Re-thinking Data Availability Attacks Against Deep Neural Networks

Bin Fang[1,*] Bo Li[2,*,†] Shuang Wu[2] Shouhong Ding[2] Ran Yi[1] Lizhuang Ma[1,3,†]
[1] Shanghai Jiao Tong University, Shanghai [2] Youtu Lab Tencent, Shanghai
[3]MoE Key Lab of Artificial Intelligence, Shanghai Jiao Tong University
{fang-bin, ranyi, lzma}@sjtu.edu.cn njumagiclibo@gmail.com {calvinwu, ericshding}@tencent.com

Abstract

The unauthorized use of personal data for commercial purposes and the covert acquisition of private data for training machine learning models continue to raise concerns. To address these issues, researchers have proposed availability attacks that aim to render data unexploitable. However, many availability attack methods can be easily disrupted by adversarial training. Although some robust methods can resist adversarial training, their protective effects are limited. In this paper, we re-examine the existing availability attack methods and propose a novel two-stage min-max-min optimization paradigm to generate robust unlearnable noise. The inner min stage is utilized to generate unlearnable noise, while the outer min-max stage simulates the training process of the poisoned model. Additionally, we formulate the attack effects and use it to constrain the optimization objective. Comprehensive experiments have revealed that the noise generated by our method can lead to a decline in test accuracy for adversarially trained poisoned models by up to approximately 30%, *in comparison to SOTA methods.* [1]

1. Introduction

Over the last decade, remarkable advancements have been made in the field of Artificial Intelligence (AI), leading to significant impacts across a wide range of domains. The key driving force behind impressive achievements of deep learning has been access to vast quantities of high-quality data. In fact, many major AI breakthroughs have been realized only after obtaining the appropriate training data. The recent advances in large foundation models [4, 21] and generative models [22, 25] stand as strong evidence. Nonetheless, behind these remarkable accomplishments lies an issue that cannot be overlooked: the unauthorized collection and utilization of data. There is evidence to suggest that technology corporations are engaged in the collection and utilization of unauthorized data for the purpose of training their commercial models [2, 9, 33, 42, 43].

To mitigate the unauthorized use of data, availability attacks have been proposed [1]. Numerous studies demonstrate that injecting imperceptible noise, known as unlearnable noise, into data can considerably impair the performance of models reliant on such poisoned data [6, 7, 13, 27, 28, 35–37, 41]. The poisoned data is called *unlearnable examples*, which is firstly proposed by Error-Minimizing (EM) [13]. Nevertheless, the unlearnable noise produced by these approaches can be readily neutralized by adversarial training [7, 8, 13, 27, 32, 34], thereby undermining the protective efficacy with respect to the unlearnable examples. Researches have argued that EM [13] noise is inadequately equipped to defend against adversarial training due to its standard training of the surrogate model, which solely extracts non-robust features [14]. To address this limitation, Robust Error-Minimizing (REM) [8] and Entangled Features (EntF) [34] have been introduced to diminish the detrimental impact of adversarial training on unlearnable examples.

Although REM [8] and EntF [34] noise partially mitigate the detrimental effects of adversarial training on unlearnable examples, their theoretical underpinnings remain uncertain. Upon examining their optimization objectives, it becomes evident that the goal of REM [8] closely resembles that of EM [13], suggesting that the surrogate model employed in REM also lacks robustness. Furthermore, concerning EntF [34], previous research [10, 24] has demonstrated that constraining noise with only features makes it challenging to achieve conservation effects in classification tasks.

Building upon these analyses, we put forward two key proposals: **(1)** A surrogate model, trained from scratch utilizing adversarial training techniques, has the potential to generate more robust protective noise, thereby mitigating the adverse effects of adversarial training. **(2)** Furthermore, EM [13] provides a definition for the performance of poisoned models on clean examples, stating that *"DNNs trained on unlearnable examples will exhibit performance equivalent to random guessing on normal test examples"*. This definition has been overlooked in previous research. For the

*Equal contribution
†Corresponding author
[1]Code is available at EuterpeK/Rethinking-Data-Availability-Attacks

first time, we propose *Average Randomness Constraint* to formalize the intended effect of unlearnable noise based on its definition and adapt our optimization objective to encompass this understanding. As a result, our method attains remarkable protective performance in both standard and adversarial training. In summary, our contributions are as follows:

