Записки научных
семинаров ПОМИ
Том 540, 2024 г.

A. Zhavoronkin, M. Pautov, N. Kalmykov, E. Sevriugov, D. Kovalev, O.Y. Rogov, I. Oseledets

UNGAN: MACHINE UNLEARNING STRATEGIES THROUGH MEMBERSHIP INFERENCE

Abstract. As regulatory and ethical demands for data privacy and the right to be forgotten increase, the ability to effectively unlearn specific data points from machine learning models without retraining from scratch becomes paramount. Machine unlearning aims to efficiently eliminate the influence of certain data points on a model. We propose the **UnGAN**, a novel approach to machine unlearning that leverages Generative Adversarial Networks (GANs) to address the growing need for efficient and reliable data removal from trained models. UnGAN proposes a unique unlearning strategy through membership inference, where a discriminator network is trained to identify whether a given input was part of the model's training set. The discriminator is a three-layer fully connected network employing ReLU activation functions, receiving inputs from the output of the model undergoing unlearning and the class label. This architecture enables the discriminator to learn the membership status of data points with high precision, thereby guiding the unlearning process.

1. Introduction

Recently, neural network-based algorithms have achieved remarkable success in many areas of machine learning (ML), including natural language processing [1], computer vision [2] and generative artificial intelligence [3]. To achieve such success, neural networks have had to significantly increase in size; the largest architectures now have billions of parameters [4]. As a consequence, ML algorithms have become able to remember specific training samples, possibly threatening the privacy of personal information, especially in cases related to finances and healthcare. Since it may now be possible to find out whether a particular sample belongs to the training set of a model [5, 6], it has become crucial to remove certain data samples from ML models and systems on request [7] to avoid privacy violations.

Key words and phrases: Machine unlearning, generative adversarial networks, deep learning, trustworthy AI.

Unfortunately, data removal from ML systems is a challenging task: it is not enough to delete the data record from a training database because the model itself is prone to memorizing the data. To satisfy the user's right "to be forgotten", the process of *unlearning* has to be applied to the model. Importantly, unlearning of a specific data record should satisfy several crucial properties: (i) it should not affect the performance of the model on the data that should not be removed, (ii) should be verifiable, i.e., an end-user should have a mechanism to verify that their data is removed from the ML system, and (iii) should not be costly, i.e., should not require the full retraining of the model.

In this work, we propose UnGAN, an approach to machine unlearning that interprets training item removal as an adversarial game: given the training dataset $\mathcal{D}$, the forget set $\mathcal{D}_f$, the retain set $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$ and the hold-out unseen dataset $\mathcal{D}_u$, the initial model $\hat{G}$ is fine-tuned to have similar behavior on $\mathcal{D}_u$ and $\mathcal{D}_f$ while retaining its predictions on $\mathcal{D}_r$. At the same time, a separate small discriminative network D is trained to distinguish between $\mathcal{D}$ and $\mathcal{D}_u$. To assess the efficiency of the proposed approach, we conduct the membership inference attack (MIA) on the modified model $\hat{G}$: intuitively, if the training item removal is successful, the membership inference attack should indicate that the set $\mathcal{D}_f \cup \mathcal{D}_u$ is not part of $\mathcal{D}$, but that the set $\mathcal{D}_r$ is part of $\mathcal{D}$.

Our contributions can be summarized as follows:

• we propose an approach for machine unlearning that leverages an auxiliary generative adversarial network;

• we compare the proposed method to the baselines and experimentally show that it is effective both in terms of the success of the item removal task and computation time;

• to assess the quality of the proposed method, we evaluate the effectiveness of the membership inference attack on the unlearned model, showing that the performance of the proposed approach is comparable to the gold standard of machine unlearning – the full retraining of the model.

2. Related Work

2.1. Machine Unlearning. Machine unlearning was formulated as a data-forgetting algorithm in statistical query learning [19]. Brophy and Lowd [21] introduced a variant of random forests that supports data forgetting with minimal retraining. Data deletion in k-means clustering was studied in [10, 18]. The results in [11] give a certified information removal

framework based on Newton's update removal mechanism for convex learning problems. The data removal is certified using a variation of the differential. Another notable solution [13] presents a projective residual update method to delete data points from linear models. A method to hide the class information from the output logits is presented in [12]. This, however, does not remove the information present in the network weights. Unlearning in a Bayesian setting using variational inference is explored for regression and Gaussian processes in [15]. The authors of [14] study the results of a gradient descent-based approach to unlearning in convex models. All these methods are designed for convex problems, whereas we aim to present an unlearning solution for deep learning models.

