Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-26)
Special Track on AI4Tech: AI Enabling Critical Technologies

A Durable Machine Unlearning Framework to Nullify Recall of Sensitive Data on Incremental Training

Qingqing Cao[1], **Liang Hu**[2*], **Dora D. Liu**[2],
Jiaxing Miao[2], **Jian Cao**[1*], **Zhongyuan Lai**[4], **Wei Cao**[5]
[1]Shanghai Jiao Tong University
[2]Tongji University
[3]Macquarie University
[4]Ballsnow AI Technology
[5]Hefei University of Technology
cicely_2020@sjtu.edu.cn, lianghu@tongji.edu.cn
liudongmei_0506@163.com, jiaxingmiao@tongji.edu.cn
cao-jian@sjtu.edu.cn, abrikosoff@yahoo.com, weicao@hfut.edu.cn

Abstract

The advancement of data privacy regulations has spurred the development of Machine Unlearning (MU), which is designed to remove the influence of sensitive data from a trained model and results in an unlearned model (ULM). Despite rapid progress in MU techniques, their vulnerabilities remain underexplored, which poses risks due to potential leakage of unlearned information. In realistic scenarios, ULMs always need to be incrementally trained with newly collected data samples, which can lead to the consequences of recalling sensitive information if the new dataset contains similar or even the same unlearned samples. To address this issue, we devise a Durable Unlearning Enhancement (DUE) framework to avoid restoring unwanted sensitive information from incremental training data samples. The DUE framework has three key components that identify sensitive samples and suppress their gradients to update ULMs. Extensive experiments on state-of-the-art MU methods across multiple real-world datasets show that the proposed DUE framework can effectively nullify the recall of sensitive information after MU, and even improve the performance of ULMs. Consequently, our work establishes a new fundamental research direction in safe training against post-MU vulnerabilities.

1 Introduction

Data privacy is an immediate concern in our era of big data and large models. If training datasets contain sensitive information, the parameters of a trained model could inadvertently memorize individual data instances, which could be used by bad-faith actors to retrieve sensitive information from the model. In response to this, regulatory frameworks, such as GDPR [Hoofnagle *et al.*, 2019], explicitly empower users to request the removal of specific sensitive data. In this context, Machine Unlearning (MU) [Nguyen *et al.*, 2025] has been proposed as a principled paradigm that aims to eliminate the influence of a given forgetting dataset from a trained model (TRM) to produce an *unlearned model* (ULM).

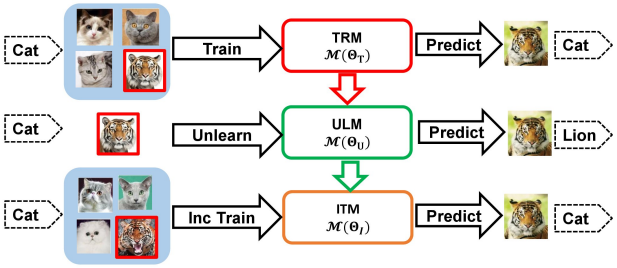

Figure 1: The demonstration of recall of forgetting class memberships that have been unlearned via MU. This study is critical to build a safe post-MU incremental training framework.

Rapid advances in MU research, while enhancing privacy, have simultaneously revealed potential security vulnerabilities—particularly the risk of privacy breaches through the recovery of erased information. This underscores a significant gap in understanding the potential threats faced by ULMs. Recently, relevant work [Hu *et al.*, 2024; Sui *et al.*, 2025] has emerged, which explicitly explores recall attacks against MU systems. Models always need to be incrementally tuned with newly collected data (see Figure 1), which may pose an unintentional risk of recalling sensitive information from the ULMs if the incremental training dataset contains similar or even the same unlearned samples. As a result, we have to rerun the MU procedure to remove sensitive information from the incrementally trained model (ITM) or even retrain the model from scratch with a refined dataset. Obviously, the additional rerun or retrain procedures cost significant computation time and resources, especially for large models.

The above challenge raises a fundamental question—"*Can*

*Corresponding author

we introduce a durable machine unlearning mechanism that is capable of nullifying the recall of sensitive information when incremental training?" It is well understood that a model can only internalize information from a sample if that sample contributes correct gradients to updating parameters during training. Conversely, if gradients associated with a subset of samples are systematically suppressed or reversed, then the trained model is fundamentally unable to retain the information specific to those samples. To this end, we devise a Durable Unlearning Enhancement (DUE) framework that enables a post-MU learning mechanism to avoid restoring undesired sensitive information from new datasets when performing incremental training. As a result, ITMs will be free of rerunning MU, and even improve their performance through the proposed framework. As most MU models restrict the scope of the investigation to the area of image classification tasks [Bourtoule *et al.*, 2021; Fan *et al.*, 2024; Jia *et al.*, 2023; Chen *et al.*, 2023; Chen *et al.*, 2024], we correspondingly study the DUE framework with these MU models in this area.

The DUE framework can work with current MU methods, which consists of three key components: (1) **Sensitive Sample Recognizer**, (2) **Sensitive Sample Gradient Suppressor** and (3) **Sensitive Sample Probability Reverser**. These components work together to identify sensitive samples and suppress the influence of their gradients on model updating. Current MU methods for image classification focuses on removing the effect at either the instance or class level. Instance-level MU methods deal with the request to unlearn information related to an individual or a group of instances [Cha *et al.*, 2024; Foster *et al.*, 2024], while class-level MU methods aim to eliminate the influence of an image class [Graves *et al.*, 2021]. As a result, we design a unified DUE framework for both instance-level and class-level post-MU training.

We summarize our main contributions as follows:

- To our knowledge, this is the *first attempt* study on how to avoid recalling sensitive information on incremental training, which can effectively avoid data privacy breaches and promote the robustness of the MU study.
- We propose DUE that is a post-MU training framework to avoid restoring unwanted sensitive information from new datasets when performing incremental training.
- We implement the DUE framework with three key components that can identify sensitive samples and suppress the influence of gradients to update the parameters of ULMs.
- We conduct extensive experiments on four real datasets and multiple state-of-the-art MU methods, showing that the proposed DUE framework can effectively nullify the recall of sensitive information after MU, and even improve the performance of ULMs.