- We provide an overview of prior methods for availability attacks and assess the limitations of these strategies.
- We propose a reliable optimization objective (*min-max-min*) that efficiently mitigates the disruptive effects of adversarial training, offering robust data protection.
- For the first time, we propose *average randomness constraint* to formulate the expected effect of unlearnable examples and use this constraint to adjust our optimization objective, subsequently resulting in significant performance improvements.
- We establish a foundation for future research, allowing for the easy integration of additional constraints into our optimization objective, thus promoting further progress.

2. Related Work

2.1. Poisoning Attacks

Data poisoning attacks aim to compromise the training process of a model by introducing noise into the training dataset, resulting in significant testing errors on specific or unseen samples during the testing phase. Backdoor attacks constitute a prevalent form of data poisoning attack, often characterized by the injection of triggers into training samples, which subsequently provokes the misclassification of images containing these triggers during the testing phase [16, 17, 20]. However, it is worth noting that these attacks usually impact only samples with trigger patterns, while leaving clean samples unaffected and accurately classified [3, 29].

2.2. Availability Attacks

Availability Attacks aim to safeguard data from unauthorized exploitation by generating imperceptible, unlearnable noise. The data compromised by this type of attack are referred to as unlearnable examples. Deep neural networks trained on unlearnable examples display performance similar to random guessing on clean test examples.

Model-free Attacks. These attacks generally produce unlearnable noise at the pixel level instead of the feature level. As a result, methods in this category, such as LSP [36], AR [28], CUDA [27] and OPS [35], do not require any feature information on clean data, leading to an unrelated connection with data features. However, this inherent design principle makes unlearnable examples susceptible to feature-based defense methods, such as adversarial training [8, 18].

Model-based Attacks. These attacks typically generate unlearnable noise using surrogate models. This group of methods trains a surrogate model, also referred to as a noise generator. The training process of the surrogate model is used to mimic the training process of poisoned models. Based on whether the surrogate models employ adversarial training techniques, these methods can be classified into two distinct categories. EM [13] and REM [8] have substantiated that prevalent data augmentation techniques, such as Cutout [5], Mixup [40], and CutMix [38], do not compromise the protective efficacy of unlearnable noise.

I. Non-robust Model-based Attacks. This type of attack involves training surrogate models as non-robust models that learn non-robust features, such as TAP [7], NTGA [37], and EM [13]. As a result, the unlearnable noise generated by this approach only targets poisoned models subjected to standard training and merely prevents models from learning standard data features. Once the poisoned models undergo adversarial training, the protective effects are disrupted.

II. Robust Model-based Attacks. This type of attack entails training surrogate models as robust models that learn robust features, such as REM [8] and EntF [34]. Although these methods can withstand adversarial training, their protective effect remains limited.

3. Methodology

We provide the list of symbols used throughout the main manuscript. These symbols are summarized in Table 1.

Symbol	Description
$x_i \in \mathbb{R}^{[w \times h \times c]}$	The i-th example from a dataset $\mathcal{D}$.
$y_i \in \mathcal{Y} = \{1, \dots, K\}$	The class label associated with x_i for supervised learning (one-hot encoded).
$\mathcal{D} = \{(x_1, y_1), \dots, (x_N, y_N)\}$	Training set: N data-label pairs.
f'_θ	The surrogate model(noise generator).
$\delta_i^u \in [-\rho_u, \rho_u]$	The i-th unlearnable noise associated with x_i generated by the surrogate model f'_θ.
$x'_i = x_i + \delta_i^u$	The i-th unlearnable example.
$\delta_i^a \in [-\rho_a, \rho_a]$	The i-th adversarial noise associated with x'_i generated by the surrogate model f'_θ.
$\rho_u \in \mathbb{R}$	Unlearnable noise radius.
$\rho_a \in \mathbb{R}$	Adversarial noise radius.
ℓ or $\mathcal{L}$	Loss function.
$P \in \mathbb{R}^K$	Predicted probability vector
$R \in \mathbb{R}$	Averaged prediction randomness
$\mathcal{R} = \frac{1}{K}^K$	Random guessing probability
$\Theta \in \mathbb{R}$	Predicted probability

Table 1. The list of symbols used in this paper.