Efforts to facilitate unlearning have also considered the strategic segmentation of training data, thereby allowing for easier data disassociation and minimizing its impact on the learning process [17, 16]. Such strategies, however, lead to considerable storage requirements due to the necessity to maintain various model and gradient snapshots to achieve effective unlearning. These techniques are designed to be agnostic to the learning algorithm employed and rely on an efficient partitioning of training data. In addition, they involve retraining certain model segments, contrary to our goal of devising an unlearning method that incurs no additional memory costs. Gupta et al. [23] have put forward a strategy for managing a series of adaptive deletion requests, presenting a new direction in this field.

2.2. Generative Adversarial Networks. Generative adversarial neural networks (GANs) [8] are well known for their ability to reproduce complex data distributions through optimization of min-max loss. It consists of two parts: Generator and Discriminator. The goal of the discriminator is to distinguish between real and generated samples. It is done by maximizing discriminator scores for input corresponding to real data distribution and minimizing them for the generated one. In contrast, the generator tries to fool the discriminator by maximizing its score for fake samples. Finally, the generator output distribution becomes close to the real one. Min-max objectives could be formulated in different ways, leading to a variety of GAN types. Basic GAN minimizes the Jensen–Shannon divergence between real and fake samples but may suffer from mode collapse, when the same class outputs are generated by different inputs from the input space [30]. This problem was solved in [31] by training GAN (CGAN) in a class-conditional manner. Despite being a simple technique, it has proven

to be sufficient prevent mode collapse. Another disadvantage of the basic Wasserstein generative adversarial networks is the problem of biased sample gradients. This issue was fixed in [35] by integrating the Cramer distance into the vanilla Wasserstein GAN.

2.3. Membership Inference Attacks. The goal of a membership inference attack (MIA) is to determine whether a particular sample was used to train the target model or not [24]. Machine learning models of different architectures tend to be vulnerable to this attack. Hence, MIAs have become one of the most widely studied classes of privacy attacks [25]. The majority of works on membership inference attacks leverage the information about the model's loss [26], confidence [27], or entropy [28]. Since membership inference attack is the detection problem, one of the most important questions is how to evaluate different MIAs approaches. It was recently claimed [6] that the correct way to evaluate the effectiveness of the attack is to compute true positive rates on low false positive rates. Another way to conduct membership inference attacks is to leverage likelihood ratio attack (LiRA) [6]. In a nutshell, given the test target model $\hat{G}$, its unknown training dataset D, a sample of interest (x, y) and test statistic $s(x, y)$, LiRA performs hypothesis test between two hypotheses: $H_0 : (x, y) \notin D,\ H_1 : (x, y) \in D$. It is known [29] that the optimal hypothesis test is based on computing the likelihood ratio of the test statistic under the null and alternative hypothesis. Since the exact likelihoods are unknown, they are estimated based on the values of $s(x, y)$ computed by the collection of the shadow models [5, 6].

3. Method

Suppose that the entire dataset $\mathcal{D}$ is split into two non-intersecting subsets: the forget set $\mathcal{D}_f$ and the retain set $\mathcal{D}_r$. Given a model $\hat{G}$ trained on the whole set $\mathcal{D}$, the goal of the machine unlearning task is to obtain a model G that performs on the subset $\mathcal{D}_f$ as the model $\hat{G}$ on the hold-out dataset $\mathcal{D}_u \notin \mathcal{D}$, but at the same time performs as $\hat{G}$ on the dataset $\mathcal{D}_r$.

To evaluate an unlearning approach, one can measure the distance between model output distributions obtained on unseen set and forget set, $\rho_u^G = \rho(G(\mathcal{D}_u))$ and $\rho_f^G = \rho(G(\mathcal{D}_f))$, respectively. Notably, the smaller the distance $d(\rho_u^G, \rho_f^G)$, the more similar the performance of model G on $\mathcal{D}_f$ to the performance of $\hat{G}$ on $\mathcal{D}_u$. There are several known options to measure

the distance between two probability distributions, including the Kullback–Leibler (KL) divergence, Jensen-Shannon divergence (JSD), Wasserstein distance, and Cramer distance.

In our work, we use Wasserstein and Cramer distances, since the KL divergence and JSD are not everywhere differentiable functions (for a locally Lipschitz function G, [9]). The practical definition of p-Wasserstein distance W_p is the following:

$$W_p(P,Q) = \left(\int_0^1 \left| F_P^{-1}(u) - F_Q^{-1}(u) \right|^p du \right)^{1/p}, \tag{1}$$

where F_P is the cumulative distribution function of the distribution P. In practice, the distribution P is represented by set of samples $P_M = \{X\}_{m=1}^M$. Thus, to approximate the distribution P via parametric distribution Q_θ the expression from Eq. (1) is minimized with the inverse cumulative distribution function of P replaced with its empirical counterpart.

