Towards Efficient Machine Unlearning with Data Augmentation: Guided Loss-Increasing (GLI) to Prevent the Catastrophic Model Utility Drop

Dasol Choi[1,6] Soora Choi[2] Eunsun Lee[3] Jinwoo Seo[4] Dongbin Na[5†]
[1]Yonsei University [2]Chung Ang University [3]Kyung Hee University [4]Millennialsworks Inc. [5]POSTECH [6]MODULABS

Abstract

Machine unlearning algorithms aim to make a model forget specific data that might be used in the training phase. To solve this problem, various studies have adopted loss-increasing methods. For example, some unlearning methods have presented data augmentation methods to generate synthesized images that maximize loss values for ***images to be forgotten****. In contrast, some unlearning methods directly update the model in the direction of increasing loss for the images to be forgotten. In this paper, we first revisit these loss-increasing methods and analyze their limitations. We have found that these simple loss-increasing strategies can be effective in the aspect of the* ***forgetting score****, however, can hurt the* ***original model utility*** *unexpectedly, we call this phenomenon* ***catastrophic model utility drop****. We propose a novel data augmentation method, Guided Loss-Increasing (GLI), that restricts the direction of the data update to resolve the utility drop issue. This is achieved by aligning updates with the model's existing knowledge, thereby ensuring that the unlearning process does not adversely affect the model's original performance. Our extensive experiments demonstrate our method shows superior (1) model utility and (2) forgetting performance compared to the previous state-of-the-art (SOTA) methods. Furthermore, we demonstrate Jensen–Shannon divergence can be utilized to robustly evaluate the forgetting score. The source codes are publicly available at* https://github.com/Dasol-Choi/Guided_Loss_Increasing.

1. Introduction

With the rapid evolution of computational power and data, deep learning networks stand out as a strong standard for various industries due to their high classification performance. However, the heavyweight of the machine learning model that has been trained on the data containing personal identities can induce *privacy leakage*. To address these issues, regulatory frameworks such as the European Union's General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) have been established [17, 54]. Among the numerous provisions in these regulations, *the right to be forgotten* have stood out, empowering individuals with the authority to request the removal of their data [52]. To resolve this problem, **machine unlearning** has arisen as a crucial research domain and has attracted attention in various industry and research fields.

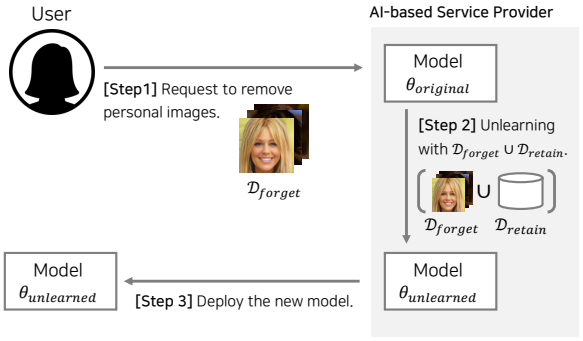


Figure 1. Illustration of a traditional machine unlearning pipeline.

Assume that users want to remove data containing their privacy from the classification model as shown in Figure 1. In this scenario, just removing all the forget data $x_{forget} \in \mathcal{D}_{forget}$ to be forgotten from the originally trained model $\theta_{original}$ is fundamentally difficult. As a naive solution, retraining a model on the data samples to be retained, $x_{retain} \in \mathcal{D}_{retain} = \mathcal{D}_{train} \setminus \mathcal{D}_{forget}$, from scratch can be a possible solution [1, 2]. However, this retraining approach needs all the training data and a burden of computational resources, which might be impractical for real-world scenarios. Therefore, various machine unlearning algorithms have been devised to efficiently adjust (fine-tune) the originally trained model $\theta_{original}$ to unlearn the data to be forgotten.

Specifically, various recent studies have focused on the loss-increasing over the data to be forgotten [9, 50]. These methods have been known to show improved forgetting scores. However, in this work, we show that the **catastrophic model utility drop** that indicates the phenomenon where the classification performance of the model drasti-

†Correspondence to dongbinna@postech.ac.kr

cally degrades in the middle of the training phase can arise. This unexpected outcome might occur due to the improper gradient direction. In this paper, we also demonstrate how the naive loss-increasing method might hurt some feature representations that are useful to the original task. To remedy this issue, we present a novel method, *Guided Loss-Increasing (GLI)*. Our GLI contains the additional **classification loss guidance**, which leads the updated images in the proper direction that does not hurt the original model utility. With extensive experiments, we have demonstrated that our proposed method significantly outperforms the state-of-the-art (SOTA) methods despite its simplicity. Our main contributions are as follows:

- We have found that the previously proposed machine unlearning methods utilizing a loss-increasing approach can suffer from the *catastrophic model utility drop*, thus, we analyze this unexpected outcome.
- In this work, we propose a novel method, guided loss-increasing (GLI) that provides useful gradient guidance when we use the loss-increasing methods.
- Extensively compared to the previous SOTA methods, our method shows superior performance despite the simplicity of the proposed method.

2. Releted Work

2.1. Deep Learning Security and Privacy

The deep learning models have produced remarkable success. However, the AI service provider must consider the personal-privacy leakage before the model deployment [27, 42, 44–46, 55], and model robustness [30] in the security-crucial industries such as identity verification [5, 12, 32–34, 40, 47, 53, 56]. In some previous studies, the over-parameterized large foundation models are also inherent in these vulnerabilities [4, 58]. Moreover, modern deep neural networks are unexpectedly sensitive to a small perturbation of the input data [39, 48]. Model robustness is crucial in the recent machine learning industries, various robust learning methods have been also proposed [6, 18, 31, 43]. In this work, we focus on the *data extraction* attack [3, 11, 27, 38] as a potential threat rather than adversarial attacks.

2.2. Machine Unlearning

With the growth of privacy and security concerns, the user can require their *right to be forgotten* for the AI-based companies utilizing machine unlearning algorithms. However, removing the particular personal identity from the already trained classification models is fundamentally difficult. Thus, various previous studies have proposed diverse methods for machine unlearning such as optimizing the loss function [50, 51], separating the data [1, 37], or manipulating the models [8, 26].