3.1. Limitations of Robust Model-based Methods

As we have discussed in Section 2.2, to resist the disruptive effects of adversarial training on unlearnable examples, currently, there is only one way to go, which is robust model-based availability attacks. However, REM [8] and EntF [34] have some flaws.

REM [8] posits that the protective effect of unlearnable examples is compromised in adversarial training because

	REM	Ours
Optimization Object	min-(min-max)	(min-max)-min
Surrogate Model	non-robust	robust
Randomness Constrain	w/o	w

Table 2. The main differences between REM and our method. Randomness Constraint is detailedly introduced in Section 3.3.

the model can learn knowledge from adversarial examples during the process. Based on this perspective, REM generates unlearnable noise for adversarial examples. REM proposes a **min-(min-max)** optimization procedure. The training objective of REM is as follows:

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \min_{\|\delta_i^u\| \le \rho_u} \max_{\|\delta_i^a\| \le \rho_a} \ell(f'_\theta(x_i + \delta_i^u + \delta_i^a), y_i). \quad (1)$$

The optimization objective in Equation 1 can be partitioned into two distinct steps. The initial step involves the inner **min-max** as depicted, which is employed to generate adversarial samples and produce the corresponding unlearnable noise. Subsequently, the second step consists of the outer min_θ, which functions to update the surrogate model.

Given that $\rho_a < \rho_u$, the inputs utilized for updating the surrogate model are essentially the same as those in EM [13]. Consequently, REM is fundamentally akin to EM, with both surrogate models exhibiting non-robust characteristics. Nonetheless, the poisoned model becomes robust following adversarial training, even when the training data comprises unlearnable instances. As such, we posit that a robust surrogate model should be employed to generate robust unlearnable noise. We propose a **min-max-min** optimization objective that significantly deviates from REM. Table 2 illustrates the primary distinctions. Section 3.2 offers a detailed overview of our optimization objective. An in-depth analysis and extensive experiments highlighting the insufficiency of REM objective can be found in Appendix B.

EntF [34] utilizes a pre-trained robust feature extractor, aiming to challenge the premise of adversarial training by making similar features more aggregated. However, according to previous studies [10, 24], the noise generated via feature constraints cannot protect classification tasks.

3.2. Two-Stage Optimization Procedure

In light of our analysis, we regard that a robust surrogate model is essential for generating unlearnable noise that can effectively protect unlearnable examples against adversarial training. We propose a two-stage **min-max-min** optimization process to train a robust surrogate model capable of generating robust unlearnable noise. The two stages have different targets. **The first stage** involves an inner minimization process, where unlearnable noise is obtained for a noise generator that undergoes adversarial training. Since adversarial training can extract robust features, the unlearnable noise generated by robust models can naturally resist adversarial training. **The second stage** consists of an external min-max optimization process equivalent to adversarial training. The input for this stage includes images with robust unlearnable noise added, allowing the external procedure to *simulate the adversarial training process and closely resemble the training process of a poisoned model using adversarial training*. **Consequently**, both the first and second stages complement each other, and the internal generation of unlearnable noise results in better protective effects. The two-stage min-max-min optimization process is specified below:

a) Generate unlearnable examples x' from the surrogate model f'_θ by

$$\delta_i^u = \min_{\|\delta_i^u\| \le \rho_u} \ell(f'_\theta(x_i + \delta_i^u), y_i). \quad (2)$$

b) Perform adversarial training of the surrogate model f'_θ to extract the robust features of the unlearnable examples

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \max_{\|\delta_i^a\| \le \rho_a} \ell(f'_\theta(x_i + \delta_i^u + \delta_i^a), y_i). \quad (3)$$

Additionally, we expect $\rho_a < \rho_u$. As suggested in REM [8], when $\rho_a \ge \rho_u$, the generated unlearnable noise δ_u could not suppress any learnable knowledge.