To speed up the convergence of the parametric distribution Q_θ to the empirical counterpart P_M of an unknown distribution P, we use Cramer distance $l_p(P,Q)$ in the form

$$\text{CramerDistance}(P,Q) = \left(\int_{-\infty}^{+\infty} |F_P(x) - F_Q(x)|^p \, dx \right)^{1/p}. \tag{2}$$

Gradients of the Cramer distance are unbiased [35], namely,

$$\nabla_\theta \text{CramerDistance}(P, Q_\theta) = \mathbb{E}_X \nabla_\theta \text{CramerDistance}(P_M, Q_\theta). \tag{3}$$

To perform the machine unlearning task removal, we minimize the distance defined in (2). We formulate this minimization in the context of training a separate Generative Adversarial Network (GAN).

3.1. UnGAN – Unlearning via GAN. In general, GAN is trained to reproduce some complex distribution ρ_x by finding optimal generator G^* such that the distribution $\rho(G(z))$ is approximately equal to ρ_x:

$$G^* = \arg\min_G \text{CramerDistance}(\rho_{G(z)}, \rho_x), \tag{4}$$

where ρ_x represents the distribution of the real data, and $z \sim \rho_z$, where ρ_z represents the distribution of the generator's input data.

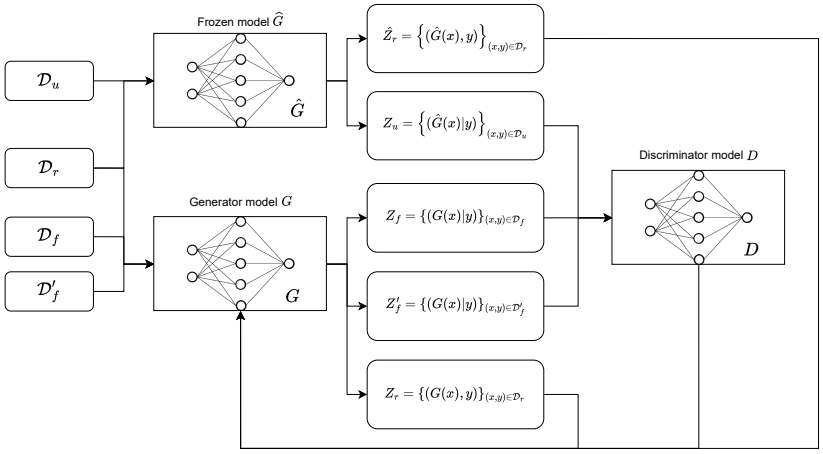

Figure 1. Illustration of the proposed method. Given the original frozen model $\hat{G}$ trained on some dataset $\mathcal{D}$, the hold-out unseen dataset $\mathcal{D}_u : \mathcal{D}_u \cap \mathcal{D} = \emptyset$, the forget set $\mathcal{D}_f \subset \mathcal{D}$, the retain dataset $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$, our approach is to train a separate (generator) model G to (i) retain the behavior of $\hat{G}$ on dataset $\mathcal{D}_r$ while (ii) achieving the same behavior on the dataset $\mathcal{D}_f$ as on $\mathcal{D}_u$, while training a separate (discriminator) network D to distinguish between the outputs of generator on sets $\mathcal{D}_u$ and $\mathcal{D}_f$. In the notation, "|" denotes concatenation.

An explicit calculation of the Cramer distance is infeasible. However, using the surrogate energy functional [35] in the form

$$G^* = \arg\min_G d(\rho_{G(z)}, \rho_x) = \arg\min_G \left[\mathbb{E}_{x\sim\rho_x} f^*(x) - \mathbb{E}_{z\sim\rho_z} f^*(G(z))\right] \quad (5)$$

yields an optimization problem with the same solution as (4). Here, $f^*(x) = \mathbb{E}_{z'\sim\rho_z}\|x - G(z')\|_2 - \mathbb{E}_{x'\sim\rho_x}\|x - x'\|_2$ is the critic function.

According to [35], when the raw data representations are transformed into embeddings, it may prevent the overfitting to empirical data distribution. In practice, the training of the embedding function D by maximizing the Cramer distance yields more diverse embeddings.