If we can access the whole retain dataset $\mathcal{D}_{retain}$ without any exception, we can adopt the federated learning-based approaches. For example, SISA (Sharded, Isolated, Sliced, and Aggregated) splits the training data into non-overlapping shards or slices and retraining only the data samples that should be retained [1, 37] and achieves improved forgetting performance. Some studies have utilized the teacher-student frameworks to induce the teacher model to transfer knowledge to the student model for retaining the previous information while unlearning the data to be forgotten [8, 26]. Moreover, various machine unlearning algorithms utilize this teacher-student framework to obtain a modest forgetting score. However, they require additional computational resources, more than approximately 2 times, because they can require additional teacher models. Previous work uses a generative model as a teacher model to create synthetic data similar to the original training data and transfer the information worth to be kept to the students with error-minimizing-maximizing noise [9] They also demonstrate that the model can unlearn the data to be forgotten without accessing the training dataset in the *zero-shot scenario*. Moreover, previous work shows that when using neural tangent kernel (NTK) DNNs, machine unlearning can handle better space for weights, which is useful for machine unlearning. As more related to our work, a previous study has proposed a data augmentation-based approach, UNSIR (Unlearning by Selective Impair and Repair) [50]. This work constructs a noise dataset that maximizes the loss values and fine-tunes the model on these synthesized noise samples. This method also allows for the forgetting of data from single or multiple classes without requiring access to the forgotten data. However, they only consider the class-unlearning setting in which the target to remove indicates specific classes. We note that this class-unlearning can show unexpected outcomes because they change the functionality of the model. In this work, we focus on instance machine unlearning without altering the fundamental problem the model addresses. Specifically, we solve the machine unlearning problem as the *task-agnostic problem* [10].

3. Problem Definition

3.1. Task-Agnostic Machine Unlearning

Recent studies have mainly focused on the class-unlearning setting where the target to unlearn is a specific class [7, 14, 15, 50]. However, forgetting specific classes can lead to an output space shift for the model. In these class-unlearning setups, the original task of the model could be drastically changed after the unlearning process, especially for models with small classes [10]. The class-unlearning setup does not address even the binary classification task. In the real-world scenario, various industries utilize many binary classification models including disease prediction

models [23, 25, 35]. Furthermore, the data removal request is not limited to the specific classes. For example, some people request to delete a specific photo that contains the personal identity of a person. Therefore, in this work, we focus on the task-agnostic unlearning setting following a previous study [10]. In the task-agnostic unlearning setup, the *original task* that the model is trained to solve is not modified at all by the machine learning algorithms. Thus, we adopt this setting.

3.2. Machine Unlearning Configuration

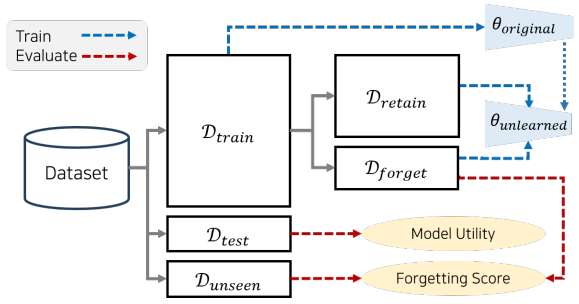


Figure 2. The configuration for machine unlearning datasets.

For the experiments, we split an original dataset into three datasets, the training dataset $\mathcal{D}_{train}$, the test dataset $\mathcal{D}_{test}$, and the unseen dataset $\mathcal{D}_{unseen}$ as shown in Figure 2. The training dataset $\mathcal{D}_{train}$ and the test dataset $\mathcal{D}_{test}$ are used for training classification models. The unseen dataset $\mathcal{D}_{unseen}$ is used for evaluating the forgetting performance of the machine unlearning algorithm. We note that the unseen dataset $\mathcal{D}_{unseen}$ consists of target subjects to be forgotten who do not overlap with the training dataset $\mathcal{D}_{train}$ and test dataset $\mathcal{D}_{test}$. Secondly, for the machine unlearning experiments, we divide training dataset $\mathcal{D}_{train}$ into a retain dataset $\mathcal{D}_{retain}$ and a forget dataset $\mathcal{D}_{forget}$ in a ratio of 85:15. Then, we perform various machine unlearning algorithms using three ingredients (1) the originally trained model $\theta_{original}$ [1] (2) the forget dataset $\mathcal{D}_{forget}$, and (3) the retain dataset $\mathcal{D}_{retain}$ to obtain the unlearned model $\theta_{unlearned}$. Finally, to evaluate the performance of the machine unlearning, we calculate the model utility (accuracy) by testing the unlearned model $\theta_{unlearned}$ on the $\mathcal{D}_{test}$. We also assume an attacker who uses Membership Inference Attack (MIA) [45] distinguishes between the model behaviors on the $\mathcal{D}_{unseen}$ and $\mathcal{D}_{test}$. Specifically, we train an additional binary classifier that classifies the loss values of $x_{test} \in \mathcal{D}_{test}$ and $x_{unseen} \in \mathcal{D}_{unseen}$. We note that the machine unlearning algorithms aim to successfully unlearn the data to be forgotten successful unlearning without performance drop for $\theta_{unlearned}$.

[1]For experiments, we train the original model $\theta_{original}$ that performs classification tasks on the training dataset $\mathcal{D}_{train}$. We start with the machine unlearning at the original model $\theta_{original}$. We explain the $\theta_{original}$ in more detail in Section 5.