2 Related Work

2.1 Machine Unlearning

The gold standard for exact unlearning is to retrain the model from scratch after removing specific data. Given the high cost of exact methods [Bourtoule *et al.*, 2021], research has shifted towards approximate unlearning, which accepts a modest sacrifice in the precision of forgetting for substantial gains in efficiency. Gradient Ascent (**GA**) [Graves *et al.*, 2021; Golatkar *et al.*, 2020; Thudi *et al.*, 2022; Miao *et al.*, 2024] finetunes the model by maximizing the loss on the forget set, pushing its parameters away from those that memorized the data. Random Labeling (**RL**) [Golatkar *et al.*, 2020] finetunes the model on the forget set using randomly assigned labels to disrupt and corrupt the learned associations. Influence-based Methods estimate the specific impact of the forget samples on the model's parameters. Unlearning is performed by subtracting this estimated influence, using tools like the Fisher Information Matrix (**FF**) [Becker and Liebig, 2022] or Influence Functions (**IU**) [Koh and Liang, 2017; Izzo *et al.*, 2021]. Techniques like ℓ_1-Sparse (**L1-SP**) [Jia *et al.*, 2023] integrate weight sparsity constraints during unlearning, promoting the removal of connections specifically tied to the forget data. Many MU methods degrade model performance, and lead to "over-unlearning". Some recent work has explored more precise unlearning on forget samples. Boundary Unlearning (**BU**) [Chen *et al.*, 2023] adjusts the model's decision boundary to mimic the behavior of a retrained-from-scratch model, aiming for precise forgetting with minimal collateral damage; **SalUn** [Fan *et al.*, 2024] employs a "weight saliency" metric to identify and modify only the most critical parameters for forgetting, narrowing the performance gap with exact unlearning; **UNSC** [Chen *et al.*, 2024] constrains the unlearning update to a null space defined by the remaining data, theoretically guaranteeing that the model's performance on the retain set is preserved throughout the forgetting process. Decoupled Distillation To Erase (**DELETE**) [Zhou *et al.*, 2025] applies a mask to separate forgetting logits from retention logits to optimize both the forgetting and refined retention components simultaneously.

2.2 Post-unlearning Vulnerabilities

Recent research has revealed significant security and privacy vulnerabilities that persist or emerge after the unlearning process. [Chen *et al.*, 2021] pioneered research on how machine unlearning jeopardizes privacy through membership inference attacks. [Hu *et al.*, 2024] introduced unlearning inversion attacks, extending traditional model inversion attacks to the unlearning context. Their methodology identifies unlearning data leakage in deep neural networks by exploiting the residual information retained in model parameters after data removal. [Zhang *et al.*, 2023] pioneered reconstruction attacks with conditional matching GANs, achieving high reconstruction accuracy by exploiting gradient residuals that persist after approximate unlearning. [Sui *et al.*, 2025] proposed a membership recall attack (MRA), a model-agnostic attack framework, to effectively recover class memberships of forgotten instances from ULMs after unlearning.

Different from the above study of various post-unlearning attacks on ULMs, in this paper we aim to study how to construct a safe post-MU training framework to avoid recalling sensitive information.

3 Preliminaries

We first introduce the datasets and models used in our study, followed by a formal definition of the problem.

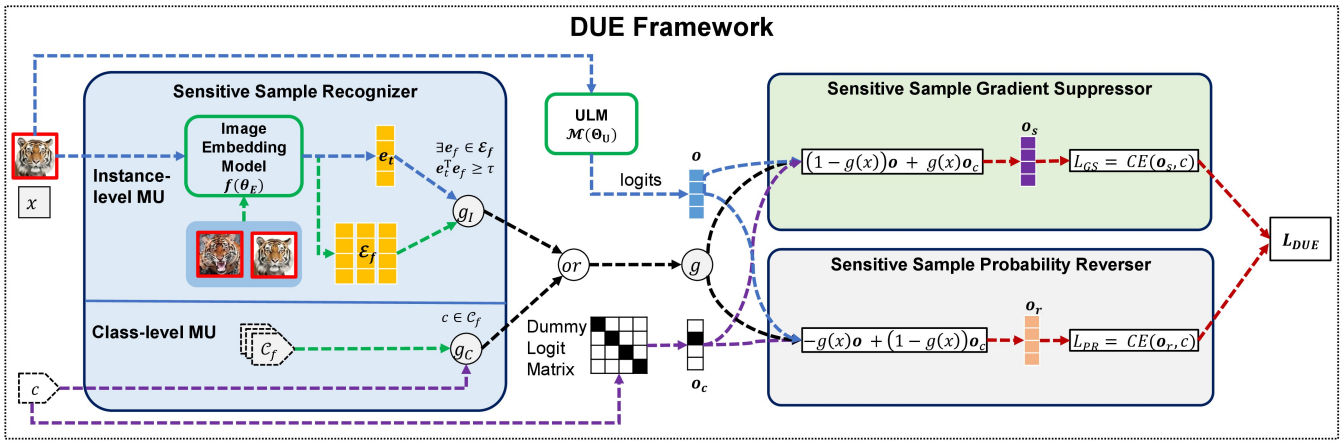

Figure 2: The architecture of DUE framework. It consists of three key components to nullify the recall of sensitive information: (1) **Sensitive Sample Recognizer (SSR)**, (2) **Sensitive Sample Gradient Suppressor (SSGS)** and (3) **Sensitive Sample Probability Reverser (SSPR)**.

3.1 Involved Datasets

Training dataset $\mathcal{D}_{tr} : \{\mathcal{X}_{tr}, \mathcal{Y}_{tr}\}$ denotes the data used to initially train models, where $\mathcal{X}_{tr}$ denotes the image set and $\mathcal{Y}_{tr}$ denotes the corresponding label set.
Forgetting dataset $\mathcal{D}_f : \{\mathcal{X}_f, \mathcal{Y}_f\}$ is a subset of $\mathcal{D}_{tr}$, that is, $\mathcal{D}_f \subset \mathcal{D}_{tr}$. In MU, $\mathcal{D}_f$ is a set of sensitive samples that should be unlearned, that is, the ULM cannot tell the true labels when $\mathcal{X}_f$ is input.
Incremental dataset $\mathcal{D}_i : \{\mathcal{X}_i, \mathcal{Y}_i\}$ denotes the new data for incremental training, where $\mathcal{X}_i$ contains very similar or even the same images in the forgetting set $\mathcal{X}_f$.
Test dataset $\mathcal{D}_{ts} : \{\mathcal{X}_{ts}, \mathcal{Y}_{ts}\}$ is the unseen dataset for test.

3.2 Involved Models

Trained Model (TRM) $\mathcal{M}(\Theta_T)$ is the model that has been trained on the training dataset $\mathcal{D}_{tr}$.
Unlearned Model (ULM) $\mathcal{M}(\Theta_U)$ is the model that has unlearned the forgetting dataset $\mathcal{D}_f$ based on $\mathcal{M}(\Theta_T)$.
Incrementally Trained Model (ITM) $\mathcal{M}(\Theta_I)$ is the model incrementally trained on dataset $\mathcal{D}_i$ based on $\mathcal{M}(\Theta_U)$.
DUE Model (DUE) $\mathcal{M}(\Theta_{DUE})$ is the model incrementally trained on the dataset $\mathcal{D}_i$ under the DUE framework.