In the setting of UnGAN, the real data samples z_u are the outputs of the initial model $\hat{G}$ on unseen dataset $\mathcal{D}_u$. Similarly, the fake data samples z_f

are the outputs of the generator model G on the forget dataset $\mathcal{D}_f$. Hence, the generator objective is to minimize the Cramer distance between the outputs of the initial model on samples from $D(z_u)$ and the outputs of the generator on samples from $D(z_f)$:

$$G^* = \arg\min_G \|D(z_u) - D(z_f)\|_2 + \|D(z_u) - D(z'_f)\|_2 - \|D(z_f) - D(z'_f)\|. \quad (6)$$

Here, $z_f = G(x_f)$ and $z'_f = G(x'_f)$, where $x_f \sim \mathcal{D}_f$ and $x'_f \sim \mathcal{D}_f$. The objective of the critic function is to maximize the Cramer distance $\mathcal{L}_{\text{surrogate}}$ between samples from $\mathcal{D}_u$ and samples from $\mathcal{D}_f$:

$$D^* = \arg\min_D \left[-\mathcal{L}_{\text{surrogate}} + \lambda(\|\nabla_{\hat{z}} D(\hat{z})\|_2 - 1) \right]. \quad (7)$$

Here, $\lambda > 0$ is the regularization coefficient, $\hat{z} = \epsilon z_u + (1 - \epsilon) z_f$, for some $\epsilon \sim U(0, 1)$, and the surrogate loss has the form

$$\begin{aligned} &\mathcal{L}_{\text{surrogate}} \\ &= \|D(z_u) - D(z'_f)\|_2 + \|D(z_f)\|_2 - \|D(z_f) - D(z'_f)\|_2 - \|D(z_u)\|_2. \end{aligned} \quad (8)$$

Importantly, both densities ρ_f^G and ρ_u^G depend [32] on the generator G. Thus, the problem of mode collapse arises as it produces the same output for different inputs. To overcome this problem, we copy and freeze the initial model (denoted as $\hat{G}$) to use it to compute the density $\rho_u^{\hat{G}}$. We use the Conditional GAN (CGAN) training pipeline in our experimental setup.

It is important to keep the performance of the unlearned model G on the retain subset $\mathcal{D}_r$. This is done by introducing the Cross-Entropy (CE) term and Kullback-Leibler term (KL) into the generator objective:

$$\begin{aligned} G^* = \arg\min_G \max_D \big[&\text{CramerDistance}(D(\rho_f^G), D(\rho_u^{\hat{G}})) \\ &+ \alpha \text{CE}(G(x_r), y_r) + \gamma KL(\rho_r^G, \rho_r^{\hat{G}}) \big]. \end{aligned} \quad (9)$$

Here $\alpha > 0$ and $\gamma > 0$ are the parameters that control the trade-off between model performance on the retain set and the forget set, $x_r \sim \mathcal{D}_r$ is the sample from the retain set, and y_r – is the true label corresponding to the sample. An illustration of the pipeline is presented in Figure 1. The details of the training algorithm are presented in Algorithm 1.

Algorithm 1 The UnGAN Algorithm

Input: Initial model $\hat{G}$ trained on dataset $\mathcal{D}$, forget set $\mathcal{D}_f \subset \mathcal{D}$, retain set $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$, and unseen dataset $\mathcal{D}_u$, where $\mathcal{D}_u \cap \mathcal{D} = \emptyset$.
Output: Model G^* unlearned on forget set $\mathcal{D}_f$.
Initialize generator G aiming to approximate function $\hat{G}$ on $\mathcal{D}_r$ and alter its behavior on $\mathcal{D}_f$.
Initialize discriminator D, aiming to distinguish between G's outputs on $\mathcal{D}_f$ and $\hat{G}$'s outputs on $\mathcal{D}_u$.
for $n = 0...N_e$ **do**
  for $i = 0...T_d$ **do**
    $D \leftarrow$ DISCRIMINATOR-STEP$(D, G, \hat{G}, \mathcal{D}_f, \mathcal{D}_u)$
  end for
  for $i = 0...T_g$ **do**
    $G \leftarrow$ GENERATOR-STEP$(D, G, \mathcal{D}_f)$
  end for
  for $i = 0...T_f$ **do**
    $G \leftarrow$ FINE-TUNING-STEP$(D, G, \hat{G}, \mathcal{D}_r)$
  end for
end for

4. Experiments

To evaluate the proposed method, we conduct experiments on the CIFAR-10 image classification dataset [20]. In our experiments, we use ResNet18 [22] that is trained on the whole training dataset $\mathcal{D}$ as the initial model $\hat{G}$ and a 3-layer fully connected network as the discriminator model D.

