonly used setting where $n_f \approx 10\%\ n$. Tab. 1 reports the utility and efficiency comparison across various unlearning methods. Fig. 3 and Fig. 7 show Speedup vs. F_{acc}.

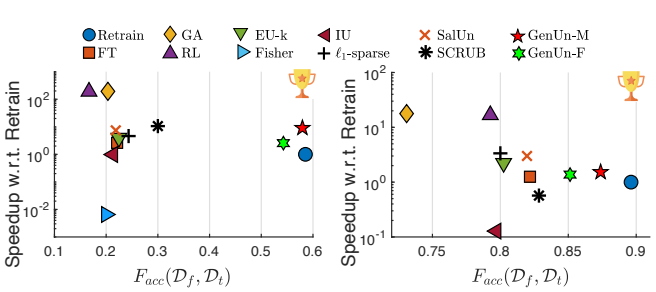


Figure 3: Comparison of MU methods about Speedup w.r.t. Retrain versus F_{acc} for VGG-16 (L) and ViT (R) models.

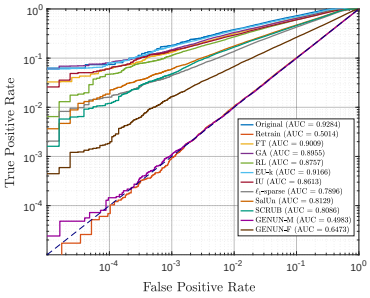


Figure 4: U-LiRA of MU methods for ResNet-18 on CIFAR-10.

Table 1: Performance comparison of various machine unlearning methods for random-subset forgetting w.r.t. **fidelity** ($Acc(\mathcal{D}_r)$), **efficacy** ($Acc(\mathcal{D}_f)$), **generalization ability** ($Acc(\mathcal{D}_t)$ and $|$ Diff $|$) and **computation efficiency** (Time) over CIFAR-10, SVHN, and CIFAR-100 and different models. Fisher for ViT model is excluded due to a runtime timeout.

Dataset (model)	Metric	Original	Retrain	FT	GA	RL	EU-k	Fisher	IU	ℓ_1-sparse	SalUn	SCRUB	$\widehat{\Theta}_k$	GENUN-M	GENUN-F
CIFAR-10 (ResNet-18)	$Acc(\mathcal{D}_r)$	98.29	98.61	98.66	97.50	93.51	98.90	98.27	85.03	96.20	98.51	93.93	83.62	98.40	98.05
	$Acc(\mathcal{D}_t)$ ↑	80.80	80.61	81.18	79.54	76.19	81.55	80.65	71.20	78.70	80.51	78.22	77.38	81.99	78.42
	$Acc(\mathcal{D}_f)$	98.34	81.24	96.95	96.32	93.64	98.72	98.16	85.44	86.52	92.26	89.48	82.30	82.70	80.30
	$\lvert$ Diff $\rvert$ ↓	17.54	0.63	15.77	16.78	17.45	17.17	17.51	14.24	7.82	11.75	11.26	4.92	**0.71**	1.88
	Time (s) ↓	–	230.10	16.61	1.08	1.65	85.28	3679.90	41.07	65.98	86.95	11.93	–	33.41	124.63
SVHN (All-CNN)	$Acc(\mathcal{D}_r)$	98.35	97.99	98.47	98.21	98.00	98.39	97.11	80.19	97.82	98.23	95.50	83.43	97.12	97.01
	$Acc(\mathcal{D}_t)$ ↑	91.31	90.87	91.28	91.01	90.96	91.38	90.09	75.84	91.41	91.29	89.29	81.81	90.76	88.93
	$Acc(\mathcal{D}_f)$	98.19	91.23	97.29	98.07	97.81	98.07	96.99	78.85	96.87	97.99	95.18	82.79	91.34	90.12
	$\lvert$ Diff $\rvert$ ↓	6.88	0.36	6.01	7.06	6.85	6.69	6.90	3.01	5.46	6.70	5.89	0.98	**0.58**	1.19
	Time (s) ↓	–	528.92	20.54	1.16	0.93	97.01	733.81	140.67	11.94	45.01	16.64	–	33.34	78.39
CIFAR-100 (VGG-16)	$Acc(\mathcal{D}_r)$	99.53	99.53	99.59	99.02	98.41	99.88	99.17	99.53	99.85	99.53	99.95	59.92	97.64	96.02
	$Acc(\mathcal{D}_t)$ ↑	60.35	58.63	59.81	58.80	57.61	61.15	59.49	60.35	60.64	60.11	61.48	51.53	59.75	58.06
	$Acc(\mathcal{D}_f)$	99.40	58.72	97.50	97.26	98.54	99.84	98.98	99.38	96.94	98.36	92.96	58.56	61.54	61.84
	$\lvert$ Diff $\rvert$ ↓	39.05	0.09	37.69	38.46	40.93	38.69	39.49	39.03	36.3	38.25	31.48	7.03	**1.79**	3.78
	Time (s) ↓	–	366.09	139.61	1.90	1.93	92.63	55865.00	372.10	79.50	49.45	34.58	–	40.44	143.70
CIFAR-100 (ViT)	Acc($\mathcal{D}_r$)	98.03	97.33	99.45	89.76	97.87	97.84	–	97.80	96.74	99.24	99.79	93.48	96.80	95.37
	Acc($\mathcal{D}_t$) ↑	88.96	90.20	90.40	80.95	88.65	89.03	–	88.84	88.06	90.18	91.02	87.86	89.05	87.94
	Acc($\mathcal{D}_f$)	98.16	90.78	98.62	88.80	98.04	97.82	–	97.98	96.12	98.40	99.20	93.94	90.72	90.75
	$\lvert$ Diff $\rvert$ ↓	9.20	0.58	8.22	7.85	9.39	8.79	–	9.14	8.06	8.22	8.18	6.08	**1.67**	2.81
	Time (s) ↓	–	420.53	334.91	23.65	25.10	189.12	–	3287.35	125.43	140.18	739.71	–	275.63	306.28

As we can see, GENUN consistently achieves the best utility results (GENUN-M is the best one, followed by GENUN-F)—the smallest |Diff| and the largest F_{acc} while maintaining comparable fidelity $Acc(\mathcal{D}_r)$ across all cases; they even surpass Retrain in $Acc(\mathcal{D}_t)$ (except for ViT). In terms of efficiency, GENUN significantly outperforms Retrain, achieving prominent speedups (e.g., 15× for All-CNN). While other methods, such as FT, GA, RL, and SCRUB, have yielded even greater speedups relative to GENUN, their forget performance is substantially worse. Among them, RL and GA achieve the highest speedups but struggle with $Acc(\mathcal{D}_f)$; Fisher takes more time than Retrain; IU performs the worst for F_{acc} for 3 models, while GA is the worst in F_{acc} for the ViT model.

Fig. 5 shows MIA results using the SVM attacker on various features from ResNet-18 over CIFAR-10 (see more in Appendix). GENUN shows superior privacy protection for forget data, with the ASR of all attackers across all features remaining near 0.5, closely matching Retrain. It confirms GENUN's effectiveness in accurate data unlearning and privacy preserving. ℓ_1-sparse achieves near-optimal ASRs by enforcing sparsity in model weights to mitigate overfitting; however, its forget efficacy ($Acc(\mathcal{D}_f)$) and efficiency remain subpar. SCRUB and SalUn perform better than others

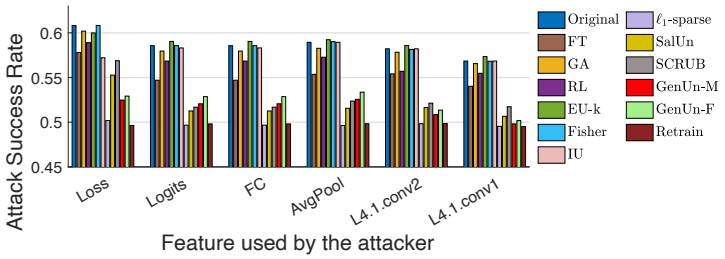
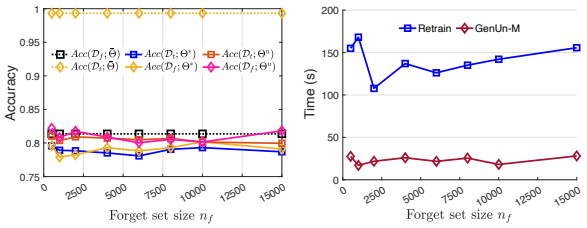


Figure 5: ASR of MIA with SVM attacker for the audit data constructed with different features of ResNet-18's last layers on CIFAR-10.

(consistently worse than GENUN) while their ASRs on '*Loss*' still exceed 0.55 for all attackers, though they perform less conspicuously on other features. Bad-Teaching, despite its good F_{acc}, fails to sufficiently protect against MIA, with most of its ASRs—except for LR on *Loss*—ranking highest among the baselines and exceeding 0.6. Notably, all others suffer from significant forget information leakage issues, as evidenced by ASR results. These conclusions hold consistently across datasets.

Fig. 4 illustrates the results of using a stronger MIA attack, U-LiRA (Carlini et al., 2022; Hayes et al., 2025). GENUN algorithms also achieve the best performance, AUC of GENUN-M is 0.498 close to Retrain, 0.647 for GENUN-F as the second, signigicantly outperforms other baselines.

Impact of forget size n_f. Fig. 6 shows GENUN-M's robust performance in unlearning forget sets of varying sizes. With the validation set size fixed at $5K$, the forget set size varies from 500 to $15,000$. As shown, GENUN-M consistently maintains strong generalization ability and forget efficacy performance, even when unlearning half of $\mathcal{D}$. It surpasses Retrain in terms of $Acc(\mathcal{D}_t)$ and $Acc(\mathcal{D}_f)$. While GENUN-M's runtime keeps stable and maintains significant efficiency compared to Retrain, offering superior and stable performance, even for larger size $\mathcal{D}_f$.