3.3 Problem Formulation

When the ULM $\mathcal{M}(\Theta_U)$ is incrementally trained on $\mathcal{D}_i$, the ITM $\mathcal{M}(\Theta_I)$ tends to recall sensitive information due to the sensitive data subset $\mathcal{D}_s \subset \mathcal{D}_i$. To avoid such data privacy breaches, we aim to implement the DUE that is a post-MU training framework to identify sensitive samples of $\mathcal{D}_s$ and re-encode their gradients when updating $\mathcal{M}(\Theta_U)$. As a result, it will lead to a safe ITM $\mathcal{M}(\bar{\Theta}_I)$ in which no information from $\mathcal{D}_s$ is retained. Moreover, we discuss the unified implementation of DUE framework on both class-level and instance-level post-MU training.

4 Proposed Method

4.1 Overview

First, We obtain different ULMs $\{\mathcal{M}(\Theta_U)\}$ in terms of running various MU methods on a TRM $\mathcal{M}(\Theta_T)$. Figure 2 shows the architecture of proposed DUE framework, which consists of three key components: (1) **Sensitive Sample Recognizer (SSR)**, (2) **Sensitive Sample Gradient Suppressor (SSGS)** and (3) **Sensitive Sample Probability Reverser (SSPR)**.

4.2 Sensitive Sample Recognizer

To avoid ULMs $\{\mathcal{M}(\Theta_U)\}$ restoring sensitive information, it is necessary to recognize sensitive samples when performing incremental training. Given a sample $(x_i, y_i) \in \mathcal{D}_i$, we check if it is a sensitive sample, i.e., $(x_i, y_i) \in \mathcal{D}_s$. Here, we consider the ULMs having been removed sensitive information through class-level MU and instance-level MU, respectively.

Class-level Recognition. For class-level MU, it has eliminated the influence of a forgetting class from a TRM $\{\mathcal{M}(\Theta_T)\}$. Therefore, we can simply check if a sensitive training sample (x_i, y_i) whose label y_i is in the forgetting class set $\mathcal{C}_f$.

$$g_C(x_i) = \begin{cases} 1 & y_i \in \mathcal{C}_f \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Instance-level Recognition. For instance-level MU, all the samples in the forgetting set, i.e., $(x, y) \in \mathcal{D}_f$, should be recognized as sensitive data. Furthermore, given a sensitive sample x, we argue that similar images that are transformed (e.g., resize, flip, crop) from x should also be considered as sensitive samples. As a result, we take images $\mathcal{X}_f$ of $\mathcal{D}_f$ as prototype instances, and propose a prototype-based contrastive learning algorithm to train the sensitive sample recognizer.

We use an image embedding model $f_{\theta_E}(\cdot)$, which can be obtained by a copy of TRM $\mathcal{M}(\Theta_T)$ or a pretrained backbone (e.g. ResNet, ViT), to extract an embedding vector. Given a prototype image $x_p \in \mathcal{X}_f$, a transformed image $x_q \sim x_p$, and a randomly selected image $x_n \in \mathcal{X}_{tr}$, we easily obtain their normalized embeddings.

$$\langle \mathbf{e}_p, \mathbf{e}_q, \mathbf{e}_n \rangle = normalize(f_{\theta_E}(\langle x_p, x_q, x_n \rangle)) \quad (2)$$

Then, we set up the following loss:

$$\ell = \max(\mathbf{e}_p^\top \mathbf{e}_n - \mathbf{e}_p^\top \mathbf{e}_q + \alpha, 0) + \gamma \max(\beta - \mathbf{e}_p^\top \mathbf{e}_q, 0) \quad (3)$$

where the first term is a triplet loss to optimize cosine similarity between $\mathbf{x}_p$ and $\mathbf{x}_n$ and that between $\mathbf{x}_p$ and $\mathbf{x}_q$ with a margin α in the embedding space; the second term aims to regularize cosine similarity between $\mathbf{e}_p$ and $\mathbf{e}_q$ at least β. When $f_{\theta_E}(\cdot)$ is trained, we can obtain the normalized embedding set for all forgetting images, i.e. prototypes, $\mathcal{E}_f = normalize(f_{\theta_E}(\mathcal{X}_f))$. Then, we can check if a training sample x_t contains sensitive information.

$$g_I(x_t) = \begin{cases} 1 & \exists \mathbf{e}_f \in \mathcal{E}_f,\ \mathbf{e}_t^\top \mathbf{e}_f \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

4.3 Sensitive Sample Gradient Suppressor

In the image classification problem, gradient ∇L is derived from the cross entropy loss (CE).

$$\mathbf{o} = \mathcal{M}(x|\Theta_U) \tag{5}$$
$$L = CE(\mathbf{o}, c) = -\log\left(softmax(\mathbf{o})[c]\right) \tag{6}$$

where $\mathbf{o}$ denotes logits output and c is the target label.

For any sensitive sample x, i.e., $g_C(x) = 1$ or $g_I(x) = 1$, we need to suppress its gradient to update model parameters. In particular, we create a dummy logit matrix $\mathbf{O} = a\mathbf{I}_N$ where $\mathbf{I}_N$ is an $N \times N$ identity matrix and a is a constant to scale the diagonal ($a = 32$ is used in this paper). Given a sensitive sample (x, c), we can immediately obtain the dummy logit vector $\mathbf{o}_c = \mathbf{O}[c, :]$. That is, the probabilities $\mathbf{p} = softmax(\mathbf{o}_c)$ tend to output an approximate one-hot vector, i.e., the probability on c is close to 1. Then, we have the logits selected by $g(x) = g_C(x) \vee g_I(x)$.

$$\mathbf{o}_s = (1 - g(x))\mathbf{o} + g(x)\mathbf{o}_c \tag{7}$$
$$L_{GS}(x, c) = CE(\mathbf{o}_s, c) \tag{8}$$

Accordingly, we obtain the adaptive CE loss (Eq (8)) which tends to be zero when $g(x) = 1$, i.e., $\nabla L_{GS}(x, c) = \mathbf{0}$. As a result, the gradients associated with sensitive samples are successfully suppressed when updating ULM $\mathcal{M}(\Theta_U)$.