4.1. Dataset Split. Complexity of the item removal machine unlearning task depends on the ratio $\kappa = \frac{|\mathcal{D}_f|}{|\mathcal{D}_r|}$ of sizes of the sets $\mathcal{D}_f$ and $\mathcal{D}_r$. Intuitively, the smaller the value of κ, the more reasonable it is to use machine unlearning, while for larger values of κ it becomes more reasonable to retrain the whole model. In our experimental setup, we fix $\kappa = 5/95$ i.e. the forget set $\mathcal{D}_f$ represents a random 5% of the initial training set $\mathcal{D}$.

4.2. Membership Inference Attack as an Evaluation Tool. To evaluate the proposed method, we perform the online membership inference attack (MIA) [6] on the unlearned model G to determine the membership statuses of samples from $\mathcal{D}_r$ and $\mathcal{D}_f$.

Algorithm 2 DISCRIMINATOR-STEP

Input: Discriminator D with weights ω_D, generator G, initial model $\hat{G}$, forget set $\mathcal{D}_f$, unseen set $\mathcal{D}_u$, learning rate μ
Output: Updated discriminator D
Sample $(x_u, y_u) \in \mathcal{D}_u$ – sample from unseen set
Sample $(x_f, y_f), (x'_f, y'_f) \in \mathcal{D}_f$ – two independent samples
Sample $\epsilon \sim$ Uniform(0, 1) a random number
$z_f \leftarrow (G(x_f) \mid y_f)$
$z'_f \leftarrow (G(x'_f) \mid y'_f)$
$z_u \leftarrow (\hat{G}(x_u) \mid y_u)$
$\hat{z} \leftarrow \epsilon z_u + (1 - \epsilon) z_f$
$f(z) = \|D(z) - D(z'_f)\|_2 - \|D(z)\|_2$
$L_{surrogate} = f(z_u) - f(z_f)$
$L_D = -L_{surrogate} + \lambda(\|\nabla_{\hat{z}} D(\hat{z})\|_2 - 1)$
$\omega_D \leftarrow \omega_D - \mu \nabla_{\omega_D} L_D$

Algorithm 3 GENERATOR-STEP

Input: Discriminator D, generator G with weights ω_G, initial model $\hat{G}$, forget set $\mathcal{D}_f$, unseen set $\mathcal{D}_u$, learning rate μ
Output: Updated generator G
Sample $(x_u, y_u) \in \mathcal{D}_u$ – sample from unseen set
Sample $(x_f, y_f), (x'_f, y'_f) \in \mathcal{D}_f$ – two independent samples
$z_f \leftarrow (G(x_f) \mid y_f)$
$z'_f \leftarrow (G(x'_f) \mid y'_f)$
$z_u \leftarrow (\hat{G}(x_u) \mid y_u)$
$L_G = \|D(z_u) - D(z_f)\|_2 + \|D(z_u) - D(z'_f)\|_2 - \|D(z_f) - D(z'_f)\|_2$
$\omega_G \leftarrow \omega_G - \mu \nabla_{\omega_G} L_G$

Namely, we create k shadow models $g_1, g_2, \ldots, g_k$ of the same architecture as G and train them on respective datasets $\mathcal{D}_i^s$.

To compute the dataset $\mathcal{D}_i^s$, we prepare a hold-out dataset $\mathcal{D}^s$ that does not overlap with $\mathcal{D}$ of size n and then sample n objects without repetitions from the set $\mathcal{D} \cup \mathcal{D}^s$. Following the evaluation protocol for the membership inference attack [5, 6], we ensure that every sample from $\mathcal{D} \cup \mathcal{D}^s$ is included in exactly $k/2$ sets $\{\mathcal{D}_1^s, \ldots, \mathcal{D}_k^s\}$.

Algorithm 4 FINE-TUNING-STEP

Input: Generator G with weights ω_G, initial model $\hat{G}$, retain set $\mathcal{D}_r$, the weight α of CE term, the weight γ of KL term, learning rate μ
Output: Updated generator G
Sample $(x_r, y_r) \in \mathcal{D}_r$ – sample from retain set
$L_G = \alpha CE(G(x_r), y_r) + \gamma KL(G(x_r), \hat{G}(x_r))$
$\omega_G \leftarrow \omega_G - \mu \nabla_{\omega_G} L_G$

Thus, given a sample (x, y) we form two sets of shadow models: the *in* shadow models $g_{i_1}, \ldots, g_{i_{k/2}}$ were trained with the inclusion of (x, y) to their training datasets; in contrast, the *out* shadow models $g_{i_{k/2+1}}, \ldots, g_{i_k}$ did not see (x, y) during training. In the next step, we compute the scores for the sample (x, y) in the form

$$\phi(p) = \log\left(\frac{p}{1-p}\right), \tag{10}$$

where $p = g_i(x)_y$ corresponds to the probability to assign the object x to class y by model g_i. After that, we fit two Gaussian distributions, namely $\mathcal{N}(\mu_{\text{in}}, \sigma^2_{\text{in}})$ and $\mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}})$ to approximate the densities of $\phi(p|in)$ and $\phi(p|out)$, respectively.