(a) Model accuracy vs. n_f. (b) Running time vs. n_f.

Figure 6: Performance of GENUN-M w.r.t. different n_f.

5 RELATED WORK

Machine Unlearning. To avoid the high computational overhead of exact unlearning (Cao & Yang, 2015), Various approximate unlearning methods have been proposed including Quasi-Newton or influence-based Golatkar et al. (2020a;b), differential privacy-based (Sekhari et al., 2021; Izzo et al., 2021; Ullah et al., 2021), quantization-based (Zhang et al., 2022; Che et al., 2023), relabeling-based (Graves et al., 2021; Fan et al., 2023), and adversarial training-based (Liu et al., 2023a; Zhang et al., 2024c), etc. (Fan et al., 2023; Zhang et al., 2024b; Gandikota et al., 2024; Zhang et al., 2024a; Li et al., 2024) involved unlearning for generative or diffusion models. Membership inference attack (MIA) introduced by (Shokri et al., 2017) is used to evaluate the privacy preservation in MU.

Model Generalization. Being a central focus in learning theory, generalization theory and analysis for modern machine learning algorithms have attracted great attention (Kawaguchi et al., 2017; Zhang et al., 2017; Neyshabur et al., 2019; Saxe et al., 2019; Arora et al., 2018; Zhang et al., 2021a). Various techniques have been designed to improve the generalization capabilities of deep models (Neyshabur et al., 2017; Rohlfs, 2025). E.g., regularization includes weight decay (D'Angelo et al., 2023) and Dropout (Helmbold & Long, 2015). Data augmentation contains MixUp (Zhang et al., 2018; 2021b), CutMix (Yun et al., 2019), AutoAugment (Cubuk et al., 2019), RandErasing (Zhong et al., 2020), AugMix (Hendrycks* et al., 2020), CutOut (DeVries, 2017), and RandAugment (Cubuk et al., 2020). Robust training paradigms have adversarial training, contrastive learning (Mustafa et al., 2022), etc.

A detailed discussion of related work is provided in Appendix A.

6 CONCLUSION

In this work, we revisit the machine unlearning problem through the lens of model generalization, which forms a fundamental understanding about MU and is supported by empirical validation through experiments on the privacy risk of deep models trained with different generalization-improving strategies. Building upon key observations and insights, we propose generalization-guided unlearning algorithms: GENUN-M and GENUN-F for different assumption settings. Comprehensive evaluation validates GENUN consistently outperforms 11 SOTA methods, achieving near-ideal forget efficacy and efficiency while maintaining significant speed-ups. Additionally, the results confirm that GENUN can effectively protect privacy for the forget data, establishing it as a robust and scalable solution.

AI USE STATEMENT

In this work, we used large language models to assist in the writing polish and proof-reading. We have not used generative AI tools for developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, or interpreting results. Generating synthetic datasets does not apply to this work.

We have reviewed all AI-assisted work: all translated and readability-edited text was reviewed by the authors against the original content to ensure that the intended meaning was preserved and that no unsupported claims or substantive changes were introduced. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We have considered the ethical implications of this work and do not identify any specific ethical concerns requiring further discussion.

REPRODUCIBILITY STATEMENT

We provide sufficient details to facilitate the reproducibility of our results. The main paper describes the proposed method and experimental setup, including the model architecture and training procedure. Additional implementation details, hyperparameters, and experimental configurations are provided in the Appendix. Besides, the anonymous link of the implementation code has been provided at the footnote of the main text.

REFERENCES

California consumer privacy act, civil code, division 3, part 4, title 1.81.5, §1798.105 (right to deletion). California Legislative Information, 2024. URL https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?division=3.&part=4.&title=1.81.5&lawCode=CIV. Online; accessed 28-Jan.-2024.

Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. Stronger generalization bounds for deep nets via a compression approach. In *International Conference on Machine Learning*, pp. 254–263. PMLR, 2018.

Devansh Arpit, Stanisław Jastrzkebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

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.

Olivier Bousquet and André Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

George-Octavian Buarbulescu and Peter Triantafillou. To each (textual sequence) its own: improving memorized-data unlearning in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2025.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, pp. 463–480. IEEE, 2015.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pp. 1897–1914. IEEE, 2022.

Tianshi Che, Yang Zhou, Zijie Zhang, Lingjuan Lyu, Ji Liu, Da Yan, Dejing Dou, and Jun Huan. Fast federated machine unlearning with nonlinear functional theory. In *International Conference on Machine Learning*, pp. 4241–4268. PMLR, 2023.

Lin Chen, Michal Lukasik, Wittawat Jitkrittum, Chong You, and Sanjiv Kumar. On bias-variance alignment in deep models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=i2Phucne30.

Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. Boundary unlearning. *arXiv preprint arXiv:2303.11570*, 2023.

S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pp. 539–546 vol. 1, 2005. doi: 10.1109/CVPR.2005.202.

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*, pp. 7210–7217, 2023.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 113–123, 2019.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 702–703, 2020.

Francesco D'Angelo, Maksym Andriushchenko, Aditya Varre, and Nicolas Flammarion. Why do we need weight decay in modern deep learning? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2023.

Terrance DeVries. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

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

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 954–959, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891, 2020.

Rohit Gandikota, Hadas Orgad, Yonatan Belinkov, Joanna Materzynska, and David Bau. Unified concept editing in diffusion models. In *WACV*, pp. 5099–5108. IEEE, 2024.

Xiangshan Gao, Xingjun Ma, Jingyi Wang, Youcheng Sun, Bo Li, Shouling Ji, Peng Cheng, and Jiming Chen. Verifi: Towards verifiable federated unlearning. *arXiv preprint arXiv:2205.12709*, 2022.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. Making ai forget you: Data deletion in machine learning. *Advances in neural information processing systems*, 32, 2019.

Shashwat Goel, Ameya Prabhu, and Ponnurangam Kumaraguru. Evaluating inexact unlearning requires revisiting forgetting. *CoRR abs/2201.06640*, 2022a.

Shashwat Goel, Ameya Prabhu, Amartya Sanyal, Ser-Nam Lim, Philip Torr, and Ponnurangam Kumaraguru. Towards adversarial evaluations for inexact machine unlearning. *arXiv preprint arXiv:2201.06640*, 2022b.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX 16*, pp. 383–398. Springer, 2020a.

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, 2020b.

Anirudh Goyal and Yoshua Bengio. Inductive biases for deep learning of higher-level cognition. *Proceedings of the Royal Society A*, 478(2266):20210068, 2022.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11516–11524, 2021.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. Certified data removal from machine learning models. *arXiv preprint arXiv:1911.03030*, 2019.

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

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *2025 IEEE 3rd Conference on Secure and Trustworthy Machine Learning*. IEEE, 2025.

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.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

David P Helmbold and Philip M Long. On the inductive bias of dropout. *The Journal of Machine Learning Research*, 16(1):3403–3454, 2015.

Dan Hendrycks*, Norman Mu*, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple method to improve robustness and uncertainty under data shift. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1gmrxHFvB.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54(11s): 1–37, 2022.

Li Hu, Anli Yan, Hongyang Yan, Jin Li, Teng Huang, Yingying Zhang, Changyu Dong, and Chunsheng Yang. Defenses to membership inference attacks: A survey. *ACM Computing Surveys*, 56 (4):1–34, 2023.

Sergey Ioffe. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. Approximate data deletion from machine learning models. In *International Conference on Artificial Intelligence and Statistics*, pp. 2008–2016. PMLR, 2021.

Matthew Jagielski, Om Thakkar, Florian Tramer, Daphne Ippolito, Katherine Lee, Nicholas Carlini, Eric Wallace, Shuang Song, Abhradeep Guha Thakurta, Nicolas Papernot, and Chiyuan Zhang. Measuring forgetting of memorized training examples. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=7bJizxLKrR.

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, 2023. URL https://openreview.net/forum?id=0jZH883i34.

Kenji Kawaguchi, Leslie Pack Kaelbling, and Yoshua Bengio. Generalization in deep learning. *arXiv preprint arXiv:1710.05468*, 1(8), 2017.

Kenji Kawaguchi, Zhun Deng, Xu Ji, and Jiaoyang Huang. How does information bottleneck help deep learning? In *International Conference on Machine Learning*, pp. 16049–16096. PMLR, 2023.

Hyunjune Kim, Sangyong Lee, and Simon S Woo. Layer attack unlearning: Fast and accurate machine unlearning via layer level attack and knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 21241–21248, 2024.

Junyaup Kim and Simon S Woo. Efficient two-stage model retraining for machine unlearning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4361–4369, 2022.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, pp. 1885–1894. PMLR, 2017.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. *Advances in neural information processing systems*, 36, 2024.

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

Xinfeng Li, Yuchen Yang, Jiangyi Deng, Chen Yan, Yanjiao Chen, Xiaoyu Ji, and Wenyuan Xu. Safegen: Mitigating unsafe content generation in text-to-image models. In *CCS*, 2024.

Junxu Liu, Mingsheng Xue, Jian Lou, Xiaoyu Zhang, Li Xiong, and Zhan Qin. Muter: Machine unlearning on adversarially trained models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4892–4902, 2023a.

Ziyao Liu, Yu Jiang, Jiyuan Shen, Minyi Peng, Kwok-Yan Lam, and Xingliang Yuan. A survey on federated unlearning: Challenges, methods, and future directions. *arXiv preprint arXiv:2310.20448*, 2023b.