4. Proposed Methods

4.1. Preliminaries

4.1.1 Loss Optimization Strategies

Various machine unlearning algorithms adopt similar approaches that increase the loss value for the forget data $x_{forget} \in \mathcal{D}_{forget}$ [50, 51]. This behavior can help the model unlearn the data to be forgotten. For example, we can simply fine-tune the original model $\theta_{original}$ on the retain dataset $\mathcal{D}_{retain}$. By refining the model using the data samples to be retained, the model is expected to re-adjust its model weights [22, 29]. We call this method **Fine-tuning**. This method fundamentally makes the model consider the forgotten data as *unseen* data, which tends to increase the loss for $\mathcal{D}_{forget}$ and could effectively unlearn the model according to the following equation:

$$min_{\theta}\mathbb{E}_{(x,y)\sim\mathcal{D}_{retain}} l(x, y; \theta), \tag{1}$$

Where the l denotes the loss function that can be a standard cross-entropy function. We note that the parameter $\theta_{unlearned}$ for the machine unlearning phase is initialized with its original model weights, $\theta_{original}$. In contrast, we can optimize the model by maximizing the loss [36] directly utilizing the forget dataset $\mathcal{D}_{forget}$. We call this method **NegGrad** following the previous work [15]. Using the NegGrad, the model can directly reduce the noticeable activation for the forget data samples $x_{forget} \in \mathcal{D}_{forget}$ by adopting the gradient ascent in the direction of increasing loss according to the following equation:

$$max_{\theta}\mathbb{E}_{(x,y)\sim\mathcal{D}_{forget}} l(x, y; \theta) \tag{2}$$

4.1.2 Error Maximizing Noises

Recent studies have adopted loss-maximizing data augmentation methods [9, 21, 50]. For example, UNSIR shows utilizing artificial noises that increase the loss value for the forget instances could be effective [9, 50]. They use the following equation to synthesize the artificial noise dataset $\mathcal{D}_{noise}$.

$$\arg\max_{\mathcal{D}_{noise}} \mathbb{E}_{x\sim\mathcal{D}_{Noise}}[l(x, y_{target}; \theta)] \tag{3}$$

In their work, the noise dataset, represented as $\mathcal{D}_{noise}$, encompasses informative features used for removing the features of $\mathcal{D}_{forget}$ by maximizing loss values of the artificial noise samples over the forget samples.

However, the original UNSIR method only considers the class-unlearning scenario for removing the specific target class y_{target}. Thus, their method can not be directly applied

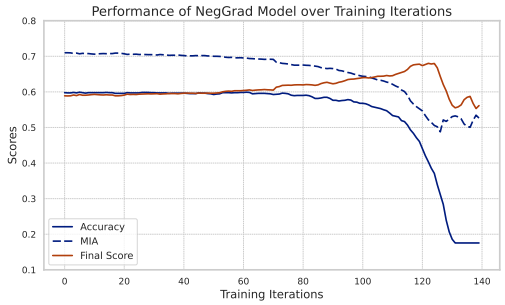


Figure 3. An example of the **catastrophic model utility drop**. The naive loss-increasing method can induce the catastrophic model utility drop. After the 100-th training iteration, the test accuracy on the original task catastrophically decreases. For MIA accuracy, the closer to 0.5 indicates the better, and the higher accuracy indicates the better.

to our task-agnostic setting. Therefore, we introduce the feature level UNSIR method, $\text{UNSIR}_{\text{feature}}$, using the following equation:

$$\arg\max_{\mathcal{D}_{noise}} \mathbb{E}_{x\sim\mathcal{D}_{Noise}}[d(F(x), F(x'); \theta)], \tag{4}$$

where the x denotes the generated data sampled from the $\text{UNSIR}_{\text{feature}}$, $d(\cdot)$ denotes the distance measure function such as the cosine similarity function, and the $x' \in \mathcal{D}_{forget}$ is the samples to be unlearned. For calculating the semantic distance between the two images, we extract the logit feature vectors by forwarding the original image x into the feature extractor parts $F(\cdot)$ (before softmax) of the trained model θ. We note that the UNSIR [50] adopts two stages. UNSIR firstly achieves a high forgetting score using the generated $\mathcal{D}_{noise}$ in stage 1 and leverages the **Fine-tuning** in stage 2.

4.2. Catastrophic Model Utility Drop

In this work, we revisit the loss-increasing methods including the aforementioned NegGrad method. Some previous studies have reported the NegGrad (loss-increasing) or their adaptation could show competitive forgetting performance [10, 36, 50]. However, we have observed the interesting phenomenon that the model utility catastrophically drops in the middle of the training phase. As shown in Figure 3, the model utility drastically drops at a specific point in the training step. Thus, in this work, we return to **NegGrad** and start with this method. Then, we also adopt the data augmentation method.

However, we have found that the simple Gaussian noise of UNSIR [50] might hurt the important semantic information that is related to the original task. Furthermore, directly optimizing $\mathcal{D}_{forget}$ for machine unlearning also poses inherent challenges because the image data x has semantic information itself. Therefore, to solve these challenges,

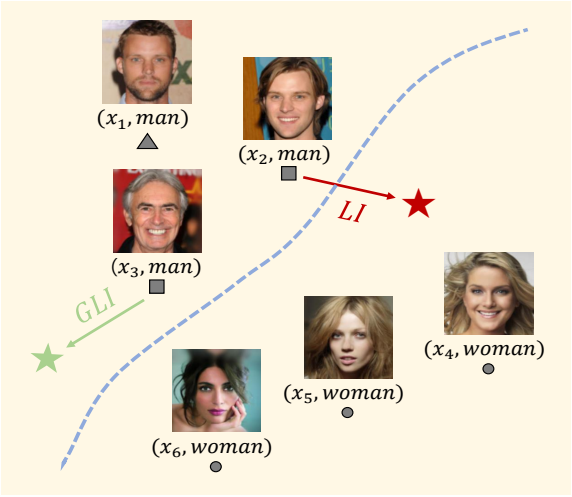


Figure 4. The conceptual illustration that explains our *catastrophic utility drop* hypothesis. The naive Loss-Increasing (LI) approach can hurt the original utility (classification performance) of the model due to the unexpected modification of the task-related semantic features. In contrast, our guided loss-increasing (GLI) method relatively shows superior performance by modifying the task-related features.

we adopt the loss-increasing method with a novel guidance mechanism named Guided Loss-Increasing (GLI).