Finally, we compute the confidence of the membership inference algorithm to assign the sample (x, y) to the training set of the unlearned model G in the form

$$s(x, y) = \frac{p(\phi(p^*) \mid \mathcal{N}(\mu_{\text{in}}, \sigma^2_{\text{in}}))}{p(\phi(p^*) \mid \mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}}))}, \tag{11}$$

where, $p^* = G(x)_y$.

4.3. Experimental setup. In our experiments, the initial model $\hat{G}$ was trained on the whole training dataset $\mathcal{D}$ for 40 epochs with the use of SGD optimizer with the following parameters: learning rate of 0.1, momentum of 0.9 and cosine annealing learning rate scheduler.

The UnGAN is applied to the initial model $\hat{G}$ with the following parameters:

- number of epochs: $N_e = 8$;
- number of discriminator steps for each epoch: $T_d = 4$;
- number of generator steps for each epoch: $T_g = 4$;
- number of fine-tuning steps for each epoch: $T_f = 88$.

Table 1. Quantitative results of the method. We report the accuracy on the retain and test datasets to demonstrate the model's performance on the target task (image classification). The accuracy on the forget set is reported for the information - intuitively, as a result of unlearning the accuracy on forget and test sets must be the same. Also, we report the True Negative Rate at a fixed False Negative Rate of 1% of LiRA and the execution time of the methods in seconds, T_U.

	Retain Acc. ↑	Forget Acc.	Test Acc. ↑	LiRA ↑	T_U ↓
Original	89.05	89.16	75.21	1.07	**0.00**
Retrain	88.77	76.37	74.83	8.57	49.12
Fine-Tuning	89.46	88.97	75.27	1.26	12.00
Random Labels	87.27	75.83	73.56	3.70	15.00
Neg Grads	88.76	87.30	75.04	1.84	4.00
Adv. Neg Grads	86.70	74.02	72.72	6.40	4.02
UNSIR	85.35	82.12	73.82	1.98	13.78
SCRUB	**94.80**	74.81	73.21	9.63	15.12
CF	89.07	89.12	**75.28**	1.08	2.62
UnGAN (Ours)	93.52	77.62	74.25	**10.32**	13.06

The discriminator's weights are updated by SGD optimizer with a learning rate of 10^{-3} and weight decay 0.5×10^{-3}. The UnGAN has Adam optimizer for generator and fine-tuning steps with learning rate 0.5×10^{-3} and weight decay 0.5×10^{-4}. The hyper-parameters α and γ used in fine-tuning steps are $1/16$ and $1/32$, respectively. The batch size was set to be 256.

For the membership inference attack, we form the set of shadow models of the same architecture as the initial model $\hat{G}$ and train them in the same way as $\hat{G}$ is trained.

4.4. Evaluation of the Method. We evaluated our method by computing the accuracy on the retain and test datasets and evaluating its susceptibility to membership inference attacks. Following the protocol from [6], the susceptibility to the membership inference attack is evaluated by computing the true negative rate of the MIA on the low values of the false negative rate. The intuition behind using this metric is simple: all the objects from forget set $\mathcal{D}_f$ should be recognized by MIA as non-members of

the training set (although they are initially from the training set $\mathcal{D}$); at the same time, the False Negative Rate should be low, since the objects from retain set $\mathcal{D}_r$ to be recognized as members.

To assess the effectiveness of the proposed approach, we evaluate it against several baseline methods of unlearning.