However, the optimization objective in Equation 3 does not reflect the constraint on the prediction performance for

---

Algorithm I: Training the noise generator

Input: Training data set $\mathcal{D}$, training iteration M,
classes K, learning rate η
PGD parameters ρ_u, α_u and K_u for stage 1,
PGD parameters ρ_a, α_a and K_a for stage 2.
Output: Our noise generator f'_θ.
Initialize source model parameter θ.
for i **in** $1, \cdots, M$ **do**
  Sample a minibatch $(x, y) \sim \mathcal{D}$.
  Initialize δ^u.
  for k **in** $1, \cdots, K_u$ **do** (stage1)
    $g_k \leftarrow \frac{\partial}{\partial \delta^u} \ell(f'_\theta(x + \delta^u), y)$
    $\delta^u \leftarrow \prod_{\|\delta\| \le \rho_u} (\delta^u - \alpha_u \cdot \mathrm{sign}(g_k))$
  end for
  for k **in** $1, \cdots, K_a$ **do** (stage2)
    $g_k \leftarrow \frac{\partial}{\partial \delta^u} \ell(f'_\theta(x + \delta^u + \delta^a), y)$
    $\delta^a \leftarrow \prod_{\|\delta\| \le \rho_a} (\delta^a + \alpha_a \cdot \mathrm{sign}(g_k))$
  end for
  $g_k \leftarrow \frac{\partial}{\partial \theta} [\ell(f'_\theta(x + \delta^u + \delta^a), y)$
    $+ \frac{1}{K} \sum_{k=1}^{K} (f'_\theta(x_i)[k] - \frac{1}{K})^2]$
  $\theta \leftarrow \theta - \eta \cdot g_k$
end for
return f'_θ

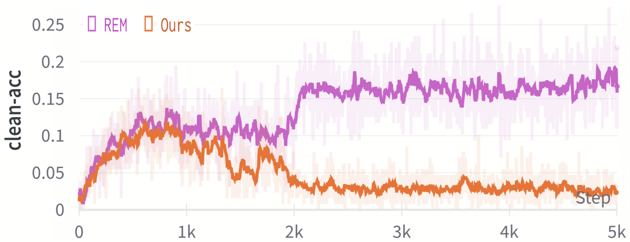

Figure 1. The test accuracy of the clean examples on the training phase of the surrogate model.

clean samples. Therefore, we need to add constraints to this optimization objective. The Algorithm I shows the two-stage optimization procedure with constraints on the performance for clean examples. Section 3.3 will introduce this constraint in detail.

3.3. Average Randomness Constraint

EM [13] posits that a model trained on unlearnable examples should exhibit random guessing behavior when applied to clean samples. Noting that the majority of prior studies did not constrain performance on clean samples. Additionally, as illustrated in Figure 1, the test accuracies of clean examples during noise generator training reveal that the noise produced by REM [8] still permits the surrogate models to learn a substantial amount of information. For the first time, we propose the *Average Randomness Constraint* to formulate the behavior of surrogate models on clean examples and utilize it to modify the optimization objective of Equation 3 in the second stage. Figure 1 showcases the effects of our method, with further details provided below.

Definition 1 (Average Prediction Randomness)**.** *Let $\mathcal{D}$ indicate a dataset consisting of N samples, where $x_i \in \mathcal{X}$ is the i-th sample and y_i is the corresponding label. Let a classifier denote $C : \mathcal{X} \rightarrow \mathcal{Y}$. Let P_k be the probability vector of model predictions on samples with ground-truth label k, where the j-th element of P_k is*

$$P_k^j \triangleq \frac{\sum_{i=1}^{N} \mathbb{I}\{C(x_i) = j\} \cdot \mathbb{I}\{y_i = k\}}{\sum_{i=1}^{N} \mathbb{I}\{y_i = k\}}. \tag{4}$$

The average prediction randomness metric R_p is defined as

$$R_p \triangleq \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} \mathbb{I}\{y_i = k\} \cdot \mathcal{L}\left(P_k\right), \tag{5}$$

where $\mathcal{L}(\cdot)$ denotes a distance function.

R_p measures the distance between the current predicted distribution and the uniform distribution. The smaller the value of R_p, the better dispersion. However, R_p is non-differentiable and cannot be optimized directly. We introduce a differentiable modified formulation to alleviate this problem, that is Definition 2.