4.4 Sensitive Sample Probability Reverser

In our experiments, we find that the ULMs can still recall sensitive information after incremental training, even when the above gradient suppressor is applied. This is because the information in forgetting samples could be shared in multiple classes (e.g., bird and airplane might share the background information of sky). As a result, the updating of ULMs on training samples that have features correlated with forgetting samples may indirectly recall the unlearned sensitive information.

In this paper, we additionally design a simple but effective probability reverser to enhance the MU result. Given the logits $\mathbf{o} = \mathcal{M}(x|\Theta_U)$ of sample (x, c), we compute the probabilities in terms of softmin, i.e.,

$$\mathbf{p} = softmin(\mathbf{o}) = softmax(-\mathbf{o}) \tag{9}$$

That is, the smallest logit will produce the largest probability. As a result, the CE loss $L_{GR} = CE(-\mathbf{o}, c)$ aims to optimize the parameter to produce the smallest predicted probability on the true-label class. Finally, we have the following adaptive CE loss:

$$\mathbf{o}_r = -g(x)\mathbf{o} + (1 - g(x))\mathbf{o}_c \tag{10}$$
$$L_{PR}(x, c) = CE(\mathbf{o}_r, c) \tag{11}$$

The above CE loss results in the gradient $\nabla L_{GR}(x, c)$ to update ULM $\mathcal{M}(\Theta_U)$ towards the reverse direction to produce the highest probability on true label for sensitive samples, i.e., $g(x) = 1$ while dummy logit vector $\mathbf{o}_c = \mathbf{O}[c, :]$ is applied to generate zero loss, i.e., $g(x) = 0$.

4.5 Recall Nullified Incremental Training

With the above three components, we can conduct incremental training based on ULM $\mathcal{M}(\Theta_U)$. Given the incremental dataset $\mathcal{D}_i$, the loss for mini-batch learning is:

$$L_{DUE} = \frac{\sum_{(x,y)\in\mathcal{B}} L_{GS}(x, y)}{\sum_{(x,y)\in\mathcal{B}} (1 - g(x))} + \lambda \frac{\sum_{(x,y)\in\mathcal{B}} L_{PR}(x, y)}{\sum_{(x,y)\in\mathcal{B}} g(x)}$$

where $\mathcal{B} \subset \mathcal{D}_i$ denotes a mini-batch training data. The first term is the average loss to update ULMs with a batch of new samples where the gradient associated with sensitive samples has been suppressed (cf Eq (8)), while the second term is the average loss to enhance MU results in terms of probability reverse (Eq (11)). After incremental training with the proposed framework, we obtain the DUE model $\mathcal{M}(\Theta_{DUE})$.

5 Experiments

5.1 Experiment Setup

Data Preparation

In our experiments, four real datasets are used to provide a comprehensive evaluation. CIFAR-10 [Krizhevsky *et al.*, 2009] and STL-10 [Coates *et al.*, 2011] are low-resolution image datasets with 10 classes, which are used for class-level post-MU evaluation. Oxford-IIIT Pet (Pet-37) [Parkhi *et al.*, 2012] and Oxford 102 Flower (Flower-102) [Nilsback and Zisserman, 2008] are high-resolution image datasets, which are used for instance-level post-MU evaluation.

As shown in Table 1, we use the official splits of the training dataset $\mathcal{D}_{tr}$ for initial training, and we further split the official test dataset into incremental training dataset $\mathcal{D}_i$ and our test dataset $\mathcal{D}_{ts}$ for evaluation. Moreover, we select two classes in CIFAR-10 and STL-10 as the forgetting classes, and randomly sampling 10% of data from $\mathcal{D}_{tr}$ in CIFAR-10's forgetting classes while 20% of data from $\mathcal{D}_{tr}$ in STL-10's forgetting classes to construct the class-level forgetting datasets $\mathcal{D}_f$. Furthermore, we randomly sample individual forgetting instances from five classes in Pet-37 and Flower-102 as the instance-level forgetting datasets $\mathcal{D}_f$. Then, we apply a set of transforms (including random flip, crop, and

Dataset	# Classes	$\lvert\mathcal{D}_{tr}\rvert$	$\lvert\mathcal{D}_{ts}\rvert$	$\lvert\mathcal{D}_f\rvert$	$\lvert\mathcal{D}_i\rvert$
CIFAR-10	10	50,000	10,000	1,000	3,500
STL-10	10	8,000	5,000	320	1,570
Pet-37	37	3,680	3,669	100	278
Flower-102	102	6,149	1,020	60	1,014

Table 1: Statistical summary of the datasets and their splits

Dataset		TRM		GA	FF	RL	IU	BU	L1-SP	SalUn	DELETE	UNSC
CIFAR-10	$\mathcal{D}_f \downarrow$	1.0000	ULM	0.2460	0.5280	0.3010	0.5140	0.4980	0.2250	0.3970	0.2140	0.1020
			ITM	0.9960	0.9710	0.9980	0.9960	0.9910	0.9980	0.9980	0.9970	0.9970
			DUE	**0.2330**	**0.0680**	**0.0000**	**0.3890**	**0.2930**	**0.0510**	**0.3440**	**0.1560**	**0.0300**
			ΔAcc	-0.0130	-0.4600	-0.3010	-0.1250	-0.2050	-0.1740	-0.0530	-0.0580	-0.0720
	$\mathcal{D}_{ts}^f \downarrow$	0.8600	ULM	0.2005	0.4780	0.1820	0.4085	0.3690	0.1945	**0.2535**	0.1930	0.0635
			ITM	0.9020	0.9170	0.9110	0.9040	0.9055	0.9110	0.9115	0.9155	0.9170
			DUE	**0.1650**	**0.0455**	**0.0000**	**0.2695**	**0.1920**	**0.0320**	0.2590	**0.1110**	**0.0115**
			ΔAcc	-0.0355	-0.4325	-0.1820	-0.1390	-0.1770	-0.1625	0.0055	-0.0820	-0.0520
	$\mathcal{D}_{ts}^r \uparrow$	0.8331	ULM	0.8085	0.5884	0.8418	0.7973	0.7473	0.8124	0.7991	0.7785	0.6928
			ITM	0.8625	0.8376	0.8604	0.8653	0.8616	0.8595	0.8601	0.8606	0.8645
			DUE	**0.8804**	**0.8708**	**0.8804**	**0.8793**	**0.8821**	**0.8796**	**0.8855**	**0.8806**	**0.8804**
			ΔAcc	0.0719	0.2824	0.0386	0.0820	0.1349	0.0673	0.0864	0.1021	0.1876
STL-10	$\mathcal{D}_f \downarrow$	1.0000	ULM	0.4344	0.7594	0.2375	0.4969	0.4469	0.3531	0.4781	0.4125	/
			ITM	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	/
			DUE	**0.3063**	**0.4688**	**0.0625**	**0.4563**	**0.0531**	**0.1813**	**0.0688**	**0.2844**	/
			ΔAcc	-0.1281	-0.2906	-0.1750	-0.0406	-0.3938	-0.1719	-0.4094	-0.1281	/
	$\mathcal{D}_{ts}^f \downarrow$	0.9250	ULM	0.3150	0.7190	0.1500	0.3390	0.3580	0.2590	0.3430	0.3310	/
			ITM	0.9520	0.9590	0.9460	0.9500	0.9490	0.9500	0.9490	0.9490	/
			DUE	**0.1790**	**0.3190**	**0.0450**	**0.2600**	**0.0220**	**0.1040**	**0.0480**	**0.1610**	/
			ΔAcc	-0.1360	-0.4000	-0.1050	-0.0790	-0.3360	-0.1550	-0.2950	-0.1700	/
	$\mathcal{D}_{ts}^r \uparrow$	0.9163	ULM	0.9100	0.1588	0.9160	0.8363	0.8335	0.9078	0.9135	0.9205	/
			ITM	0.9425	0.9293	0.9388	0.9425	0.9390	0.9410	0.9443	0.9445	/
			DUE	**0.9435**	**0.9373**	**0.9440**	**0.9423**	**0.9435**	**0.9355**	**0.9363**	**0.9433**	/
			ΔAcc	0.0335	0.7785	0.0280	0.1060	0.1100	0.0278	0.0228	0.0228	/