Waleed Mustafa, Yunwen Lei, and Marius Kloft. On the generalization analysis of adversarial learning. In *International Conference on Machine Learning*, pp. 16174–16196. PMLR, 2022.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, pp. 4. Granada, 2011.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and A Ng. The street view house numbers (svhn) dataset. *Technical Report*, 2018.

Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. *Advances in neural information processing systems*, 30, 2017.

Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. The role of over-parametrization in generalization of neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=BygfghAcYX.

Thanh Tam Nguyen, Thanh Trung Huynh, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A survey of machine unlearning. *arXiv preprint arXiv:2209.02299*, 2022.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Chao Pan, Jin Sima, Saurav Prakash, Vishal Rana, and Olgica Milenkovic. Machine unlearning of federated clusters. *arXiv preprint arXiv:2210.16424*, 2022.

Protection Regulation. General data protection regulation. *Intouch*, 25:1–5, 2018.

Chris Rohlfs. Generalization in neural networks: A broad survey. *Neurocomputing*, 611:128701, 2025.

Andrew M Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan D Tracey, and David D Cox. On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124020, 2019.

Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22522–22531, 2023.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086, 2021.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 3–18. IEEE, 2017.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Ayush K. Tarun, Vikram S. Chundawat, Murari Mandal, and Mohan Kankanhalli. Fast yet effective machine unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10, 2023. doi: 10.1109/TNNLS.2023.3266233.

Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*, pp. 1–5. IEEE, 2015.

Eleni Triantafillou, Fabian Pedregosa, Jamie Hayes, Peter Kairouz, Isabelle Guyon, Meghdad Kurmanji, Gintare Karolina Dziugaite, Peter Triantafillou, Kairan Zhao, Lisheng Sun Hosoya, et al. Neurips 2023-machine unlearning, 2023. *URL https://kaggle. com/competitions/neurips-2023-machine-unlearning*, 2023.

Enayat Ullah, Tung Mai, Anup Rao, Ryan A Rossi, and Raman Arora. Machine unlearning via algorithmic stability. In *Conference on Learning Theory*, pp. 4126–4142. PMLR, 2021.

Fei Wang, Baochun Li, and Bo Li. Federated unlearning and its privacy threats. *IEEE Network*, 2023.

Junxiao Wang, Song Guo, Xin Xie, and Heng Qi. Federated unlearning via class-discriminative pruning. In *Proceedings of the ACM Web Conference 2022*, pp. 622–632, 2022.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020a.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020b.

Zaitian Wang, Pengfei Wang, Kunpeng Liu, Pengyang Wang, Yanjie Fu, Chang-Tien Lu, Charu C Aggarwal, Jian Pei, and Yuanchun Zhou. A comprehensive survey on data augmentation. *arXiv preprint arXiv:2405.09591*, 2024.

Chen Wu, Sencun Zhu, and Prasenjit Mitra. Federated unlearning with knowledge distillation. *arXiv preprint arXiv:2201.09441*, 2022.

Jianqiang Xiao, Weiwen Guo, Junfeng Liu, and Mengze Li. Generalization gap in data augmentation: Insights from illumination. In *International Conference on Pattern Recognition*, pp. 190–205. Springer, 2024.

Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 2521–2530, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Heng Xu, Tianqing Zhu, Lefeng Zhang, Wanlei Zhou, and Philip S Yu. Machine unlearning: A survey. *ACM Computing Surveys*, 56(1):1–36, 2023.

Yi Xu. Machine unlearning for traditional models and large language models: A short survey. *arXiv preprint arXiv:2404.01206*, 2024.

Jin Yao, Eli Chien, Minxin Du, Xinyao Niu, Tianhao Wang, Zezhou Cheng, and Xiang Yue. Machine unlearning of pre-trained large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8403–8419, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.457. URL https://aclanthology.org/2024.acl-long.457/.

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.

Xiaojian Yuan, Tianyu Pang, Chao Du, Kejiang Chen, Weiming Zhang, and Min Lin. A closer look at machine unlearning for large language models. *arXiv preprint arXiv:2410.08109*, 2024.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6023–6032, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy8gdB9xx.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021a.

Gong Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Forget-me-not: Learning to forget in text-to-image diffusion models. In *CVPR*, pp. 1755–1764. IEEE, 2024a.

Gong Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Forget-me-not: Learning to forget in text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1755–1764, 2024b.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=r1Ddp1-Rb.

Linjun Zhang, Zhun Deng, Kenji Kawaguchi, Amirata Ghorbani, and James Zou. How does mixup help with robustness and generalization? In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=8yKEo06dKNo.

Yimeng Zhang, Xin Chen, Jinghan Jia, Yihua Zhang, Chongyu Fan, Jiancheng Liu, Mingyi Hong, Ke Ding, and Sijia Liu. Defensive unlearning with adversarial training for robust concept erasure in diffusion models. *CoRR*, 2024c.

Zijie Zhang, Yang Zhou, Xin Zhao, Tianshi Che, and Lingjuan Lyu. Prompt certified machine unlearning with randomized gradient smoothing and quantization. *Advances in Neural Information Processing Systems*, 35:13433–13455, 2022.

Kairan Zhao, Meghdad Kurmanji, George-Octavian B*u*arbulescu, Eleni Triantafillou, and Peter Triantafillou. What makes unlearning hard and what to do about it. *Advances in Neural Information Processing Systems*, 37:12293–12333, 2024.

Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 13001–13008, 2020.

A Extended Related Work

Machine Unlearning. (Cao & Yang, 2015) first highlighted the importance of enabling trained models to efficiently and securely remove sensitive data without retraining from scratch, and proposed an exact forgetting algorithm for statistical query learning; (Bourtoule et al., 2021) shards data and creates multiple models, which enables the exact unlearning of certain data partitions. Various approximate unlearning methods have been proposed as a post-training procedure for efficiently and effectively scrubbing target data, including Quasi-Newton or influence-based Koh & Liang (2017); Guo et al. (2019); Golatkar et al. (2020a;b), differential privacy-based (Ginart et al., 2019; Sekhari et al., 2021; Izzo et al., 2021; Ullah et al., 2021), quantization-based (Zhang et al., 2022; Ginart et al., 2019; Che et al., 2023), relabeling-based (Graves et al., 2021; Fan et al., 2023), knowledge distillation-based (Kim & Woo, 2022; Chundawat et al., 2023; Kurmanji et al., 2024; Kim et al., 2024), and adversarial training-based (Liu et al., 2023a; Goel et al., 2022b; Zhang et al., 2024c), etc. Besides, (Tarun et al., 2023; Chen et al., 2023) proposed efficient unlearning approaches for class-level forget tasks specifically. (Fan et al., 2023; Zhang et al., 2024b; Schramowski et al., 2023; Gandikota et al., 2024; Zhang et al., 2024a; Li et al., 2024) involved unlearning for generative or diffusion models; (Pan et al., 2022; Wu et al., 2022; Wang et al., 2022; Gao et al., 2022; Wang et al., 2023; Che et al., 2023) proposed different unlearning approaches for the federated learning scenario, and recent works (Yuan et al., 2024; Yao et al., 2024; Buarbulescu & Triantafillou, 2025) sprung up to study the machine unlearning problem for large language models. We refer the interested readers to the surveys Nguyen et al. (2022); Xu et al. (2023); Liu et al. (2023b); Xu (2024) for comprehensive summarization and exhaustive discussions. Membership inference attack (MIA) introduced by (Shokri et al., 2017), different attack and defense methods are proposed (Carlini et al., 2022; Hu et al., 2022; 2023), and it is utilized to audit training data privacy in MU (Jagielski et al., 2023; Carlini et al., 2022), while only black-box attacks are majorly considered as evaluation now.

Model Generalization. Being a central focus in learning theory, generalization theory and analysis for modern machine learning algorithms have attracted great attention (Kawaguchi et al., 2017; Zhang et al., 2017; Neyshabur et al., 2019; Saxe et al., 2019; Arora et al., 2018; Zhang et al., 2021a). Among them, the information bottleneck theory (Tishby & Zaslavsky, 2015) and induced two phases of SGD optimization (Saxe et al., 2019; Shwartz-Ziv & Tishby, 2017) capture and model the trajectory and behavior of deep models, which also serve as one way to control the generalization errors (Goyal & Bengio, 2022; Kawaguchi et al., 2023). Although those enlightening works provide a glimpse of deep models' mysterious properties, the complex relationship between model generalization and machine unlearning has not been explored in depth. There are some primary explorations, (Golatkar et al., 2020a) utilized the over-parameterized neural tangent kernel networks to design machine unlearning algorithms; (Ullah et al., 2021) introduced total variation stability to achieve exact unlearning.

Various techniques have been designed to improve the generalization capabilities of deep models (Neyshabur et al., 2017; Rohlfs, 2025). For example, regularization includes weight decay (D'Angelo et al., 2023), Dropout (Helmbold & Long, 2015), and Batch Normalization Ioffe (2015) data augmentation (Wang et al., 2024; Xiao et al., 2024) consists of MixUp (Zhang et al., 2018; 2021b), CutMix (Yun et al., 2019), AutoAugment (Cubuk et al., 2019), RandErasing (Zhong et al., 2020), AugMix (Hendrycks* et al., 2020), CutOut (DeVries, 2017), and RandAugment (Cubuk et al., 2020); robust training paradigms involve adversarial training, contrastive learning (Mustafa et al., 2022), and so on.

B Theoretical Analysis

B.1 Knowledge Learning for a Well-generalized Model.

Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ be the data space and $\mathcal{P}$ an unknown distribution over $\mathcal{Z}$. Let $\mathcal{D} = (Z_1, \ldots, Z_n) \sim \mathcal{P}^{\otimes n}$ be a training dataset. A (possibly randomized) learning algorithm $\mathcal{A}$ outputs model parameters $\Theta = \mathcal{A}(\mathcal{D})$. We define the empirical and population risks as $L_{\text{emp}}(\theta, \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} \ell(\theta, Z_i)$ and $L_{\text{pop}}(\theta) = \mathbb{E}_{Z\sim\mathcal{P}}[\ell(\theta, Z)]$, the expected generalization gap is $\text{Gen} := \mathbb{E}_{\mathcal{D}}[L_{\text{emp}}(\Theta, \mathcal{D}) - L_{\text{pop}}(\Theta)]$.

Lemma 3 (Uniform stability bounds per-sample influence and generalization)**.** *Let $\mathcal{A}$ be a (possibly randomized) learning algorithm that maps $\mathcal{D}$ to parameters $\Theta = \mathcal{A}(\mathcal{D})$. Assume the loss $\ell(\theta, z) \in [0, 1]$ is bounded. Suppose $\mathcal{A}$ is β-uniformly stable, i.e., for any datasets $\mathcal{D}$,$\mathcal{D}^{(i)}$ that differ in exactly*

one sample,

$$\sup_{z\in\mathcal{Z}} \mathbb{E}\big[\ell(\mathcal{A}(\mathcal{D}), z) - \ell(\mathcal{A}(\mathcal{D}^{(i)}), z)\big] \le \beta.$$

Then, we have

1. *The expected generalization gap satisfies,* $\big|\mathrm{Gen}\big| = \big|\mathbb{E}[L_{\mathrm{emp}}(\Theta, \mathcal{D}) - L_{\mathrm{pop}}(\Theta)]\big| \le \beta$.
2. *The influence of any individual training sample is bounded:* $\Delta_i \triangleq \sup_{z\in\mathcal{Z}} \mathbb{E}\big[\ell(\mathcal{A}(\mathcal{D}), z) - \ell(\mathcal{A}(\mathcal{D}^{\setminus i}), z)\big] \le 2\beta$.

Proof: This is a standard result in algorithmic stability. The generalization bound follows from a telescoping argument over sample replacements, and the influence bound follows from the triangle inequality applied to the stability definition. See Bousquet & Elisseeff (2002) for details.

Theorem 2. *Let $\mathcal{A}$ be a learning algorithm producing parameters $\Theta = \mathcal{A}(\mathcal{D})$ from $\mathcal{D} \sim \mathcal{P}^{\otimes n}$ with bounded loss. For $\varepsilon \in (0, 1)$, assume that at least one of the following conditions holds:*

- ***(T1) Uniform stability:*** *$\mathcal{A}$ is β-uniformly stable with $\beta \le c_1\varepsilon$ for some absolute $c_1 > 0$.*
- ***(T2) Smoothness and local density:*** *the conditional label function $\eta(x) = \mathbb{P}(Y = 1 \mid X = x)$ is L-Lipschitz w.r.t. metric d on $\mathcal{X}$: $|\eta(x) - \eta(x')| \le Ld(x, x')$; and for every x in the support of $\mathcal{P}_X$ there exists a radius $r > 0$ such that $0 < \rho \le \mathcal{P}_X(B(x, r))$ with a ball $B(x, r)$. Moreover, the effective sample size in each neighborhood exceeds the uniform convergence threshold of the hypothesis class.*
- ***(T3) Representation prior:*** *there exists a (given or prelearned) representation $\phi : \mathcal{X} \to \mathbb{R}^k$ and a low-complexity function class $\mathcal{G}$ such that $\eta(x) = g(\phi(x))$ for some $g \in \mathcal{G}$, and the effective sample complexity in representation space satisfies the usual uniform-convergence condition for error ε.*

If the expected generalization gap satisfies $\mathbb{E}\big[L_{\mathrm{emp}}(\Theta, \mathcal{D}) - L_{\mathrm{pop}}(\Theta)\big] \le \varepsilon$, then the learned model exhibits only weak dependence on any individual training sample. In particular:

1. *(Bounded per-sample influence): For a constant $C > 0$ depending only on the chosen assumption,*
$$\max_{i\in[n]} \Delta_i \le C \cdot \varepsilon.$$
2. *(Small per-sample information): Let $I(\Theta; Z_i)$ be mutual information and $C' > 0$ is an absolute constant*
$$\frac{1}{n}\sum_{i=1}^{n} I(\Theta; Z_i) \le C' \cdot \varepsilon^2.$$

Consequently, the learned parameters Θ are dominated by distribution-level information*, i.e., shared knowledge among training samples, rather than sample-specific memorization.*

Proof: Under assumption **T**1, Lemma 3 implies $\Delta_i \le 2\beta \le 2c_1\varepsilon$. Standard information–generalization inequalities (Xu & Raginsky, 2017; Feldman, 2020) then yield $I(\Theta; Z_i) \le O(\beta^2)$ on average. Under assumptions **T**2 or **T**3, uniform convergence holds locally or in representation space, implying that prediction performance is determined by shared structure rather than the identity of individual samples. Hence removing or replacing a single sample produces only $O(\varepsilon)$ changes in loss, which again yields the stated influence and mutual-information bounds.

Corollary B.1 (Necessity of Memorization for Long-Tail or Isolated Subclasses)**.** *Let $C \subset \mathcal{Z}$ be a subclass with probability mass $\mu(C)$. If the expected number of training samples $n\mu(C)$ is below the uniform-convergence threshold required to achieve error ε for the hypothesis class, and none of the structural assumptions in Theorem 2 hold on C, then any learning algorithm achieving population error $\le \varepsilon$ on C must rely on memorization.*

Formally, there exists at least one $Z_i \in C$ such that

$$\Delta_i = \Omega(1) \quad \textit{or} \quad I(\Theta; Z_i) = \Omega(1).$$

Proof: By contraposition of Theorem 2. When the effective sample size in C is insufficient and no smoothness, density, or representation prior holds, uniform convergence fails on C. Thus accurate prediction on C requires fitting the observed samples themselves, implying large per-sample influence or information.

B.2 Model Phase Transition.

Let $M(\mathbf{w})$ be the neural network with parameters $\mathbf{w} \in \mathbb{R}^d$. Assume that $\bar{\mathbf{\Theta}}$, $\mathbf{\Theta}^*$, and $\mathbf{\Theta}^u$ correspond to the weight parameter of the original model, optimal (gold standard) model, and unlearned model. $h_l(\mathbf{x})$ denotes the activation at layer l for input $\mathbf{x}$ and $\mathcal{L}(\mathbf{x}; \mathbf{w})$ is the loss function of $M(\mathbf{w})$ for $\mathbf{x}$. $\mathcal{D}$ is the training data of size n for $M(\bar{\mathbf{\Theta}})$; $\mathcal{D}_f$ and $\mathcal{D}_r$ are the forget set of size n_f and retain set of size n_r, resp. $\mathcal{D}_t$ is the test set.

Given $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n \subseteq \mathcal{X} \times \mathcal{Y}$ consist of n samples, drawn independently and identically distributed (I.I.D.); the empirical distribution $\hat{p}(\mathbf{x}, y)$ is used to approximate $p(\mathbf{x}, y)$ for $\forall(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$: $\hat{p}(\mathbf{x}, y) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}((\mathbf{x}_i, \mathbf{y}_i) = (\mathbf{x}, y))$ with the indicator function $\mathbb{I}(\cdot)$; thus, the (empirical) entropy of $\mathcal{D}$ is defined as $\hat{H}(\mathcal{D}) = -\sum_{(\mathbf{x},y)\in\mathcal{X}\times\mathcal{Y}} \hat{p}(\mathbf{x}, y) \log \hat{p}(\mathbf{x}, y)$ to quantify the uncertainty based on the observed data.

Lemma 4 (Information Persistence)**.** *Let $I(\mathbf{w}; \mathcal{D}_f|\mathcal{D}_r)$ be the conditional mutual information between model parameters and forget data given retain data. Then,*

$$I(\bar{\mathbf{\Theta}}; \mathcal{D}_f|\mathcal{D}_r) \geq C_1 \frac{n_f}{n} H(\mathcal{D}_f), \tag{4}$$

where $C_1 > 0$ is a constant. $H(\mathcal{D}_f)$ is the entropy of $\mathcal{D}_f$.

Proof: By data processing inequality, $I(\bar{\mathbf{\Theta}}; \mathcal{D}_f|\mathcal{D}_f) \geq I(h_l; \mathcal{D}_f|\mathcal{D}_r)$ for any layer l, which means that the information about $\mathcal{D}_f$ stored in $\bar{\mathbf{\Theta}}$ is at least as much as that stored in the intermediate representations h_l.

For the optimal layer l^* in model $M(\bar{\mathbf{\Theta}})$,

$$I(h_{l^*}; \mathcal{D}_f|\mathcal{D}_r) \geq \frac{n_f}{n} H(\mathcal{D}_f) - D_{KL}(P(h_{l^*}|\mathcal{D}_f)||P(h_{l^*}|\mathcal{D}_r)).$$

Therefore, by training convergence, we have

$$D_{KL}(P(h_{l^*}|\mathcal{D}_f)||P(h_{l^*}|\mathcal{D}_r)) \leq (1 - C_1)\frac{n_f}{n} H(\mathcal{D}_f).$$

■

Lemma 5 (Representation Entanglement)**.** *For any layer l in $M(\bar{\mathbf{\Theta}})$, the activation distributions satisfy*

$$\|P(h_l|\mathcal{D}_r) - P(h_l|\mathcal{D}_r \setminus \mathcal{D}_f)\|_1 \geq C_2 \frac{n_f}{n}, \tag{5}$$

where $C_2 > 0$ is a constant.