4.3. Guided Loss-Increasing (GLI)

In this work, we introduce the novel methodology **Guided Loss Increasing (GLI)**, for the pursuit of effective machine unlearning. Our method leverages the classification-loss guidance that might guide the proper gradient direction while preserving the classification accuracy (model utility) of the model. The key point of our approach is to update the augmented images $x_{optimized} \in \mathcal{D}_{retain}$ in the direction of reducing the original classification loss for the true label y. This additional guidance loss serves to pull the perturbed images toward their original class domains, ensuring the original model utility (accuracy). We illustrate the concept of our method as shown in Figure 4. Assume that the image to forget x_{forget} and x_{retain} belongs to the same class. In this setting, naively updating the x_{retain} in the direction of increasing loss values for the x_{forget} might affect the semantic features related to the gender class. In this case, the model tends to pull the features of other instances of the female class toward the male class, which induces the degradation of the model utility. Thus, we utilize the guided classification loss, which makes the optimized data stay in the manifold of semantic features for the true class y_{retain} (male) as shown in Figure 4.

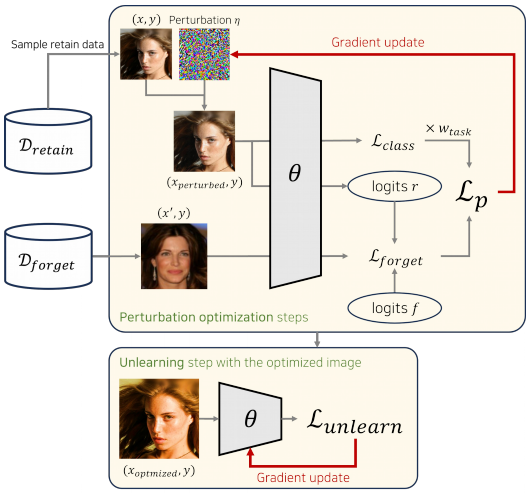


Figure 5. The illustration of our proposed method, GLI, utilizes the joint loss with the guidance loss $\mathcal{L}_{class}$ that could be a simple cross-entropy loss and the feature distancing loss $\mathcal{L}_{forget} = d(\cdot)$.

4.3.1 Class-Loss Guidance

We introduce the class-loss guidance that is straightforward, yet, can be greatly useful when we utilize the loss-increasing methods. Our method uses a retain data x_{retain} as a starting point data for optimization from the $\mathcal{D}_{retain}$. We optimize the data to retain the direction to be far away from a x_{forget} in the feature space:

$$\arg\max_{\eta} \mathbb{E}_{(x,y)\sim\mathcal{D}_{retain}}[d(F(x+\eta), F(x'); \theta) - l(x+\eta, y)] \quad (5)$$

We note that the additional term for class-loss guidance $l(x+\eta, y)$ is extremely useful for obtaining a good model utility while maintaining higher forgetting performance. During the iterative update process of the perturbed images, the addition of this class-loss guidance is crucial to prevent the model from changing the task-related features. Consider we have a gender classification model. When we simply optimize the x_{retain} into the direction of increasing loss for some x_{forget} samples, the semantic features that are related to gender could be changed as shown in Figure 4. In the absence of this guiding loss $l(x+\eta, y)$, the model is susceptible to significant accuracy degradation (model utility drop) due to the over-amplification of the forgetting loss. Therefore, the classification loss $l(x+\eta, y)$ plays an important role in maintaining the accuracy of the model during the machine unlearning process.

4.3.2 Distance Loss for GLI

For our Guided Loss Increasing (GLI) method, the distance loss d is crucial to achieve a high forgetting score. We have extensively explored various distance algorithms to enhance the feature differentiation between the augmented image and forget data across diverse tasks. We have experimented with various loss functions including L_1 norm, L_2 norm, and cosine similarity. Within these distance losses, we have observed subtle variations in performance across different tasks. We have observed that the L_2 norm loss consistently outperforms others in terms of overall performance. The L_2 norm is particularly effective when the data is normalized. Especially each attribute is independent of the other i.e., facial recognition [19, 24, 28]. The L_2 norm also helps mitigate the vanishing gradient problem, maintaining a steady update pace even for features with smaller magnitudes. Thus, we utilize the L_2 norm to measure the feature distance between retain data and forget data. Our GLI aims to increase the L_2 norm between these two samples in the feature space, thus, we expect that the feature of

Algorithm 1 Guided Loss-Increasing (GLI) Single Epoch

Input: Retain dataset $\mathcal{D}_{\text{retain}}$, forget dataset $\mathcal{D}_{\text{forget}}$, the number of iterations N, the number of optimization steps in a epoch T, distance function $d(\cdot)$, original model $\theta_{original}$, the feature extractor $F(\cdot)$ of the $\theta_{original}$, the warming-up schedule function $S(\cdot)$, the step size α, the optimizer for perturbation $O(\cdot)$
Output: The unlearned model $\theta_{unlearned}$
Initialize the model: $\theta_{unlearned} \leftarrow \theta_{original}$
for $iteration = 1, \ldots, N$ **do**
  Sample retain data: $(x_{\text{retain}}, y_{\text{retain}}) \sim \mathcal{D}_{\text{retain}}$
  Sample forget data: $(x_{\text{forget}}, y_{\text{forget}}) \sim \mathcal{D}_{\text{forget}}$
  Generate noise: $\boldsymbol{\eta} \sim \mathcal{N}(0, 0.01^2)$
  Initialize the optimizer: $O.init(\eta)$
  for $i = 1$ to T **do**
    Get the warm-up weight: $w \leftarrow S(i)$
    Perturb retain data: $x_{\text{perturbed}} \leftarrow x_{\text{retain}} + \boldsymbol{\eta}$
    $x_{\text{perturbed}} \leftarrow clip(x_{\text{perturbed}}, 0, 1)$
    Get model predictions:
      $f_{\text{perturbed}} \leftarrow F(x_{\text{perturbed}})$
      $f_{\text{forget}} \leftarrow F(x_{\text{forget}})$
    Calculate the distance loss:
      $\mathcal{L}_{\text{forget}} \leftarrow -1 \times d(f_{\text{perturbed}}, f_{\text{forget}})$
    Calculate the class-guidance loss:
      $\mathcal{L}_{\text{class}} \leftarrow CE(f_{\text{perturbed}}, y_{\text{retain}})$
    Compute the total perturbation loss:
      $\mathcal{L}_{\text{perturb}} \leftarrow w \cdot \mathcal{L}_{\text{class}} + (1-w) \cdot \mathcal{L}_{\text{perturb}}$
    Perform back-propagation and optimize:
      Clear gradients: $O.zerograd()$
      Compute gradients: $\mathcal{L}_{\text{perturb}}.backward()$
      Update noise: $\boldsymbol{\eta} \leftarrow \boldsymbol{\eta} - \alpha \cdot \text{sign}(\nabla \boldsymbol{\eta})$
  end for
  Perturb retain data: $x_{\text{perturbed}} \leftarrow x_{\text{retain}} + \boldsymbol{\eta}$
  $x_{\text{perturbed}} \leftarrow clip(x_{\text{perturbed}}, 0, 1)$
  Update $\theta_{unlearned}$ with $(x_{perturbed}, y_{retain})$
end for