Definition 2 (Differentiable Average Randomness)**.** *Let $\mathcal{D}$ represent a dataset consisting of N samples, where $x_i \in \mathcal{X}$ is the i-th sample and y_i is the corresponding label. Let a parameterized machine learning model be represented as f_θ. The averaged sample-wise randomness of predictions given by the classifier $f_\theta(\cdot)$ is defined as*

$$R_s \triangleq \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(f(x_i)\right). \tag{6}$$

The differentiable average randomness represents the average dispersion of predicted probability vectors.

Theorem 1. *Let $f_\theta(x_i)[k]$ indicates the k-th value of the predicted vector. Let $\mathcal{R} = (\frac{1}{K}, \ldots, \frac{1}{K}) \in R^K$ denotes the random guess probability. Then, we have*

$$0 \leq \frac{1}{K} \sum_{j=1}^{K} (f_\theta(x)[j] - \frac{1}{K})^2 < \frac{4}{K}.$$

Distance function $\mathcal{L}$ in Definition 2 assessing the distance between the predicted distribution and a uniform distribution is of paramount importance. There are three common types of loss functions: Mean Squared Error (MSE), Kullback-Leibler (KL) divergence, and Cross-Entropy (CE). We analyze these three loss functions in detail below.

(1) Previous study [23] suggests that models trained using a distance-based loss function frequently outperform those trained with non-distance loss functions. Cross-entropy and KL divergence are prevalent measures for calculating the distance between distributions; nevertheless, neither constitutes a distance function. While MSE is a distance function. (2) CE and KL are equivalent as demonstrated by Lemma in **Appendix A**. In most cases (under our experimental settings), the MSE is smaller than the CE loss. Additionally, both loss functions achieve their minimum values under the same conditions, suggesting that optimizing MSE is equivalent to optimizing CE. (3) MSE is smoother. Concerning KL loss, when the prediction probability for a specific class $f_\theta(\cdot)[k]$ is exceedingly small, both the loss and gradient become infinite, leading to gradient explosion and complicating the training process. In contrast, MSE showcases relative smoothness within its value range, possessing well-defined upper and lower bounds (given in Theorem 1), thereby easing model training.

Based on these insights, MSE serves as the distance function $\mathcal{L}$ in Definition 2 to assess Differentiable Average Randomness (DAR). A smaller DAR implies increased randomness in the output probability of f_θ on clean samples, indicating a reduced degree of knowledge acquisition by the model. The sample-wise DAR (i.e., our average randomness constraint) is then defined as follows:

Dataset	Adv. Train. ρ_a	Clean	EM	TAP	NTGA	REM $\rho_a = 4/255$	EntF $\rho_a = 4/255$	Ours $\rho_a = 4/255$
CIFAR-10	0	94.66	13.20	22.51	16.27	22.93	94.65	**12.71**
	1/255	93.74	22.08	92.16	41.53	30.00	93.56	**14.71**
	2/255	92.37	71.43	90.53	85.13	30.04	92.00	**15.38**
	3/255	90.90	87.71	89.55	89.41	31.75	91.04	**15.51**
	4/255	89.51	88.62	88.02	88.96	48.16	89.52	**23.12**
CIFAR-100	0	76.27	**1.60**	13.75	3.22	11.63	75.83	3.27
	1/255	71.90	71.47	70.03	65.74	14.48	71.88	**7.79**
	2/255	68.91	68.49	66.91	66.53	16.60	68.94	**7.73**
	3/255	66.45	65.66	64.30	64.80	20.70	66.43	**9.91**
	4/255	64.50	63.43	62.39	62.44	27.35	63.94	**23.00**
ImageNet Subset	0	80.66	**1.26**	9.10	8.42	13.74	78.96	4.08
	1/255	76.20	74.88	75.14	63.28	21.58	75.34	**11.80**
	2/255	72.52	71.74	70.56	66.96	29.40	72.10	**16.88**
	3/255	69.68	66.90	67.64	65.98	35.76	67.88	**22.34**
	4/255	66.62	63.40	63.56	63.06	41.66	63.60	**31.64**