Proof: The change in activations caused by removing $\mathcal{D}_f$ can be modeled with the triangular inequality as

$$\|P(h_l|\mathcal{D}_r) - P(h_l|\mathcal{D}_r \setminus \mathcal{D}_f)\|_1 \geq \|P(h_l|\mathcal{D}) - P(h_l|\mathcal{D}_r)\|_1 - \|P(h_l|\mathcal{D}_f)\|_1.$$

Through optimization during training, the distributional change cased by $\mathcal{D}_f$ is proportional to $\frac{n_f}{n}$, that is,

$$\|P(h_l|\mathcal{D}) - P(h_l|\mathcal{D}_r)\|_1 \geq C_2 \frac{n_f}{n},$$

which signifies the significant influences on the model's internal representation for removing $\mathcal{D}_f$.

Consequently, the network capacity ensures that

$$\|P(h_l|\mathcal{D}_f)\|_1 \leq (1 - C_2)\frac{n_f}{n},$$

which bounds the magnitude of learned entanglement from the forget set $\mathcal{D}_f$. ■

Lemma 6 (Update Insufficiency)**.** *For any parameter update δ with $\|\delta\| \leq \varepsilon_0$ for the model w.r.t. the weights $\mathbf{w}$ inheriting from $\bar{\mathbf{\Theta}}$:*

$$\|P(M(\mathbf{w}+\delta)|\mathcal{D}_r) - P(M(\mathbf{\Theta}^*)|\mathcal{D}_r)\|_1 \geq \|P(M(\mathbf{w})|\mathcal{D}_r) - P(M(\mathbf{\Theta}^*)|\mathcal{D}_r)\|_1 - C_3 \frac{\varepsilon_0}{\sqrt{n_r}}, \tag{6}$$

where $C_3 > 0$ is a constant. ■

Proof: Assume that network $M(\mathbf{w})$ is *L-Lipschitz* for a constant L, that is, for any input $\mathbf{x}$,

$$\|P(M(\mathbf{w}+\delta)|\mathbf{x}) - P(M(\mathbf{w})|\mathbf{x})\|_1 \leq L \cdot \|\delta\|,$$

which bounds the update to the model parameters.

Consider averaging the output distribution difference over the retain set $\mathcal{D}_r$. According to the central limit theorem, the total variation in the distribution is bounded by,

$$\|P(M(\mathbf{w}+\delta)|\mathcal{D}_r) - P(M(\mathbf{w})|\mathcal{D}_r)\|_1 \leq \frac{L\sigma\|\delta\|}{\sqrt{n_r}},$$

where $\sigma > 0$ quantifies the variability or spread in the updates δ across individual samples $\mathbf{x}$.

Applying the triangular inequality to the update process ensures that the distributional difference (hypothesize about change) after an update is bounded by

$$\|P(M(\mathbf{w}+\delta)|\mathcal{D}_r) - P(\mathbf{\Theta}^*|\mathcal{D}_r)\|_1 \geq \|P(M(\mathbf{w})|\mathcal{D}_r) - P(\mathbf{\Theta}^*|\mathcal{D}_r)\|_1 - \frac{L\sigma\|\delta\|}{\sqrt{n_r}}.$$

■

Lemma 7 (Accuracy Gap)**.** *For any intermediate model M_t during unlearning from $M(\bar{\mathbf{\Theta}})$ to $M(\mathbf{\Theta}^u)$:*

$$|Acc(M_t(\mathcal{D}_r)) - Acc(M(\mathcal{D}_t;\mathbf{\Theta}^*))| \geq \ C_4\|P(M_t|\mathcal{D}_r) - P(M(\mathbf{\Theta}^*)|\mathcal{D}_r)\|_1, \tag{7}$$

where $C_4 > 0$ is a constant.

Proof: Based on the definition of accuracy, it holds that

$$\begin{aligned} & |Acc(M_t(\mathcal{D}_r)) - Acc(M(\mathcal{D}_t;\mathbf{\Theta}^*))| \\ = \ & \mathbb{E}_{\mathbf{x}\sim P(\mathbf{x}|\mathcal{D}_r)}[\mathbb{I}(M_t(\mathbf{x}) = y)] - \mathbb{E}_{\mathbf{x}\sim P(\mathbf{x}|\mathcal{D}_t)}[\mathbb{I}(M(\mathbf{x};\mathbf{\Theta}^*) = y)]| \\ \geq \ & C_4 \cdot \|P(M_t|\mathcal{D}_r) - P(M(\mathbf{\Theta}^*)|\mathcal{D}_r)\|_1, \end{aligned}$$

where the last inequality is derived from the total variation distance. Specifically,

$$C_4 \approx \min_{(\mathbf{x},y)\in\mathcal{D}_r} \frac{|P(M_t(\mathbf{x}) = y) - P(M(\mathbf{x};\mathbf{\Theta}^*) = y)|}{\|P(M_t(\mathbf{x})) - P(M(\mathbf{x};\mathbf{\Theta}^*))\|_1}.$$

■

B.2.1 Main Theorem

Theorem 3. *For $n_f = \alpha \cdot n$, there exists no sequence of bounded updates $\{\delta_i\}_{i=1}^k$ with $k < K = \Omega(\sqrt{n})$ that can transform $\bar{\mathbf{\Theta}}$ to $\mathbf{\Theta}^u$ while satisfying:*

$$|Acc(M(\mathcal{D}_r;\mathbf{\Theta}^u)) - Acc(M(\mathcal{D}_t;\mathbf{\Theta}^*))| \leq \varepsilon, \tag{8}$$

while ε is a small constant indicating the desired accuracy of $\mathbf{\Theta}^u$ after unlearning.

Proof: For the initial gap: the total variation distance between the distribution of $M(\bar{\mathbf{\Theta}})$ and $M(\mathbf{\Theta}^*)$ is bounded as, by Lemma 4-5, as the initial gap,

$$\|P(M_{\bar{\mathbf{\Theta}}}|\mathcal{D}_r) - P(M_{\mathbf{\Theta}^*}|\mathcal{D}_r)\|_1 \geq C_5\frac{n_f}{n}.$$

Consider a sequence of k updates to the model parameters for the update sequence analysis. After k updates, the gap in predicted distribution between M_k and $M(\mathbf{\Theta}^*)$ is bounded as follows by repeated application of Lemma 6,

$$\|P(M_k|\mathcal{D}_r) - P(M(\mathbf{\Theta}^*)|\mathcal{D}_r)\|_1 \geq C_5\frac{n_f}{n} - \frac{kC_3\varepsilon_0}{\sqrt{n_r}}.$$

To achieve the desired accuracy ε between $M(\mathbf{\Theta}^u)$ and $M(\mathbf{\Theta}^*)$, the following condition must be satisfied by Lemma 7:

$$C_5\frac{n_f}{n} - \frac{kC_3\varepsilon_0}{\sqrt{n_r}} \leq \frac{\varepsilon}{C_4}$$

Solving for k:

$$k \geq \frac{(C_5 \frac{n_f}{n} - \frac{\varepsilon}{C_4})\sqrt{n_r}}{C_3 \varepsilon_0} = \frac{\alpha C_5 - \epsilon / C_4}{C_3 \varepsilon_0} \sqrt{(1-\alpha)n}.$$

Thus, $k = \Omega(\alpha\sqrt{n})$, which reflects the difficulty of reducing the distributional difference caused by persistent dependencies on $\mathcal{D}_f$.

Additionally, consider the complexity of the unlearning process and the multiplicative nature of updates in iterative optimization (i.e., iterative convergence). As shown by optimization theory, reducing a function from an initial value to a target value requires many iterations that are at least logarithmic in the ratio of the initial and target values. Consequently, to meet the desired accuracy ϵ, the unlearning process must reduce the distributional difference from $\Delta = C_5\alpha$ to ϵ, $\Omega(\log(\Delta/\epsilon))$ iterations are required, where Δ is proportional to α and scale with N; if each update reduces the difference multiplicatively by a factor $c < 1$, typically in optimization, $k \geq \frac{\log(\Delta/\epsilon)}{1/c} = \Omega(\log(\alpha n))$ for a larger dataset size n.

Therefore, $k \geq \max(\Omega(\sqrt{n}), \Omega(\log n)$ and $k \geq \Omega(\sqrt{n})$ for large datasets ($n \gg 1$). The Lipschitz continuity, spectral properties, and gradient norms of the model also determine the effectiveness of each update. ■

Theorem 4. *Assume the loss function $\mathcal{L}_{\mathcal{D}}(\mathbf{\Theta}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[\ell(\mathcal{F}(\mathbf{x};\mathbf{\Theta}), y)]$ is μ-strongly convex and L-smooth around the local minima $\mathbf{\Theta}^*$, the model parameters of the well-generalizable model and overfitting model are $\mathbf{\Theta}_0^{gen}$ and $\mathbf{\Theta}^{\text{overfit}}$ respectively. The number of the SGD iterations T satisfies $T_{gen} \ll T_{\text{overfit}}$.*

Proof: Using gradient descent as optimizer, i.e., $\mathbf{\Theta}_{t+1} = \mathbf{\Theta}_t - \eta\nabla\mathcal{L}(\mathbf{\Theta}_t)$, the distance to the optimum satisfies, $||\mathbf{\Theta}_t - \mathbf{\Theta}^*||^2 \leq (1-2\eta\mu)^t ||\mathbf{\Theta}_0 - \mathbf{\Theta}^*||^2$.