- Original – the original model is assessed without the application of any unlearning techniques.
- Retrain – the model is fitted from scratch without the forget set $\mathcal{D}_f$. This is not applicable in practice but is reported here as a reference.
- Fine-tuned – the original model is fine-tuned on the retain set only.
- Fine-tuned with random labels – the original model is fine-tuned on the training dataset $\mathcal{D}_f \cup \mathcal{D}_r$ but with the random ground truth labels assigned to the samples from the forget set $\mathcal{D}_f$.
- Negative Gradients – the original model is fine-tuned on the dataset $\mathcal{D}_f$ to maximize the target loss on this set.
- Advanced Negative Gradients – the original model is fine-tuned on the dataset $\mathcal{D}_f \cup \mathcal{D}_r$ but with the flipped sign of the gradients computed for the samples from forget set $\mathcal{D}_f$.
- Catastrophic Forgetting-k (CF-k) – first k layers of the original model are frozen and the remaining ones are fine-tuned on the retain dataset $\mathcal{D}_r$
- UNSIR [33] – trainable noise matrix is used to induce sharp unlearning in the model on forget set $\mathcal{D}_f$ and repair overall performance on retain set $\mathcal{D}_r$.
- SCRUB [34] – the original model is trained on the forget set $\mathcal{D}_f$ to maximize the KL distance from the original and unlearned model's outputs. And on the retain set $\mathcal{D}_r$ to minimize both the distance between the original and unlearned model's outputs and target loss.

In Table 1, we report the quantitative results of our experiments. According to the LiRA metric that demonstrates the forgetting quality of the unlearning algorithms, UnGAN outperforms all the considered baselines. At the same time, the proposed approach does not suffer from *catastrophic unlearning* [15] like Adv. Neg Grads which drop the test accuracy by more then 2 %. It is noteworthy that our method retains high accuracy on $\mathcal{D}_r$ and $\mathcal{D}$ datasets. Finally, UnGAN is notably faster than the plain retraining of the model.

5. Conclusion

In this work, we have proposed a novel approach for machine unlearning that features a membership inference routine. Forgetting subsets of data with this approach can prevent leakage of information about specific observations through model queries and can thus resolve a variety of task-specific problems including protection against backdoor attacks and privacy leaks.

References

1. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need*, in: Advances in Neural Information Processing Systems, vol. 30 (2017).
2. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*, in: 9th International Conference on Learning Representations, ICLR (2021).
3. J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, and Others, *GPT-4 Technical Report*, ArXiv preprint arXiv:2303.08774 (2023).
4. J. Lin, R. Men, A. Yang, C. Zhou, M. Ding, Y. Zhang, P. Wang, A. Wang, L. Jiang, X. Jia, and Others, *M6: A Chinese Multimodal Pretrainer*, ArXiv preprint arXiv:2103.00823 (2021).
5. R. Shokri, M. Stronati, C. Song, and V. Shmatikov, *Membership inference attacks against machine learning models*, in: 2017 IEEE Symposium on Security and Privacy (SP) (2017), pp. 3–18.
6. N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, *Membership inference attacks from first principles*, in: 2022 IEEE Symposium on Security and Privacy (SP) (2022), pp. 1897–1914.
7. T. Nguyen, T. Huynh, P. Nguyen, A. Liew, H. Yin, and Q. Nguyen, *A survey of machine unlearning*, ArXiv preprint arXiv:2209.02299 (2022).
8. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, and Y. Bengio, *Generative adversarial networks*, Communications of the ACM, vol. 63, no. 11 (2020), pp. 139–144.
9. M. Arjovsky, S. Chintala, and L. Bottou, *Wasserstein generative adversarial networks*, in: International Conference on Machine Learning (2017), pp. 214–223.
10. A. Ginart, M. Guan, G. Valiant, and J. Zou, *Making AI Forget You: Data Deletion in Machine Learning*, in: Advances in Neural Information Processing Systems (2019), pp. 3513–3526.
11. C. Guo, T. Goldstein, A. Hannun, and L. Van Der Maaten, *Certified Data Removal from Machine Learning Models*, in: International Conference on Machine Learning (2020), pp. 3832–3842.
12. T. Baumhauer, P. Schöttle, and M. Zeppelzauer, *Machine Unlearning: Linear Filtration for Logit-Based Classifiers.* — Mach. Learn. **111** (2022), 3203–3226.