$x_{optimized}$ is far from the representations of $\mathcal{D}_{forget}$.

4.3.3 Class-Loss Warming-Up

To mitigate the rapid escalation of forgetting loss during batch-wise training, we employ a *class-loss warming-up* strategy. This strategy does not apply a uniform class-loss guidance weight across all batches. Instead, our method GLI increases the guide loss incrementally according to the development of the training procedure. To modulate the guide loss across training batches, we employ a linear weight scheduler. This scheduler linearly increases the weight applied to the guide loss from an initial weight w_{init} to a final weight w_{final} throughout training. For a given batch index i, with $i = 0$ representing the first batch, the weight w_i is computed as follows:

$$w_i = w_{\text{init}} + \left(\frac{w_{\text{final}} - w_{\text{init}}}{N - 1} \right) i, \tag{6}$$

where w_{init} is the initial weight and w_{final} is the final weight. Moreover, the N denotes the total number of batches and the i is the current batch index. This linear increase in weight ensures that the guide loss's influence on the training process grows progressively, allowing the model to obtain good classification accuracy while maintaining its ability to forget. Finally, we fine-tune the model $\theta_{original}$ on the generated samples $x_{perturbed} = x_{retain} + \eta$ to forget the data to be forgotten while maintaining the original model classification performance. With extensive experiments, we observe that our min-max optimization formulation can achieve greatly high performance and also shows empirically fast convergence compared to the other SOTA methods.

5. Experiments

We start with the machine unlearning at the original model $\theta_{original}$ and aim to obtain the optimized $\theta_{unlearned}$. Moreover, we retrain the original task only using the retain dataset $\mathcal{D}_{retain}$ from scratch. We note that the $\theta_{retrained}$ serves as the ground-truth model we want to obtain by the machine unlearning process.

5.1. Training the Original Models

We train three state-of-the-art (SOTA) deep learning models: ResNet18 [20], WideResNet [57], and EfficientNet [49] to solve original classification tasks. We experiment with a batch size of 64 and a learning rate of 0.01. Specifically, we have mainly trained the two models: the age prediction model and the multi-task classification model utilizing these architectures. We adopt recently presented datasets named *MUFAC* and *MUCAC* to obtain original models and to conduct the machine unlearning algorithms following the previous work [10]. For the *MUFAC* benchmark, the model solves the multi-class classification task, age recognition. The age classification task consists of 8 classes from class 0 to class 7 where each class indicates a specific range of ages. For example, the *class 2* represents 13∼19 years old. Moreover, the *MUCAC* benchmark deals with three binary labels simultaneously, male/female, old/young, and smiling/unsmiling for the binary classification tasks. We note that the *MUFAC* and *MUCAC* include 13,068 and 30,000 "human face" images respectively. All facial images have resolutions of 128 x 128. Furthermore, all the images consist of the *personal identity number* as a label additionally. Specifically, *MUFAC* have several data (x, $y^1 = identity$, $y^2 = age$). Moreover, the *MUCAC* consists of a bunch of data (x, $y^1 = identity, y^2 = gender$, $y^3 = smiling$, $y^4 = age$) similarly to the *MUFAC*.

5.2. Evaluation Protocol

As the reliable performance measurement of machine unlearning, we focus on two primary considerations: (1) classification performance (model utility) and (2) the efficacy of the unlearning (forgetting score). After the machine unlearning process, the model should retain robust classification capabilities for the original task while effectively removing information of $\mathcal{D}_{forget}$ simultaneously. To meet these criteria, we employ two distinct evaluation metrics.

5.2.1 Classification Accuracy for Model Utility

By modifying the original model, $\theta_{original}$, we obtain the unlearned model $\theta_{unlearned}$. One of our primary goals is to maintain the good classification accuracy of the unlearned model, $\theta_{unlearned}$. Specifically, we measure the test accuracy of the unlearned model using the test set $\mathcal{D}_{test}$. Because the model's fundamental purpose is to identify the ground-truth label given an image, simply evaluating using $\mathcal{D}_{test}$ can be suitable. Thus, we employ the following equation for each (x, y) within the $\mathcal{D}_{test}$:

$$P(\arg\max_{\hat{y}} P(\hat{y}|x; \theta_{unlearned}) = y), \tag{7}$$

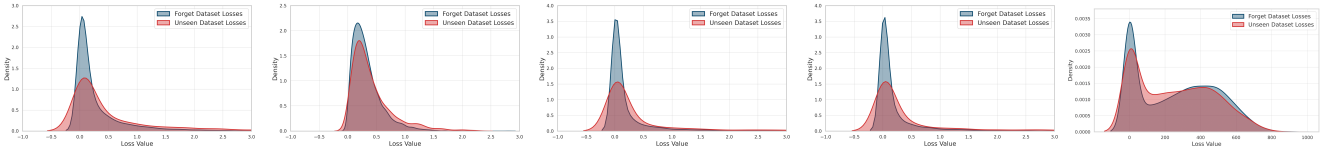
where y denotes the ground-truth label and $\hat{y}$ indicates the predicted label.