If $\mathbf{\Theta}_0 = \mathbf{\Theta}_0^{gen}$, we assume that $||\mathbf{\Theta}^{gen} - \mathbf{\Theta}^*||^2 \ll ||\mathbf{\Theta}^{\text{overfit}} - \mathbf{\Theta}^*||^2$, then fewer iterations are required to achieve $||\mathbf{\Theta}_t - \mathbf{\Theta}^*|| \leq \epsilon$. Hence, the number of steps T required satisfies:

$$T_{gen} = \mathcal{O}\left(\frac{1}{\eta\mu}\log\left(\frac{1}{\epsilon}\right)\right) \ll T_{\text{overfit}}.$$

■

B.3 Effectiveness of GenUn-F.

Proposition 2. *Assume the DNN model $\mathcal{F} = h \circ f_{\mathbf{\Theta}}$, where $f_{\mathbf{\Theta}} : \mathcal{X} \to \mathbb{R}^d$ is a neural encoder with parameter $\mathbf{\Theta}$, and $h : \mathbb{R}^d \to \mathcal{Y}$ is a classifier. The updated model $f_{\mathbf{\Theta}'}$ obtained from* GenUn-F *will improve the generalization bound over $\mathcal{D}_t$ through the contrastive learning.*

Proof: A general PAC-Bayes bound with compression and stability arguments for an encoder-based model yields, $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}[\ell(h \circ f_{\mathbf{\Theta}'}(\mathbf{x}), y)] \leq \mathcal{L}_{\mathcal{D}_r} + \mathcal{R}_{\text{eff}} + \mathcal{O}\left(\sqrt{\frac{\log(1/\delta)}{|\mathcal{D}_r|}}\right)$, where $\mathcal{R}_{\text{eff}}$ is an effective capacity or representation complexity term and δ is a confidence parameter.

Following the theoretical analysis from (Wang & Isola, 2020b), the contrastive loss controls two properties of the representation:

- **Alignment**: Closeness of prostive pairs in embedding space as $\text{Align}(f) := \mathbb{E}_{(\mathbf{x},\mathbf{x}^+)}\left[||f(\mathbf{x}) - f(\mathbf{x}^+)||_2^2\right]$
- **Uniformity**: Spreading of all features uniformly on the hypersphere as $\text{Unif}(f) := \log \mathbb{E}_{(\mathbf{x},\mathbf{x}')}\left[\exp\left(||f(\mathbf{x}) - f(\mathbf{x}')||^2\right)\right]$

The InfoNCE loss upper bounds $\mathcal{L}^{cont} \geq \text{Align}(f) + \lambda \cdot \text{Unif}(f)$. Therefore, minimizing $\mathcal{L}^{cont}$ reduces both alignment and uniformity loss, which makes similar samples close (semantic preservation) and unrelated samples well separated (deconfounding and robustness).

If $f_{\mathbf{\Theta}}$ is learned via the joint loss $\min_{\mathbf{\Theta}} \mathcal{L}_r + \lambda\mathcal{L}_f^{cont}$, then the encoder $f_{\mathbf{\Theta}}$ learns generalizable features from $\mathcal{D}_r$, and the model relies less on specific samples in $\mathcal{D}_f$ since the contrastive loss maximizes inter-sample structure. Thus, the sensitivity of $f_{\mathbf{\Theta}}$ to the presence of samples in $\mathcal{D}_f$ can be

bounded by $||f_{\mathbf{\Theta}}(\mathbf{x}) - f_{\mathbf{\Theta}'}(\mathbf{x})|| \leq \epsilon_{\text{forget}}$ for $\mathbf{x} \in \mathcal{D}_t$ with $\epsilon_{\text{forget}} \propto \frac{|\mathcal{D}_f|}{|\mathcal{D}_r|} \cdot \frac{1}{\lambda}$. Thus, higher λ means better retention of generation under the unlearning setting.

So, putting it together, we have

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_t}\left[\ell(h \circ f_{\mathbf{\Theta}'}(\mathbf{x}), y)\right] \leq \mathcal{L}_{\mathcal{D}_r} + \underbrace{\mathcal{R}_{\text{eff}}(\text{Align}, \text{Unif})}_{\text{controlled by contractive}} + \underbrace{\mathcal{O}\left(\epsilon_{\text{forget}}\right)}_{\text{forget influence}} .$$

Therefore, the generalization error is controlled by the empirical loss on $\mathcal{D}_r$, the complexity of the learned representation (low due to contrastive loss), and the stability under unlearning, i.e., enhanced by contrastive training's regularization.

C Machine unlearning Methods

We include and compare against various state-of-the-art machine unlearning methods, which are briefly summarized as

- **Original**: the model trained on all data $\mathcal{D}$ without performing any unlearning.
- **Retrain**: retraining the model from scratch without the forget set, i.e., only $\mathcal{D}_r$, which is assumed not to be viable in practice but included as a reference.
- **Fine-tuning**: finetune the original model on retain set.
- **Gradient Ascent** (Golatkar et al., 2020b): finetune the original model on forget set $\mathcal{D}_f$ with negating the gradient.
- **Random Labels** (Graves et al., 2021; Hayase et al., 2020): finetune the original model on the random relabelled forget data.
- **Fisher Forgetting** (Golatkar et al., 2020b): first locate the most relevant parameters in terms of the forget data and then scrub them by adding noise.
- **Influence Unlearning** (Koh & Liang, 2017; Izzo et al., 2021): utilize an influence function to estimate the updates required for the model weights as a result of removing $\mathcal{D}_f$ from $\mathcal{D}$.
- **EU-k** (*Exact Unlearning-k*) and **CF-k** (*Catastrophic Forgetting-k*) (Goel et al., 2022a): freeze the first k layers of the original model and either train the remaining layers from scratch on $\mathcal{D}_r$ or finetune the remain layers on $\mathcal{D}_r$.
- ℓ_1**-sparse** (Jia et al., 2023): introduce ℓ_1-norm for model weights and fine-tune it over the retain data.
- **SalUn** (Fan et al., 2023): draw a parallel with input saliency in model explanation to only update influential weights filtered by the threshold, and integrate with Random Label.
- **SCRUB** (Kurmanji et al., 2024): move the student away from the teacher and formulate a min-max bi-optimization problem by extending the contrastive learning.

Implementation. Our methods and other baselines are implemented in Python 3.11 with PyTorch Library. All experiments are performed on NVIDIA Tesla-A100 (80G) with Intel Xeon processors.

D Experiment Setup

Dataset. The basic information about the dataset we used is summarized as follows.

- *CIFAR-10*: CIFAR-10 (Krizhevsky et al., 2009) contains $60,000$ colored images of size $32 \times 32 \times 3$ with $10,000$ reserved for testing. There are 10 target classes with $5,000$ training images per class.
- *CIFAR-100*: CIFAR-100 (Krizhevsky et al., 2009) contains $60,000$ colored images with $10,000$ reserved for testing. There are 100 target classes with 500 training images per class.
- *SVHN*: SVHN (Netzer et al., 2011) is obtained from house numbers in Google Street View images, and the underlying task is of digital classification from 32×32 colored images. (73257 digits for training, 26032 digits for testing, same image size as CIFAR-10).
- *Tiny-ImageNet*: Tiny ImageNet (Le & Yang, 2015) contains 120,000 colored images of size $64 \times 64 \times 3$ with 10,000 reserved for validation and 10,000 reserved for testing. There are 200 target classes with 500 training images per class.

For the *random-subset forgetting task*, we choose to forget about $\approx 10\%$ of the training data. Therefore, for the CIFAR-10 and CIFAR-100 we have $|\mathcal{D}_f| = 5K$, and the sizes of train set $\mathcal{D}$, test

set $\mathcal{D}_t$, and validation set $\mathcal{D}_v$ correspond to $45K$, $10K$, $5K$ respectively; for the SVHN, we have $|\mathcal{D}_f| = 7K$, and the sizes of train set $\mathcal{D}$, test set $\mathcal{D}_t$, and validation set $\mathcal{D}_v$ correspond to 66257, 26032, $7K$ respectively; for the Tiny-ImageNet, we have $|\mathcal{D}_f| = 10K$, and the sizes of train set $\mathcal{D}$, test set $\mathcal{D}_t$, and validation set $\mathcal{D}_v$ correspond to $90K, 10K, 10K$, respectively.

Model. We adopt the same models with the same architectural modifications as described in (Golatkar et al., 2020b;a). For All-CNN (Springenberg et al., 2014), we reduce the number of layers and introduce batch normalization before each non-linearity layer, training the model from scratch. For ResNet, we utilize the ResNet-18 architecture, starting with the pre-trained PyTorch model and fine-tuning it as the original model. In addition, following (Kurmanji et al., 2024), we scale up to a larger setting by choosing larger models and tackling a larger classification task, that is, we used the pre-trained VGG-16 (Simonyan & Zisserman, 2014) and ViT (Dosovitskiy et al., 2021) models, finetuning them on the CIFAR-100.

Hyperparameter setting.

- **Original**: the model is trained on the entire dataset $\mathcal{D}$.
- **Retrain**: we train the same architecture on $\mathcal{D}_r$ with the same hyperparameters used as in the original model.
- **Finetune**: fine-tune the original model for 10 epochs with a fixed learning rate $lr = 0.001$.
- **Gradient Ascent** and **Random Labels**: we trained with 15 epochs with Adam, $lr = 5E-5$. Specially, $lr = 1E-6$ for the ViT model.
- **Fisher Forgetting**: we set $\lambda = 1E-8$ for the random-subset forgetting, $1e-7$ for the class-wise forgetting.
- **Influence Function**: we unfreeze the last-2 layers' weights and set the hyperparameters as, damp = 0.01, scale = 25, recursion_depth = 10, and r_averaging = 1.
- **EU-k**: we re-initialize the weights of the last layer of original model and retrain with $\mathcal{D}_r$ for 20 epochs with $lr = 0.001$.
- **SCRUB**: we set the hyperparameters to keep consistent with the original paper used: epochs = 5, decay_rate = 0.1, decay_layer = [3, 5, 9], $lr = 1E-4$, $\gamma = 1$, $\alpha = 0.8$ (random-subset forgetting on CIFAR-10), 0.1 (class-wise forgetting on CIFAR-10) and 0.5 (SVHN), m_steps = 2, forget-set_bs = 64, retain-set_bs=256, and temperature $T = 4$ for KD. Specially, $lr = 0.0005$ for VGG-16 and $lr = 1E-5$ for ViT.