Table 2: Performance comparison under various SOTA MU methods, where ULM indicates the Acc after MU while ITM indicates the Acc after incremental training, DUE indicates the Acc using the proposed DUE framework for class-level post-MU training, and ΔAcc shows the improvement of DUE based on ULM, i.e., $\Delta Acc = Acc_{DUE} - Acc_{ULM}$.

mixup [Zhang *et al.*, 2018]) based on $\mathcal{D}_f$ to construct a transformed image set as a part of incremental training dataset $\mathcal{D}_i$ to simulate the cases of similar image recognition for instance-level DUE as described in Section 4.2.

Model Configuration

The DUE framework is model-agnostic, so we use EfficientNet (EFN) [Tan and Le, 2019] on CIFAR datasets, ResNet [He *et al.*, 2016] on Pet-37, and Swin-Transformer (Swin-T) [Liu *et al.*, 2022] on Flower-102 for a comprehensive study. We use SGD optimizer for MU methods with a momentum of 0.9, a weight decay of 0.005, and AdamW for our DUE framework. Other more detailed settings can be found in the appendix. For each comparison model, we carefully tuned its hyperparameters to achieve optimal performance.

Involved MU Methods

In the experiments, a set of SOTA MU methods, including **GA** [Graves *et al.*, 2021], **RL** [Golatkar *et al.*, 2020], **FF** [Becker and Liebig, 2022], **IU** [Koh and Liang, 2017], **BU** [Chen *et al.*, 2023], **L1-SP** [Jia *et al.*, 2023], **UNSC** [Chen *et al.*, 2024], **SalUn** [Fan *et al.*, 2024] and **DELETE** [Zhou *et al.*, 2025], are involved to comprehensively evaluate the capability of recall nullification by the proposed DUE framework. In particular, **UNSC** does not support the Swin-T architecture, so the evaluation results are on CIFAR-10 only.

5.2 DUE for Class-level Post-MU Training

Firstly, we train an image classification model with $\mathcal{D}_{tr}$ and obtain the TRM $\mathcal{M}(\Theta_T)$. Secondly, various MU methods are used to unlearn $\mathcal{D}_f$ based on $\mathcal{M}(\Theta_T)$ and obtain the corresponding ULMs $\mathcal{M}(\Theta_U)$. Then, we train each $\mathcal{M}(\Theta_U)$ with $\mathcal{D}_i$ using or not using the DUE framework for comparison, which leads to the results of DUE $\mathcal{M}(\Theta_{DUE})$ and ITM $\mathcal{M}(\Theta_I)$ shown in Table 2.

DUE Performance over Different MU Methods

Table 2 show the accuracy (Acc) on $\mathcal{D}_f$, i.e., forgetting set, $\mathcal{D}_{ts}^f$, i.e., test set with forgetting classes only, and $\mathcal{D}_{ts}^r$ i.e., test set without forgetting classes, respectively. And ΔAcc specify the improvement of DUE based on ULM, i.e., $\Delta Acc = Acc_{DUE} - Acc_{ULM}$

We compared the Acc of ULM and DUE on CIFAR-10 and STL-10 datasets with diverse MU models. From this table, all of the Acc on $\mathcal{D}_f$ of TRM are 1.000 after training, while they drop to low Acc after MU (cf. the row of ULM). This shows that the MU methods can successfully mitigate the influence of $\mathcal{D}_f$ from TRM. It is easily found that the Acc of ITMs on both $\mathcal{D}_f$ and $\mathcal{D}_i^f$ are high, which demonstrates that almost all forgetting information is recalled after incremental training. In comparison, the proposed DUE framework con-