5.2.2 MIA for Forgetting Metric

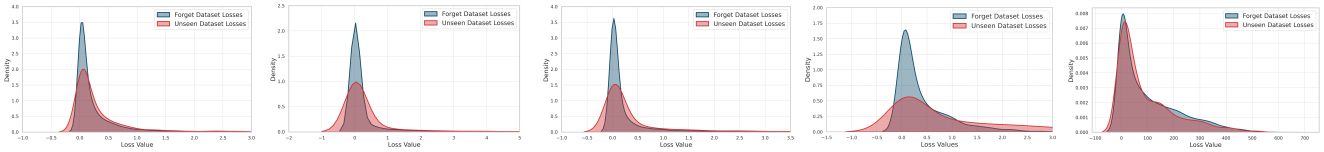
Ideally, the behavior of $\theta_{unlearned}$ should align closely with that of a retrained model, $\theta_{retrained}$, which is trained only using the dataset $\mathcal{D}_{retain}$. For example, for effective unlearning, $\theta_{unlearned}$ should exhibit unnoticeable behavior when tested on both $\mathcal{D}_{unseen}$ and $\mathcal{D}_{forget}$ i.e., the similar loss distributions. To assess this, we employ the membership inference attack (MIA) as our evaluation metric for forgetting performance like the various previous studies [10, 45, 50]. The MIA trains a simple logistic regres-

Table 1. Overall performance for two main classification tasks. We also report the detailed performance in the supplementary materials.

	Metrics	Original	Retrained	Fine-tunning [16]	CF-K [14]	NegGrad [15]	UNSIR [50]	SCRUB [26]	Bad Teaching [8]	EU-K	**Ours (GLI)**
MUFAC (multi-class)	Test Acc. ↑	0.5952	0.4880	0.6049	0.5900	0.4048	0.5925	0.5984	0.5477	0.5737	0.5685
	Top-2 Acc. ↑	0.8804	0.7667	0.8869	0.8804	0.5932	0.8674	0.8745	0.8226	0.8473	0.8362
	Forgetting Score ↓	0.2136	0.0445	0.1953	0.2126	0.0485	0.1990	0.1415	0.1714	0.1033	0.0305
	Final Score ↑	0.5839	0.6995	0.6071	0.5824	0.6539	0.5972	0.6577	0.6705	0.5855	**0.7538**
MUCAC (multi-label)	Average Test Acc.↑	0.9073	0.8871	0.9213	0.9228	0.7351	0.9218	0.9078	0.8066	0.886	0.9108
	Forgetting Score↓	0.0319	0.0032	0.0102	0.0226	0.0152	0.0164	0.0266	0.0127	0.0448	0.0032
	Final Score↑	0.9217	0.9403	0.9504	0.9388	0.8523	0.9445	0.9273	0.8906	0.8982	**0.9522**



(a) Original (b) Retrained (c) Fine-tuning (d) CF-K (e) NegGrad



(f) UNSIR (g) SCRUB (h) EU-K method (i) Bad Teaching (j) **Ours (GLI)**

Figure 6. Forgetting performance comparison with other methods. The horizontal axis is loss values and the vertical axis is density. The blue line is *forget dataset losses* and the red line is *unseen dataset losses*. Our GLI method shows very similar aspects in the distribution shapes of the *Retrained* model, which shows nearly perfect forgetting performance in the MIA measurement.

sion model on the loss values from both the training forget dataset $\mathcal{D}_{forget}$ and the unseen dataset $\mathcal{D}_{unseen}$. If the machine unlearning is ideally performed, the MIA attack success rate is 0.5. Thus, we adopt the **forgetting score** as $abs(0.5 - M)$ where M denotes the accuracy of the MIA model. For the forgetting score, the lower is better.

5.2.3 Main Experimental Results

For the total experiments, we set the **Final Score**[2] to {*Test Acc* + (1 - *Forgetting Score* * 2)} / 2. We have found this simple comprehensive metric is greatly effective for comparing various machine unlearning algorithms. Moreover, the accuracy and MIA serve as proxy evaluation metrics. With extensive experiments with the various benchmark datasets, we demonstrate our method achieves significantly superior performance for all the experiments as shown in Table 1. Specifically, our method (GLI) shows superior unlearning performance on both *MUFAC* and *MUCAC* compared to the recently proposed SOTA methods including recent SCRUB [26], UNSIR [50] and Bad Teaching [8]. Furthermore, we show that the performance on the original task remains comparable to the original model. Moreover, interestingly, our proposed method shows very similar loss distributions between $\mathcal{D}_{forget}$ and $\mathcal{D}_{unseen}$ as shown in Figure 6. These results represent that our method is a good machine unlearn algorithm, by showing that our model has almost completely forgotten the forget data while maintaining the model utility. Additionally, the naive **Fine-tuning** can be a also strong baseline. We report more detailed experimental results in the supplementary material, including performance for single-task models and overall performance based on WideResNet and EfficientNet architectures.

[2]The maximum value of forgetting score is 0.5, and closer to 0 indicates better performance. To calculate the average with test accuracy, which has a maximum value of 1, we multiply the forgetting score by 2. We then subtract it from 1 to indicate that the larger the number, the better the performance.

6. Discussion

With data security issues on the rise, machine unlearning is becoming an increasingly important challenge to protect privacy and prevent sensitive information from being exposed. However, studies in this area are still underdeveloped compared to its importance. In this work, we have found that the MIA based on the logistic regression can not capture the difference between two distributions. Thus, we have also explored various robust metrics for evaluating the machine unlearning forgetting score.

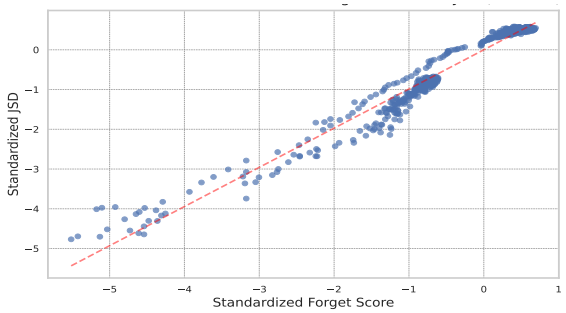
6.1. Jensen-Shannon Divergence (JSD)

While Membership Inference Attacks (MIA) have been prevalently used as a metric to measure the effectiveness

Table 2. Overall forgetting performance across the different dataset benchmarks. We note that the naive MIA metric might not capture the distribution difference properly. For example, our proposed method is sometimes better than the *Retrained* model in the MIA metric, which does not make sense in human perception. The JSD could act as robust metrics and show consistently feasible forgetting scores.