We also test different parameters for the above baselines, including k for 'EU-k', λ for 'Fisher Forgetting', $recursion_depth$, 'last-k', and $r_averaging$ for 'Influence function', and parameters for other methods. We reported the best results.

For our proposed methods, we typically set the epochs to 10, and the learning rate is chosen from [1e-5, 2e-3]; the temperature $\tau = 3$ and coefficient factor $\alpha = 5$ in default. For the MoCo used in GENUN-F, the queue size is 1024, the feature projection dimension is 512, and the momentum factor is set to be 0.9.

LiRA setup. To evaluate the privacy protection effectiveness of machine unlearning algorithms, we implemented the U-LiRA membership inference attack based on the log-likelihood ratio, which is considered the strongest per-sample attack benchmark for unlearning scenarios to date. The specific attack procedure is as follows:

- **Distribution Construction and Sample Filtering:** For each sample in the restricted sampling pool, we iterated over all shadow models and filtered them based on the sample's membership status in the models. Specifically, we collected the model outputs when the sample was in the forget set to construct the f_{out} distribution, and collected the model outputs when it was in the unseen sample set to construct the f_{in} distribution. Thanks to the aforementioned restricted sampling strategy, each target sample obtained a sufficient number of shadow model observations ($N = 128$) in both states, meeting the statistical requirements for precise Gaussian distribution fitting.
- **Attack Scoring:** For each target variant model, we computed the log-likelihood ratio of the query sample under the two Gaussian distributions as the attack score.
- **Evaluation Metrics:**We conducted evaluations within the fixed pool, selecting samples that were neither in the forget set nor in the retain set as unseen test samples. To ensure unbiased

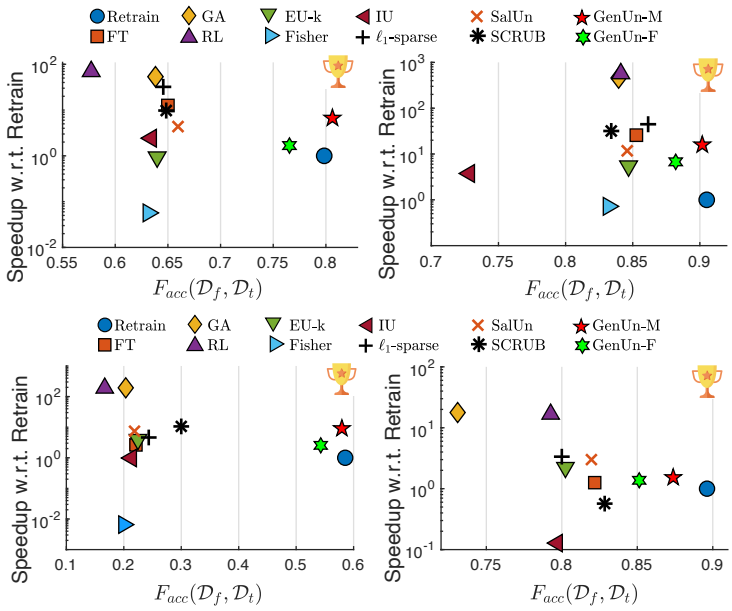


Figure 7: Comparison of MU methods about Speedup w.r.t. Retrain versus F_{acc} for four types of models. From left to right: ResNet-18 & All-CNN (Top), VGG-16 & ViT (Bottom).

evaluation metrics, we strictly implemented a 1:1 downsampling balancing strategy for positive and negative samples when calculating the AUC and attack accuracy.

We conducted experiments on the CIFAR-10 dataset, employing a standard ResNet-18 as the model architecture. To simulate realistic scenarios and construct distributions for membership inference attacks, we followed (Hayes et al., 2025) and adopted a training data subsampling strategy. Specifically, we randomly sampled 25,000 samples (i.e., half of the total training set) from the CIFAR-10 training set as the training set for each model. For the original base model, we fine-tuned the pre-trained ResNet-18 for 15 epochs, achieving an average test set accuracy of approximately 73%.

To implement the U-LiRA attack, we partitioned the entire set of base models into two independent subsets: 128 shadow models and 128 target models. The shadow models were used to estimate the parameters of the confidence distributions for members and non-members, while the target models served as the attack targets for evaluating the attack accuracy.To ensure statistical significance even with a limited number of variants, we introduced a restricted sampling pool mechanism. We pre-sampled a fixed pool of 1,000 samples from 5-th class of CIFAR-10. Unlike the original U-LiRA setup, which involved large-scale repeated sampling for each base model, we adopted a more computationally efficient strategy: when constructing the forget set for each variant, we restricted sampling to 200 samples from the intersection of the fixed pool and the base model's training set. For each base model, we randomly sampled 5 different forget sets, generating 5 variant models, resulting in a total of 1,280 trained models. This strategy significantly increased the probability that samples from the fixed pool appear in the forget sets of various variants, ensuring that even with a reduced total number of models, specific samples still obtain sufficient observations to accurately fit the distributions.

E More Experimental Results.

More MIA result. Fig. 8-9 give the ASR results of MI for different attackers on the ResNet-18 and All-CNN models.

Result over large-scale dataset. Table 2 reports the results over Tiny-ImageNet dataset. For Tiny-ImageNet, we fine-tuned a pre-trained ResNet-50 model using Adam with a fixed learning rate of 1e-4, momentum of 0.9, and batch size of 512. We can see that the conclusion also remains

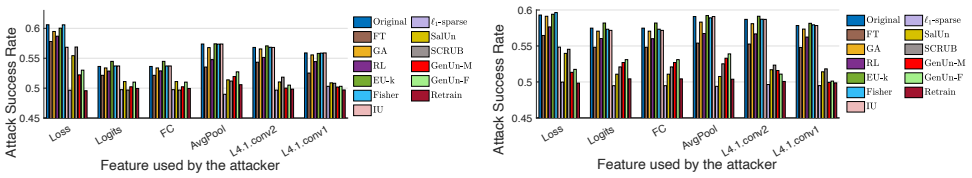


Figure 8: ASR of MIA with LR (Left) and RF (Right) as the attackers for the audit data constructed with different features of the model's last layers for ResNet-18 on CIFAR-10.

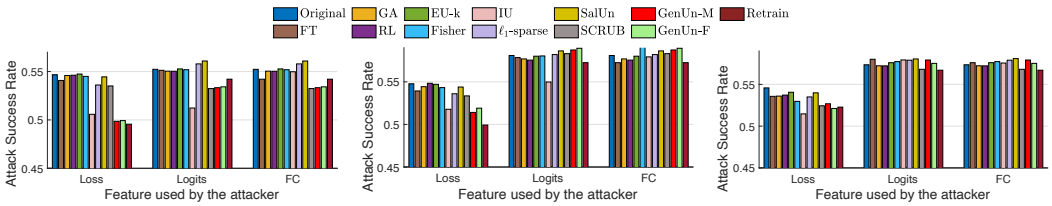


Figure 9: ASR of MIA with LR (Left), SVM (Middle), and RF (Right) as the attackers for the audit data based on 'Loss', 'Logits', and 'FC' layer output for All-CNN on SVHN.

consistent with the small-scale dataset. GENUN-M and GENUN-F achieve the lowest $|$ Diff $|$ and comparable $Acc(\mathcal{D}_t)$; it obtains more significant speedup compared with Retrain and SCRUB for the mid-size dataset. Notably, GENUN-M requires only 302.84s, which is approximately $3\times$ faster than SCRUB. Along with the results, we believe GENUN-M will have a greater advantage in terms of efficiency for larger datasets over other MU methods, and it scales well.

Unlearning Score Comparison. Following the definition of unlearning score in Kim et al. (2024),

$$US(Acc(\mathcal{D}_r), Acc(\mathcal{D}_f)) = \frac{\exp(\frac{Acc(\mathcal{D}_r)}{100}) + \exp(1 - \frac{Acc(\mathcal{D}_f)}{100}) - 2}{2 \cdot (\exp(1) - 1)},$$

Table 3 compares results for different models and unlearning methods when using US as the metric. GENUN algorithms (GENUN-M and GENUN-F) achieve the best or the second-best results here for almost allmodels, and outperform other baselins. SCRUB obtains the second-best for the Vit model.

Fig. 10 shows the best balanced accuracy of the U-LiRA in distinguishing between the forget set and the test set. The accuracy is derived from the attacker's ROC curve (as shown in Fig. 4). We sweep across all possible decision thresholds on the U-LiRA score and report the maximum achievable accuracy for each unlearning method. This metric represents the capability of an optimal adversary who selects the best threshold to maximize membership inference success. Consistent with the results shown in Fig. 4, our method significantly outperforms other baselines. **LiRA Accuracy.**

Performance of GENUN-M applied to models at different training epochs. To further evaluate the robustness of the GENUN-M algorithm across different model states, we conducted additional experiments regarding the sensitivity to training epochs. Specifically, we selected the SVHN dataset for this experiment, where the size of the forget set $|D_f|$ is set to 7,000.