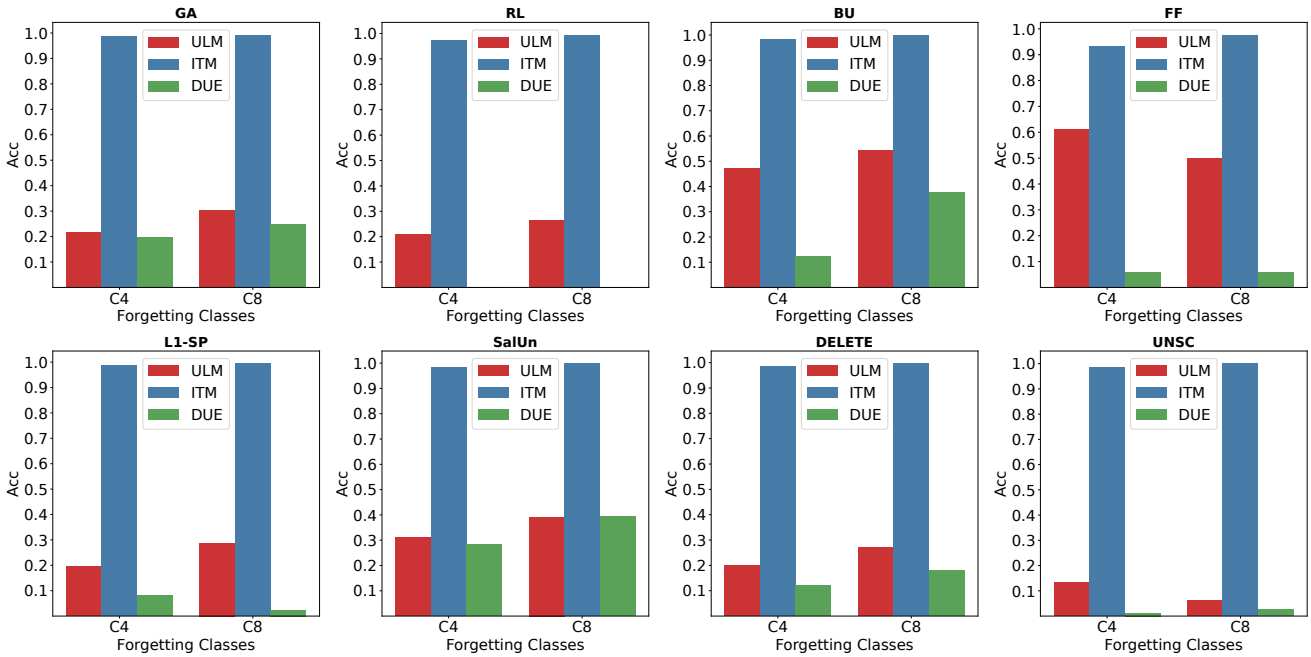

Figure 3: Comparison of the Acc between ULM, ITM and DUE w.r.t. each forgetting class on CIFAR-10 dataset.

Dataset		TRM		GA	FF	RL	IU	BU	L1-SP	SalUn	DELETE
Flower-102	$\mathcal{D}_f \downarrow$	1.0000	ULM	0.0533	0.6533	0.1867	0.2800	0.2000	0.2933	0.1467	0.2133
			ITM	1.0000	1.0000	1.0000	0.9867	1.0000	1.0000	1.0000	1.0000
			DUE	**0.0000**	**0.0267**	**0.0000**	**0.2400**	**0.0133**	**0.0000**	**0.0267**	**0.0000**
			ΔAcc	-0.0533	-0.6266	-0.1867	-0.0400	-0.1867	-0.2933	-0.1200	-0.2133
	$\mathcal{D}_i^f \downarrow$	1.0000	ULM	0.0540	0.5676	0.1622	0.2162	0.1081	0.2162	0.0541	0.1892
			ITM	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
			DUE	**0.0000**	**0.1081**	**0.1081**	**0.2703**	**0.1081**	**0.1081**	**0.1081**	**0.0000**
			ΔAcc	-0.0540	-0.4595	-0.0541	0.0541	0.0000	-0.1081	0.0540	-0.1892
Pet-37	$\mathcal{D}_f \downarrow$	1.0000	ULM	**0.0100**	0.3000	0.1400	0.3700	0.0700	0.0000	0.1800	0.3000
			ITM	0.9900	1.0000	0.9800	1.0000	1.0000	1.0000	1.0000	1.0000
			DUE	0.0700	**0.0000**	**0.0200**	**0.0700**	**0.0500**	**0.0000**	**0.0100**	**0.0800**
			ΔAcc	0.0600	-0.3000	-0.1200	-0.3000	-0.0200	0.0000	-0.1700	-0.2200
	$\mathcal{D}_i^f \downarrow$	0.8200	ULM	**0.0000**	0.1200	0.1200	**0.0000**	**0.0000**	0.0000	0.0200	**0.0200**
			ITM	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
			DUE	0.2200	**0.0400**	**0.1000**	0.0800	0.0200	**0.0000**	**0.0000**	0.1200
			ΔAcc	0.2200	-0.0800	-0.0200	0.0800	0.0200	0.0000	-0.0200	0.1000

Table 3: Performance comparison under various SOTA MU methods, where ULM indicates the Acc after MU while ITM indicates the Acc after incremental training, DUE indicates the Acc using the proposed DUE framework for instance-level post-MU training, and ΔAcc shows the improvement of DUE based on ULM, i.e., $\Delta Acc = Acc_{DUE} - Acc_{ULM}$.

sistently achieves performance comparable to that of ULMs after incremental training. Furthermore, the Acc of DUE on both $\mathcal{D}_f$ and $\mathcal{D}_i^f$ are even lower than that of ULMs, which proves that DUE can enhance the MU results by successfully suppressing and reversing the gradient of sensitive samples to update ULMs during incremental training. More specifically, the Acc of ULMs on $\mathcal{D}_f$ with **FF**, **IU** and **BU** are relatively high (above 0.45), which indicates that these MU methods can only unlearn the forgetting samples in part. Obviously, DUE effectively refines the MU results in terms of the Acc, i.e., the significantly negative of ΔAcc indicates the improvement of forgetting.

Furthermore, based on all the Acc between ITM and DUE on the test datasets $\mathcal{D}_t^r s$, we find that DUE improves the prediction accuracy on both CIFAR-10 and STL-10 for all MU models. This proves that the proposed DUE is an effective

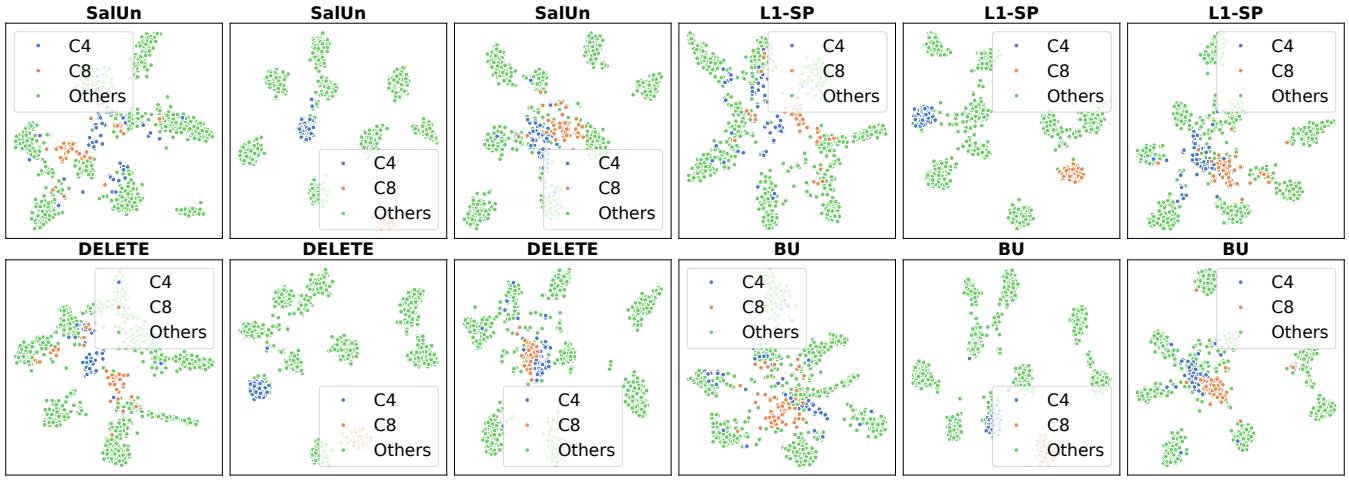

Figure 4: Comparison of t-SNE evaluated on CIFAR-10 w.r.t. ULMs (left columns), ITMs (middle columns), and DUEs (right columns)