Table 3. Test accuracy (%) of models adversarially trained with different perturbation radii. The training data, namely unlearnable examples, is generated by different availability attacks.

$$\frac{1}{K}\sum_{k=1}^{K}\left(f'_\theta(x_i)[k] - \frac{1}{K}\right)^2 \tag{7}$$

In summary, we use Equation 7 to modify Equation 3 in the second step. The final optimization objective in the second step is given by Equation 8. By adding the constraint of Equation 7, the model will learn less knowledge. Figure 1 also demonstrates the effectiveness of this constraint item.

$$\begin{aligned}\min_\theta \frac{1}{n}\sum_{i=1}^{n}\Bigg[&\max_{||\delta_i^a||\le\rho_a} \ell(f'_\theta(x_i+\delta_i^u+\delta_i^a), y_i)\\ &+\frac{1}{K}\sum_{k=1}^{K}\left(f'_\theta(x_i)[k]-\frac{1}{K}\right)^2\Bigg].\end{aligned} \tag{8}$$

4. Experiments

In this section, we have conducted extensive experiments to showcase the effectiveness and generalization ability of our method from various perspectives. Detailed settings can be found in **Appendix C**.

4.1. Experiment Setup

Datasets. To verify the effectiveness of our method on images of **varying categories and resolutions**, we use three commonly employed datasets in our experiments: CIFAR-10, CIFAR-100 [15], and ImageNet subset [26] (consists of the first 100 classes). The data augmentation technique [30] is applied in each experiment.

Surrogate Models. Following EM [13] and REM [8], we use ResNet-18 [11] as the surrogate model f'_θ for training our noise generator with Eq. (2) and Eq. (8). The L_∞-bounded noises $\|\delta_u\|_\infty \le \rho_u$ are adopted in our experiments. In all training phrases of surrogate models, the value of ρ_u is set to $8/255$, and the value of ρ_a is set to $4/255$. Furthermore, we also employ other surrogate models, including VGG-16[31], ResNet-50[11], and DenseNet-121[12], to test the generalization ability of our method.

Compared Methods. Our proposed method is compared with other state-of-the-art availability attacks, **TAP** [7], **NTGA** [37], **EM** [13], **REM** [8], and **EntF** [34].

Noise Test. Noise generated by our method is tested on both standard training and adversarial training [19]. We focus on L_∞-bounded noise $\|\delta_a\|_\infty \le \rho_a$ in adversarial training. In all training phrases of poisoned models, the adversarial training radius ρ_a is set $4/255$ unless otherwise specified. We conduct adversarial training on unlearnable examples created by our method using different poisoned models, including VGG-16 [31], ResNet-18, ResNet-50 [11], DenseNet-121 [12], and wide ResNet-34-10 [39]. It is important to note that when ρ_a is set to 0, adversarial training degenerates to standard training.

Metric. We evaluate the data protection ability of unlearnable noise by measuring the test accuracy of the model trained on unlearnable examples. Low test accuracy indicates that the model has learned little from the unlearnable examples, suggesting strong protection ability.

4.2. Effectiveness on standard and adversarial training

To evaluate the robustness against adversarial training, we first introduce unlearnable noise to the entire training set, generating unlearnable CIFAR-10 [15], CIFAR-100 [15],

Dataset	Model	Clean	EM	TAP	NTGA	REM	EntF	Ours
						$\rho_a = 4/255$	$\rho_a = 4/255$	$\rho_a = 4/255$
CIFAR-10	VGG-16	87.51	87.24	86.27	86.65	65.23	87.71	**37.78**
	RN-18	89.51	88.62	88.02	88.96	48.16	89.52	**23.12**
	RN-50	89.79	89.66	88.45	88.79	40.65	89.92	**19.30**
	DN-121	83.27	81.77	81.72	80.73	82.38	83.52	**72.42**
	WRN-34-10	91.21	79.87	90.23	89.95	48.39	91.30	**18.67**
CIFAR-100	VGG-16	57.46	56.94	55.24	55.81	58.07	57.86	**55.05**
	RN-18	64.50	63.43	62.39	62.44	27.35	63.94	**23.00**
	RN-50	66.93	66.43	64.44	64.91	26.03	66.46	**21.47**
	DN-121	53.73	53.52	52.93	52.40	56.63	53.89	**52.25**
	WRN-34-10	68.64	68.27	65.80	67.41	27.71	69.42	**20.14**