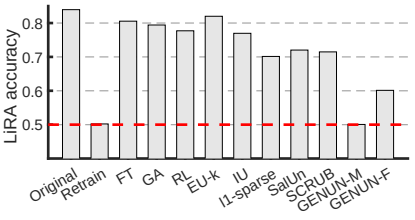


Figure 10: The best balanced accuracy of the U-LiRA attack in distinguishing between the forget set and test set. The red dashed line represents random guessing.

As shown in Fig. 11, we illustrate the variation of the hyperparameter α and the accuracy on the D_r, D_f, and D_t achieved by GENUN-M when applied to models at different training states k. The results demonstrate that GENUN-M is relatively stable concerning the degree of model training. As observed, the parameter α (indicated by the purple line on the right y-axis) maintains a stable

Table 2: Performance comparison of various MU methods for random-subset forgetting w.r.t. **fidelity** ($Acc(\mathcal{D}_r)$), **efficacy** ($Acc(\mathcal{D}_f)$), **generalization ability** ($Acc(\mathcal{D}_t)$ and $|$ Diff $|$) and **computation efficiency** (Time) for the ResNet50 models on Tiny-ImageNet. Fisher is excluded due to a runtime timeout.

Dataset(model)	Metric	Original	Retrain	FT	GA	RL	EU-k	IU	ℓ_1-sparse	SalUn	SCRUB	$\widehat{\Theta}_k$	GENUN-M	GENUN-F
Tiny-ImageNet (ResNet-50)	$Acc(\mathcal{D}_r)\uparrow$	99.39	99.94	99.74	98.79	83.81	99.96	99.39	99.56	98.98	99.81	74.77	94.49	94.26
	$Acc(\mathcal{D}_t)\uparrow$	78.74	78.34	77.51	78.14	66.67	78.57	78.75	77.69	77.54	78.90	69.99	77.22	73.43
	$Acc(\mathcal{D}_f)$	99.36	78.39	97.90	97.74	83.48	99.38	99.35	98.74	97.30	99.11	74.54	78.08	76.70
	$\lvert$Diff$\rvert\downarrow$	20.62	0.05	20.39	19.60	16.81	20.81	20.60	21.05	19.76	20.21	4.55	**0.86**	3.27
	Time (s) $\downarrow$	–	1027.23	354.36	12.52	12.32	396.83	52.84	83.00	93.60	988.39	–	302.84	809.34

Table 3: Performance comparison of various MU methods for random-subset forgetting with Unlearning Score (US) Kim et al. (2024), Fisher is excluded due to a runtime timeout for ViT and ResNet-50.

Model	Original	Retrain	FT	GA	RL	EU-k	Fisher	IU	ℓ_1-sparse	SalUn	SCRUB	GENUN-M	GENUN-F
ResNet-18		0.5491	0.4985	0.4914	0.4694	0.4951	0.4918	0.4356	0.5125	0.5117	0.4857	**0.5424**	0.5480
All-CNN		0.5109	0.4960	0.4916	0.4908	0.4930	0.4864	0.4264	0.4922	0.4920	0.4796	**0.5039**	0.5069
VGG-16		0.6450	0.5041	0.5004	0.4918	0.4995	0.4964	0.4981	0.5079	0.5011	0.5208	0.6180	**0.6043**
ViT		0.5073	0.4997	0.4575	0.4891	0.4895	–	0.4887	0.4861	0.4987	**0.5007**	0.5033	0.4924
ResNet-50		0.5697	0.5041	0.4971	0.4340	0.5015	–	0.4971	0.5002	0.4999	0.5011	**0.5289**	0.5322

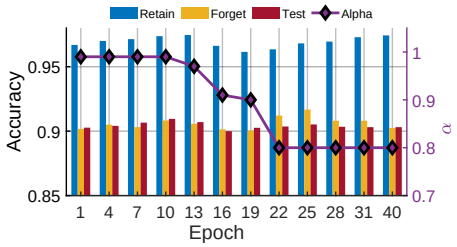


Figure 11: Performance of the GENUN-M when applied to models at different training epochs on SVHN. The bars represent the accuracy on different subsets (left y-axis), while the purple line indicates the value of α (right y-axis).

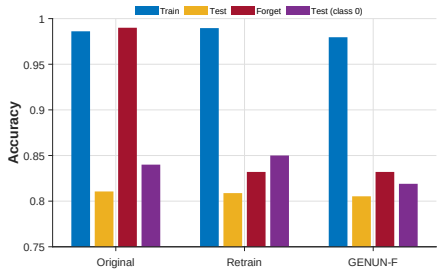


Figure 12: Performance comparison on single-class subset unlearning. The bars is the accuracy on different data subsets across Original, Retrain, and GENUN-F models. Test (Class 0) denotes generalization accuracy on all unseen test samples belonging to the target class (Class 0).

range regardless of the specific model state $\mathcal{F}(\cdot;\widehat{\mathbf{\Theta}}_k)$. Furthermore, whether the model is in early training stages or fully converged, the algorithm consistently achieves significant unlearning effects while preserving the performance on the D_r.

Performance of GENUN-F on single-class subset unlearning. To investigate whether GENUN-F risks distorting the feature space for retained samples similar to the forget set (i.e., over-correction), we designed a single-class subset unlearning task. In this setup, the forget set D_f is composed entirely of samples from a specific target class C_t, while the remaining samples of C_t and samples of other classes are retained. We selected Class 0 of the CIFAR-10 dataset as the target class C_t and constructed a forget set D_f consisting of 500 samples. The results are presented in Fig.12. As we can see, regarding unlearning efficacy, GENUN-F successfully reduces the accuracy on the forget set D_f from the original $\sim 99\%$ (indicating overfitting) to approximately 83%, which aligns closely with that of the Retrained model. Crucially, the accuracy on the held-out test samples of the target class remains robust, showing negligible degradation compared to the Retrain baseline. This stability confirms that no over-correction occurred during the GENUN-F unlearning process.

For class-level forgetting setting, we also conducted verification for GENUN and reported the results in Tab. 4. GENUN-M and GENUN-F achieve the best performance and outperforms other baselines.

Ablation study about GENUN-M. To validate the efficiency of GENUN-M, we use the GENUN-M's initilization as the intial checkpoint for other machine unlearning methods and perform random-subset unlearning on CIFAR-10, and report the results in Table 5. It shows that GENUN-M achieves the

Table 4: Class forgetting on CIFAR-10 for a randomly-selected class (1st one) and MIA result with RF as an attacker.

	Original	**Retrain**	**FT**	**GA**	**RL**	**EU-k**	**IU**	ℓ_1**-sparse**	**SalUn**	SCRUB	$\hat{\Theta}$	GENUN-M	GENUN-F
$\mathcal{D}_r$	98.89	98.63	98.50	88.92	51.05	99.28	53.97	96.48	97.54	99.96	79.50	95.81	97.06
$\mathcal{D}_{t_r}$	80.96	81.79	81.17	74.36	45.26	82.30	44.84	80.51	81.36	83.40	75.22	82.84	82.08
$\mathcal{D}_{t_f}$	77.40	0.00	0.00	12.50	61.80	0.00	0.00	0.00	0.02	1.22	68.70	**0.00**	**0.00**
$\mathcal{D}_f$	98.38	0.00	0.00	14.30	69.78	0.00	0.00	0.00	0.01	0.90	73.38	**0.00**	**0.00**
Time (s)		316.65	53.40	60.06	56.19	21.03	10.97	14.58	13.18	52.66	—	55.73	57.70
MIA(loss)	55.5	49.5	53.3	52.4	51.3	51.6	53.3	52.1	51.9	51.3	—	**49.7**	**50.2**
MIA(logits)	56.1	50.3	58.7	54.5	53.5	57.1	54.1	55.4	53.5	52.4	—	**49.2**	**51.7**

Table 5: Comparison of the different MU methods when using the GENUN-M's initilization as the intial checkpoint by performing random-subset unlearning on CIFAR-10.

	FT	**GA**	**RL**	**EU-k**	**IU**	ℓ_1**-sparse**	**SalUn**	SCRUB	GENUN-M
$\mathcal{D}_r$	97.31	56.76	68.88	88.29	32.58	96.47	91.85	94.04	98.40
$\mathcal{D}_t$	81.39	53.64	65.55	77.62	30.82	79.61	79.66	80.45	81.99
$\mathcal{D}_f$	82.72	53.14	68.44	82.86	32.46	80.82	84.54	83.22	82.70
|Diff|	1.33	0.50	2.89	5.24	1.64	1.21	4.88	2.77	**0.71**
Time (s)	200.36	5.94	3.46	90.86	11.37	198.05	21.01	100.12	33.41

Table 6: Performance comparison of GENUN-F by replacing the InfoNCE $\mathcal{L}_f^{conf}$ with other constrastive objectives over CIFAR-10.

GENUN-F	**+InfoNCE**	**+Triplet**	**+MMD**	**+N-pair**
$\mathcal{D}_r$	98.05	92.64	90.00	93.82
$\mathcal{D}_t$	78.42	76.90	75.30	77.98
$\mathcal{D}_f$	80.30	79.40	83.10	82.10
|Diff|	**1.88**	2.50	7.80	4.12
Time(s)	**124.63**	323.16	161.11	243.9

best performance considering the full-stack evaluation over other methods, even they start from the same initialization model coming from the middle-stage of the training process of the original model.

Ablation study about the InfoNCE component in GENUN-F. We replace the $\mathcal{L}_f^{conf}$ in Eq. 2 with other constrastive objectives, including Triplet, MMD, and N-pair. Table 6 illustrates the test result on CIFAR-10 dataset. Although other alternatives are applicable, InfoNCE loss show more efficient performance.