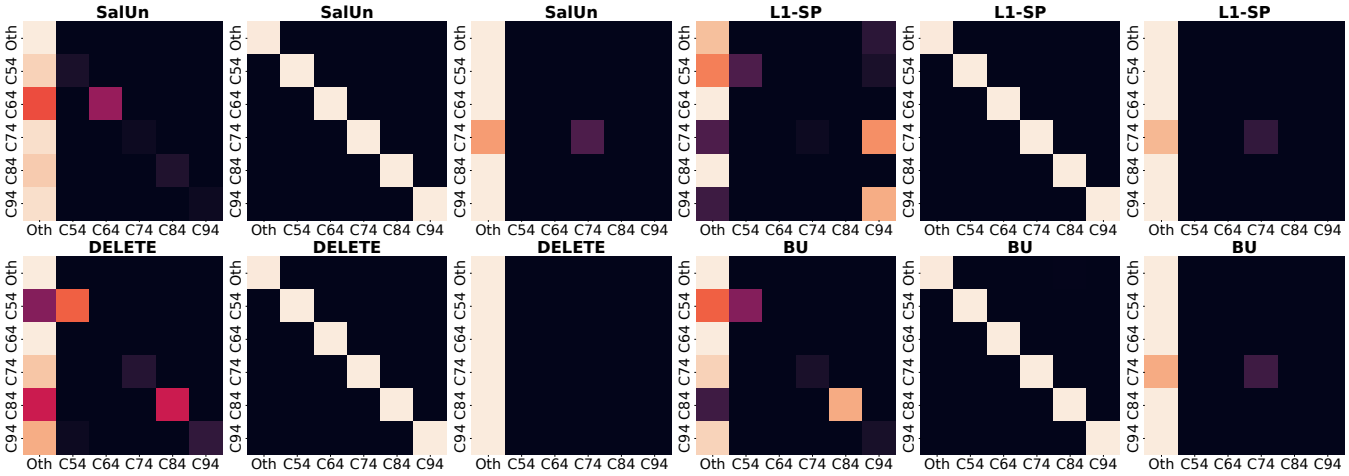

Figure 5: Comparison of normalized confusion matrices evaluated on Flower-102 w.r.t. ULMs (the left columns), ITMs (the middle columns), and DUEs (the right columns). Diagonal elements represent correct classifications, while off-diagonal elements indicate mislabeling.

and versatile model-agnostic framework that can not only nullify the recall of sensitive data but also further improve prediction performance through incremental training.

Demonstration of Class-specific Efficacy

To intuitively demonstrate the capacity of DUE to nullify recall of the forgetting samples $\mathcal{X}_f$, we further conducted detailed evaluations with respect to each forgetting class. Figure 3 demonstrate the comparison of the *Acc* of among ULM, ITM and DUE on CIFAR-10 for each forgetting class for six MU methods . By checking the improvement of *Acc* for each class, we can observe similar results as shown in Table 2, that is, ULMs can partially remove the influence of forgetting class samples but they are easily recalled by incremental training while DUE can not only nullify the recall but further enhance the forgetting effects.

Figure 4 compares the t-SNE results among ULM (left), ITM (middle) and DUE (right) on the joint image dataset $\mathcal{X}_p = \mathcal{X}_f \bigcup \mathcal{X}_i^f \bigcup \mathcal{X}_{ts}^r$. We observe that significant overlaps between both forgetting-class and non-forgetting-class instances. As a result, although ULMs partially remove the class information from forgetting samples, the inference capacity on non-forgetting-class instances is reduced due to the side effects of MU. In comparison, the boundaries between all classes are much clearer in terms of ITMs, which demonstrates the recall of forgetting-class instances after incremental training. By applying the DUE framework, we can find the boundaries between non-forgetting classes are relatively clear but there are apparent overlaps between forgetting classes.

5.3 DUE for Instance-level Post-MU Training

Analogously to the case of class-level post-MU training, we can easily obtain ULMs, ITMs, and DUEs on instance-level training datasets, Pet-37 and Flower-102.

DUE Performance over Different MU Methods

For the instance-level post-MU training case, we need to precisely unlearn the individual forgetting instances instead of the whole class. Table 3 compares the Acc on the forgetting datasets $\mathcal{D}_f$ and $\mathcal{D}_r^f$ for various MU models. Similar results

Component		BU		L1-SP		SalUn		DELETE	
SSGS	SSPR	$\mathcal{D}_{ts}^r \uparrow$	$\mathcal{D}_f \downarrow$	$\mathcal{D}_t^r s \uparrow$	$\mathcal{D}_f \downarrow$	$\mathcal{D}_t^r s \uparrow$	$\mathcal{D}_f \downarrow$	$\mathcal{D}_t^r s \uparrow$	$\mathcal{D}_f \downarrow$
	✓	0.791	0.859	0.766	0.591	0.763	0.822	0.760	0.906
✓		0.880	0.501	**0.880**	0.529	0.883	0.621	0.874	0.500
✓	✓	**0.882**	**0.293**	0.880	**0.051**	**0.886**	**0.344**	**0.881**	**0.156**

Table 4: Ablation results (*Acc*) of DUE on CIFAR-10 dataset.

Component			RL		L1-SP		SalUn		DELETE	
SSR	SSGS	SSPR	$\mathcal{D}_i^f \downarrow$	$\mathcal{D}_f \downarrow$	$\mathcal{D}_i^f \downarrow$	$\mathcal{D}_f \downarrow$	$\mathcal{D}_i^f \downarrow$	$\mathcal{D}_f \downarrow$	$\mathcal{D}_i^f \downarrow$	$\mathcal{D}_f \downarrow$
	✓	✓	0.500	0.190	0.160	0.170	0.600	0.510	0.860	0.200
✓		✓	0.440	0.320	0.480	0.580	0.220	0.340	0.620	0.800
✓	✓		0.480	0.520	0.260	0.390	0.240	0.820	0.260	0.850
✓	✓	✓	**0.100**	**0.020**	**0.000**	**0.000**	**0.000**	**0.010**	**0.120**	**0.080**

Table 5: Ablation results (Acc) of DUE on Pet-37 dataset.

can be observed compared to the class-level results shown in Table 2. We find that almost all forgetting samples are recalled, i.e., the Acc is close to 1 through incremental training. After applying the DUE framework, the Acc for each MU method drops to small values, which again proves the efficacy of DUE to enhance MU results by as obtained from incremental training.