13. Z. Izzo, M. Smart, K. Chaudhuri, and J. Zou, *Approximate Data Deletion from Machine Learning Models*, in: International Conference on Artificial Intelligence and Statistics (2021), pp. 2008–2016.
14. S. Neel, A. Roth, and S. Sharifi-Malvajerdi, *Descent-to-Delete: Gradient-Based Methods for Machine Unlearning*, in: Algorithmic Learning Theory (2021), pp. 931–962.
15. Q. Nguyen, B. Low, and P. Jaillet, *Variational Bayesian Unlearning*, in: Advances in Neural Information Processing Systems, vol. 33 (2020).
16. Y. Wu, E. Dobriban, and S. Davidson, *DeltaGrad: Rapid Retraining of Machine Learning Models*, in: International Conference on Machine Learning (2020), pp. 10355–10366.
17. L. Bourtoule, V. Chandrasekaran, C. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, *Machine Unlearning*, in: 2021 IEEE Symposium on Security and Privacy (SP) (2021), pp. 141–159.
18. B. Mirzasoleiman, A. Karbasi, and A. Krause, *Deletion-Robust Submodular Maximization: Data Summarization with "The Right to Be Forgotten"*, in: International Conference on Machine Learning (2017), pp. 2449–2458.
19. Y. Cao and J. Yang, *Towards Making Systems Forget with Machine Unlearning*, in: 2015 IEEE Symposium on Security and Privacy (2015), pp. 463–480.
20. A. Krizhevsky, *Learning Multiple Layers of Features from Tiny Images*, Master's Thesis, University of Toronto, 2009.
21. J. Brophy and D. Lowd, *Machine Unlearning for Random Forests*, in: International Conference on Machine Learning (2021), pp. 1092–1104.
22. K. He, X. Zhang, S. Ren, and J. Sun, *Deep Residual Learning for Image Recognition*, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016), pp. 770–778.
23. V. Gupta, C. Jung, S. Neel, A. Roth, S. Sharifi-Malvajerdi, and C. Waites, *Adaptive Machine Unlearning*, in: Advances in Neural Information Processing Systems, vol. 34 (2021).
24. H. Hu, Z. Salcic, L. Sun, G. Dobbie, P. Yu, and X. Zhang, *Membership Inference Attacks on Machine Learning: A Survey*, ACM Comput. Surv. **54** (2022), pp. 1–37.
25. M. Bertran, S. Tang, A. Roth, M. Kearns, J. Morgenstern, and S. Wu, *Scalable Membership Inference Attacks via Quantile Regression*, in: Advances in Neural Information Processing Systems, vol. 36 (2024).
26. A. Sablayrolles, M. Douze, C. Schmid, Y. Ollivier, and H. J'egou, *White-Box vs Black-Box: Bayes Optimal Strategies for Membership Inference*, in: International Conference on Machine Learning (2019), pp. 5558–5567.
27. A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, *ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models*, ArXiv preprint arXiv:1806.01246 (2018).
28. L. Song and P. Mittal, *Systematic Evaluation of Privacy Risks of Machine Learning Models*, in: 30th USENIX Security Symposium (USENIX Security 21) (2021), pp. 2615–2632.

29. J. Neyman and E. Pearson, *IX. On the Problem of the Most Efficient Tests of Statistical Hypotheses*, in: Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character, vol. 231 (1933), pp. 289–337.
30. H. Thanh-Tung and T. Tran, *Catastrophic Forgetting and Mode Collapse in GANs*, in: 2020 International Joint Conference on Neural Networks (IJCNN) (2020), pp. 1–10.
31. M. Mirza and S. Osindero, *Conditional Generative Adversarial Nets*, ArXiv preprint arXiv:1411.1784 (2014).
32. A. Ben-Israel, *The Change-of-Variables Formula Using Matrix Volume*, SIAM J. Matrix Anal. Appl. **21** (1999).
33. A. Tarun, V. Chundawat, M. Mandal, and M. Kankanhalli, *Fast Yet Effective Machine Unlearning*, IEEE Trans. Neural Netw. Learn. Syst. (2024), pp. 1–10.
34. M. Kurmanji, P. Triantafillou, J. Hayes, and E. Triantafillou, *Towards Unbounded Machine Unlearning*, in: Advances in Neural Information Processing Systems, vol. 36 (2024).
35. M. Bellemare, I. Danihelka, W. Dabney, S. Mohamed, B. Lakshminarayanan, S. Hoyer, and R. Munos, *The Cramer Distance as a Solution to Biased Wasserstein Gradients*, ArXiv preprint arXiv:1705.10743 (2017).

Поступило 15 ноября 2024 г.

Moscow Institute
of Physics and Technology
(National Research University)
E-mail: zhavoronkin.ao@phystech.edu

Artificial Intelligence Research Institute;
Ivannikov Institute for System Programming
of the Russian Academy of Sciences
E-mail: Pautov@airi.net

Skolkovo Institute of Science and Technology
E-mail: nikolay.kalmykov@skoltech.ru

E-mail: egor.sevriugov@skoltech.ru

Moscow Institute of Physics and Technology
(National Research University); SaluteDevices
E-mail: dmitrii.kovalev@phystech.edu

Skolkovo Institute of Science and Technology;
Artificial Intelligence Research Institute; VeinCV LLC
E-mail: rogov@airi.net

Skolkovo Institute of Science and Technology;
Artificial Intelligence Research Institute
E-mail: i.oseledets@skoltech.ru