	Metrics	Original	Retrained	Fine-tunning [16]	CF-3 [14]	NegGrad [15]	UNSIR [50]	SCRUB [26]	Bad Teaching [8]	EU-K [14]	**Ours (GLI)**
MUFAC (multi-class)	Forgetting Score	0.2136	0.0445	0.1953	0.2126	0.0485	0.1990	0.1415	0.1714	0.2013	0.0305
	JSD	0.6561	0.0633	0.6332	0.6564	0.0544	0.6086	0.4754	0.4226	0.5958	0.1103

Figure 7. The correlation between our forgetting score and JSD. (γ =0.92)



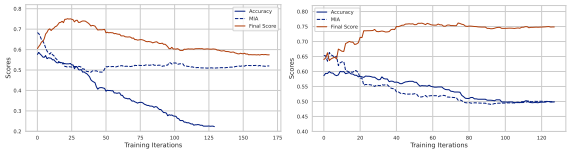
of machine unlearning [10, 45, 50], they still have limitations [41]. Traditional MIA approaches rely on binary classification to determine the membership status of data, which can oversimplify the nuanced distribution of loss values. Such simplification may not accurately reflect the complexity of models' forgetting behavior. To address these issues, we adopt the Jensen-Shannon Divergence (JSD) as a new metric for the machine unlearning research fields and show their robustness. We expect JSD that capture a more nuanced and multidimensional view of loss distributions. Unlike MIA, JSD considers the entire *density* of a probability distribution over the loss values, providing a more comprehensive assessment of how well a model has unlearned the data to be forgotten as shown in Table 2. As an advantage, the JSD evaluates the similarity between the loss distributions of the forgotten and unseen data, providing a distributional perspective rather than a mere point estimate. Moreover, the JSD is adaptable to various loss functions [13] and is not limited to binary outcomes, making it suitable for a wide range of unlearning scenarios. The calculation of JSD is performed by employing kernel density estimation to approximate the probability distributions of loss values and the Jensen-Shannon divergence to measure the distance between these distributions. The mathematical formulation is given as:

$$\mathrm{JSD}(\mathcal{Z}_{\text{forget}}, \mathcal{Z}_{\text{unseen}}) = \sqrt{\frac{\mathrm{KL}(\mathcal{Z}_{\text{forget}} \| \mathcal{H}) + \mathrm{KL}(\mathcal{Z}_{\text{unseen}} \| \mathcal{H})}{2}} \quad (8)$$

where $\mathcal{Z}_{\text{forget}}$ and $\mathcal{Z}_{\text{unseen}}$ indicate the loss distributions of the forgotten and unseen data, respectively, $\mathcal{H}$ is the mean of these two distributions, and KL denotes the Kullback-Leibler divergence. In our experiments, we have observed that the JSD measure is highly correlated to our forgetting score with Pearson Correlation Coefficient $\gamma = 0.92$.

Table 3. Efficiency of the Guidance Loss with Class Warming-up

	Metrics	LI	GLI with Fixed Class Weight	GLI with Class Warm-up
MUFAC (multi-class)	Test Acc. ↑	0.5230	0.5692	0.5685
	Top-2 Test Acc.↑	0.8128	0.8401	0.8362
	Forgetting Score ↓	0.0448	0.0823	0.0305
	Final Score↑	0.7166	0.7023	**0.7537**
MUCAC (multi-label)	Average Test Acc. ↑	0.8595	0.9170	0.9108
	Forgetting Score↓	0.0017	0.0144	0.0032
	Final Score ↑	0.9280	0.9440	**0.9522**



(a) (b)

Figure 8. Effect of Class Warming-up in Our GLI. (a) shows the score curves over the training iterations of GLI with *fixed weight for the class-loss guidance*. (b) represents the score curves over the training iterations of GLI with *class-loss warming-up*.

7. Conclusion

In this paper, we propose a novel method, Guided Loss-Increasing (GLI), that updates the images to be retained towards increasing the loss value of the target instances to be forgotten. Our experiments demonstrate these augmented images are greatly effective in achieving enhanced machine unlearning performance. In this work, data augmentation can be a crucial key for unlearning images to forget by consistently giving the model samples that should be trained. Moreover, we adopt a novel JSD-based measurement to calculate the forgetting performance and show this metric can be useful to robustly evaluate the unlearning performance given a trained model. For future work, we will explore more efficient machine unlearning algorithms and robust metrics to evaluate the unlearning performance.

8. Acknowledgements

This work was supported by Brian Impact, a non-profit organization dedicated to advancing science and technology.

References

[1] Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning, 2020. 1, 2

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

[3] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021. 2

[4] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650, 2021. 2

[5] Jun-Cheng Chen, Vishal M. Patel, and Rama Chellappa. Unconstrained face verification using deep cnn features. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, 2016. 2

[6] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning, 2017. 2

[7] Rishav Chourasia and Neil Shah. Forget unlearning: Towards true data-deletion in machine learning, 2023. 2

[8] Vikram S Chundawat, Ayush K Tarun, Murari Mandal, and Mohan Kankanhalli. Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7210–7217, 2023. 2, 7, 8

[9] Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan Kankanhalli. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*, 18:2345–2354, 2023. 1, 2, 3

[10] Dongbin Na Dasol Choi. Towards machine unlearning benchmarks: Forgetting the personal identities in facial recognition systems, 2023. 2, 3, 4, 6, 8

[11] Luigi De Angelis, Francesco Baglivo, Guglielmo Arzilli, Gaetano Pierpaolo Privitera, Paolo Ferragina, Alberto Eugenio Tozzi, and Caterina Rizzo. Chatgpt and the rise of large language models: the new ai-driven infodemic threat in public health. *Frontiers in Public Health*, 11, 2023. 2