Visualization of Confusion Matrices

Figure 5 visualizes the normalized confusion matrices on $\mathcal{D}_p = \mathcal{D}_f \bigcup \mathcal{D}_i^f \bigcup \mathcal{D}_{ts}^r$ w.r.t. ULMs (left), ITMs (middle), and DUEs (right). We can observe the highlights in the first column of each confusion matrix for ULM, which illustrates that MU methods have successfully unlearned the class information of forgetting instances. For ITMs, the diagonals of confusion matrices become clearly noticeable, that is, the class membership of forgetting samples has been successfully recalled. In comparison, the confusion matrices of DUE perform the best to eliminate the sensitive information of forgetting samples.

5.4 Ablation Study

In this section, we discuss the effectiveness of three key components of the DUE framework as presented in Section 4.1.

SSR: The component is used to recognize possible sensitive samples similar to prototype instances.

SSGS: The component is used to suppress the gradient of the samples recognized by SSR.

SSPR: The component is used to reverse the probability of the samples recognized by SSR.

Ablation for Class-level DUE

Table 4 reports the results of DUE on $\mathcal{D}_{ts}^r$ and $\mathcal{D}_f$. From the results, we can easily find that only using either **SSGS** or **SSPR** can still partially nullify the recall of sensitive samples. However, as analyzed in Section 4.4, ULMs can probably recall sensitive information if only using **SSGS** due to the common features between forgetting and non-forgetting samples. As a result, jointly using both **SSGS+SSPR** achieves the best performance as shown in the last row.

Ablation for Instance-level DUE

Table 5 reports the results of DUE on forgetting sets $\mathcal{D}_{ts}^f$ and $\mathcal{D}_f$. Here, the ablation of **SSR** component specifies whether to apply the similarity matching based on prototypes or simply perform to exact matching the forgetting samples. Comparing the first case with the last one, we easily find that using **SSR** can effectively find similar training samples to avoid unlearned information is implicitly recalled. Therefore, using all components **SSR+SSGS+SSPR** is the most effective way to nullify the recall of sensitive information.

6 Conclusion

This paper initiates a study on how to avoid recalling sensitive information from the training dataset containing unlearned samples when post-MU incremental training is conducted. As a result, the DUE framework is proposed to identify sensitive samples and suppress the influence of gradients to update the parameters of ULMs when incremental training. Extensive experiments on four real datasets and multiple state-of-the-art MU methods show that the proposed DUE framework can effectively nullify the recall of sensitive information and further enhance MU results. Consequently, this work establishes a new fundamental research direction in safe post-MU training to avoid data privacy breaches and promote the robustness of MU.

Acknowledgments

This work is partially supported by National Key R&D Program of China (2025CSJGG0039 / YDZX20263100004012) and National Natural Science Foundation of China (62276190).

Contribution Statement

Qingqing Cao and Liang Hu contributed equally to this work.

References

[Becker and Liebig, 2022] Alexander Becker and Thomas Liebig. Evaluating machine unlearning via epistemic uncertainty. *ArXiv preprint arXiv:2208.10836*, 2022.

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

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

[Chen *et al.*, 2021] Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When machine unlearning jeopardizes privacy. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, CCS '21, page 896–911, New York, NY, USA, 2021. Association for Computing Machinery.

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

[Chen *et al.*, 2024] Huiqiang Chen, Tianqing Zhu, Xin Yu, and Wanlei Zhou. Machine unlearning via null space calibration. In Kate Larson, editor, *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 358–366. International Joint Conferences on Artificial Intelligence Organization, 8 2024. Main Track.

[Coates *et al.*, 2011] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 215–223, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.

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

[Foster *et al.*, 2024] Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(11):12043–12051, Mar. 2024.

[Golatkar *et al.*, 2020] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312, 2020.

[Graves *et al.*, 2021] Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *AAAI Conference on Artificial Intelligence*, volume 35, pages 11516–11524, 2021.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[Hoofnagle *et al.*, 2019] Chris Jay Hoofnagle, Bart van der Sloot, and Frederik J. Zuiderveen Borgesius. The european union general data protection regulation: what it is and what it means*. *Information & Communications Technology Law*, 28:65 – 98, 2019.

[Hu *et al.*, 2024] Hongsheng Hu, Shuo Wang, Tian Dong, and Minhui Xue. Learn what you want to unlearn: Unlearning inversion attacks against machine unlearning. In *IEEE Symposium on Security and Privacy*, 2024.

[Izzo *et al.*, 2021] 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*, pages 2008–2016. PMLR, 2021.

[Jia *et al.*, 2023] Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. In *Neural Information Processing Systems*, 2023.

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

[Krizhevsky *et al.*, 2009] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Master's Thesis*, 2009.

[Liu *et al.*, 2022] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11999–12009, 2022.

[Miao *et al.*, 2024] Jiaxing Miao, Liang Hu, Qi Zhang, and Longbing Cao. Graph memory learning: Imitating lifelong remembering and forgetting of brain networks. *CoRR*, abs/2407.19183, 2024.

[Nguyen *et al.*, 2025] Thanh Tam Nguyen, Thanh Trung Huynh, Zhao Ren, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A survey of machine unlearning. *ACM Trans. Intell. Syst. Technol.*, 16(5):1–46, 2025.

[Nilsback and Zisserman, 2008] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008.

[Parkhi *et al.*, 2012] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, 2012.

[Sui *et al.*, 2025] Zhihao Sui, Liang Hu, Jian Cao, Dora D. Liu, Usman Naseem, Zhongyuan Lai, and Qi Zhang. Recalling the forgotten class memberships: Unlearned models can be noisy labelers to leak privacy. In James Kwok, editor, *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 6209–6218. International Joint Conferences on Artificial Intelligence Organization, 8 2025. Main Track.

[Tan and Le, 2019] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[Thudi *et al.*, 2022] Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling sgd: Understanding factors influencing machine unlearning. In *IEEE European Symposium on Security and Privacy*, pages 303–319, 2022.

[Zhang *et al.*, 2018] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *International Conference on Learning Representations*, 2018.

[Zhang *et al.*, 2023] Kaiyue Zhang, Weiqi Wang, Zipei Fan, Xuan Song, and Shui Yu. Conditional matching gan guided reconstruction attack in machine unlearning. In *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*, pages 44–49, 2023.

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