[12] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 2

[13] Erik Englesson and Hossein Azizpour. Generalized jensen-shannon divergence loss for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34: 30284–30297, 2021. 8

[14] Shashwat Goel, Ameya Prabhu, and Ponnurangam Kumaraguru. Evaluating inexact unlearning requires revisiting forgetting. *arXiv preprint arXiv:2201.06640*, 2022. 2, 7, 8

[15] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312, 2020. 2, 3, 7, 8

[16] Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations. In *ECCV 2020: 16th European Conference, Glasgow, UK, 2020, Proceedings*, pages 383–398. Springer, 2020. 7, 8

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

[18] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 5

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[21] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. *CoRR*, abs/2101.04898, 2021. 3

[22] Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning, 2023. 3

[23] Mehdi Khashei, Saeede Eftekhari, and Jamshid Parvizian. Diagnosing diabetes type ii using a soft intelligent binary classification model. *Review of Bioinformatics and Biometrics*, 1(1):9–23, 2012. 3

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2012. 5

[25] Roshan Kumari and Saurabh Kr Srivastava. Machine learning: A review on binary classification. *International Journal of Computer Applications*, 160(7), 2017. 3

[26] Meghdad Kurmanji, Peter Triantafillou, and Eleni Triantafillou. Towards unbounded machine unlearning. *arXiv preprint arXiv:2302.09880*, 2023. 2, 7, 8

[27] Haoran Li, Dadi Guo, Wei Fan, Mingshi Xu, Jie Huang, Fanpu Meng, and Yangqiu Song. Multi-step jailbreaking privacy attacks on chatgpt, 2023. 2

[28] Jing Li and Bao-Liang Lu. An adaptive image euclidean distance. *Pattern Recognition*, 42(3):349–357, 2009. 5

[29] 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 (ICCV)*, pages 4892–4902, 2023. 3

[30] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning

models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 2

[31] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019. 2

[32] Diarra Mamadou, Kacoutchy Jean Ayikpa, Abou Bakary Ballo, Brou Medard Kouassi, et al. Application of three convolutional neural network algorithms for occluded face identification and recognition for system security. *American Journal of Multidisciplinary Research and Innovation*, 1(5): 24–32, 2022. 2

[33] Saibal Manna, Sushil Ghildiyal, and Kishankumar Bhimani. Face recognition from video using deep learning. In *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, pages 1101–1106, 2020.

[34] Li Mao, Fusheng Sheng, and Tao Zhang. Face occlusion recognition with deep learning in security framework for the iot. *IEEE Access*, 7:174531–174540, 2019. 2

[35] Jae H Min and Chulwoo Jeong. A binary classification method for bankruptcy prediction. *Expert Systems with Applications*, 36(3):5256–5263, 2009. 3

[36] Quoc Phong Nguyen, Bryan Kian Hsiang Low, and Patrick Jaillet. Variational bayesian unlearning. *Advances in Neural Information Processing Systems*, 33:16025–16036, 2020. 3, 4

[37] Chao Pan, Jin Sima, Saurav Prakash, Vishal Rana, and Olgica Milenkovic. Machine unlearning of federated clusters. 2023. 2

[38] Xudong Pan, Mi Zhang, Shouling Ji, and Min Yang. Privacy risks of general-purpose language models. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 1314–1331. IEEE, 2020. 2

[39] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016. 2

[40] VM Praseetha, Saad Bayezeed, and S Vadivel. Secure fingerprint authentication using deep learning and minutiae verification. *Journal of Intelligent Systems*, 29(1):1379–1387, 2019. 2

[41] Shahbaz Rezaei and Xin Liu. On the difficulty of membership inference attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7892–7900, 2021. 8

[42] Radi P. Romansky and Irina S. Noninska. Challenges of the digital age for privacy and personal data protection. *Mathematical Biosciences and Engineering*, 17(5):5288–5303, 2020. 2

[43] Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *CoRR*, abs/1804.00792, 2018. 2

[44] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, page 1310–1321, New York, NY, USA, 2015. Association for Computing Machinery. 2

[45] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017. 3, 6, 8

[46] H Jeff Smith, Sandra J Milberg, and Sandra J Burke. Information privacy: Measuring individuals' concerns about organizational practices. *MIS quarterly*, pages 167–196, 1996. 2

[47] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. *Advances in neural information processing systems*, 27, 2014. 2

[48] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2014. 2

[49] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019. 6

[50] Ayush K Tarun, Vikram S Chundawat, Murari Mandal, and Mohan Kankanhalli. Fast yet effective machine unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, 2023. 1, 2, 3, 4, 6, 7, 8

[51] Ayush K Tarun, Vikram S Chundawat, Murari Mandal, and Mohan Kankanhalli. Deep regression unlearning, 2023. 2, 3

[52] Eduard Fosch Villaronga, Peter Kieseberg, and Tiffany Li. Humans forget, machines remember: Artificial intelligence and the right to be forgotten. *Computer Law & Security Review*, 34(2):304–313, 2018. 1

[53] Ngonadi I Vivian and Orobor Anderson Ise. Face recognition service model for student identity verification using deep neural network and support vector machine (svm). *Int J Sci Res Comput Sci Eng Inf Technol*, 6(4):11–20, 2020. 2

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

[55] Jennifer Williams, Karla Pizzi, Shuvayanti Das, and Paul-Gauthier Noé. New challenges for content privacy in speech and audio. In *2nd Symposium on Security and Privacy in Speech Communication*. ISCA, 2022. 2

[56] Xing Wu, Jianxing Xu, Jianjia Wang, Yufeng Li, Weimin Li, and Yike Guo. Identity authentication on mobile devices using face verification and id image recognition. *Procedia Computer Science*, 162:932–939, 2019. 2

[57] Zifeng Wu, Chunhua Shen, and Anton Van Den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition*, 90:119–133, 2019. 6

[58] Weichen Yu, Tianyu Pang, Qian Liu, Chao Du, Bingyi Kang, Yan Huang, Min Lin, and Shuicheng Yan. Bag of tricks for training data extraction from language models. In *Proceedings of the 40th International Conference on Machine Learning*, 2023. 2