Table 4. Test accuracy (%) of different models adversarially trained on unlearnable CIFAR-10 and CIFAR-100 datasets.

and ImageNet-subset [26]. The unlearnable noise perturbation radius, denoted as ρ_u, is set to $8/255$ for all noise-generating methods and the adversarial perturbation radius ρ_a is set as $4/255$ for REM [8], EntF [34] and our method. We then train models using different adversarial training perturbation radii ρ_a. Table 3 presents the accuracies of the adversarially trained models on the unlearnable examples generated by different availability attacks.

As shown in Table 3, the adversarial training perturbation ρ_a ranges from $1/255$ to $4/255$, with ResNet-18 [11] as the surrogate models. For adversarial training, we observe that even a small adversarial training perturbation radius of $2/255$ can damage the protecting effects of TAP [7], NTGA [37], EM[13], and EntF [34]. Table 3 demonstrates that when the unlearnable noise perturbation radius is fixed, the protective effect decreases as the adversarial training perturbation radius increases. This finding suggests that to protect data against adversarial training with a perturbation radius ρ_a, one must set the unlearnable perturbation radius ρ_u of robust methods to a value relatively larger than ρ_a. Notably, our method consistently outperforms other approaches regardless of the adversarial perturbation radii. In standard training scenarios, our method also exhibits superior performance compared to other methods.

Furthermore, our method retains significant protective effects across different datasets, irrespective of their resolution or class composition, particularly when subjected to adversarial training. Overall, these experiments indicate that our method effectively safeguards data across various datasets and adversarial training perturbation radii.

4.3. Transferability on different poisoned model architectures

Thus far, we have conducted adversarial training exclusively with ResNet-18 [11], which is the same as the source model used in unlearnable noise generation. We now evaluate the effectiveness of the unlearnable noise generated by our method under various poisoned models. Specifically, we perform adversarial training with a perturbation radius of $4/255$ and five different models, including VGG-16 [31], ResNet-18 [11], ResNet-50 [11], DenseNet-121 [12], and Wide ResNet-34-10 [39], on data protected by noise generated via ResNet-18. We set the unlearnable perturbation radius ρ_u for each type of unlearnable noise at $8/255$. Table 4 presents the test accuracies of the trained models on CIFAR-10 and CIFAR-100 [15]. The results in Table 4 reveal that our unlearnable noise, generated from ResNet-18, can effectively protect data against various adversarially trained models, surpassing the performance of other methods.

4.4. Transferability on different surrogate models

So far, our method has used ResNet-18 as a surrogate model to generate noise. ResNet-18's specific properties may contribute to our method's effectiveness. To evaluate the generalization performance of our approach, we employ different surrogate models to generate unlearnable noise. Additionally, since our ASR is not tied to any specific model architecture, it should remain effective regardless of the surrogate model used. The adversarial training perturbation radius is set to $4/255$, and the unlearnable perturbation radius ρ_u for each type of unlearnable noise is set at $8/255$. We test four noise generators with different architectures, including VGG-16 [31], ResNet-18 [11], ResNet-50 [11], and DenseNet-121 [12]. Each type of noise is tested on five different models: VGG-16, ResNet-18, ResNet-50, DenseNet-121, and WRN-34-10 [39]. Test accuracies of poisoned models on unlearnable CIFAR-10 and CIFAR-100 [15] are presented in Table 5.

As depicted in Table 5, our method demonstrates exceptional generalizability. Regardless of the surrogate model's capabilities, our method consistently outperforms other availability attacks, both robust and non-robust. In detail, using different noise generators, the average accuracy of our method on CIFAR-10 is reduced by approximately 7% to

Datasets	Surrogate Model	Method	VGG-16	ResNet-18	ResNet-50	DenseNet-121	WRN-34-10	Average
CIFAR-10	VGG-16	EM	87.75	89.21	90.19	83.58	90.83	88.31
		REM	73.60	74.73	74.16	77.63	74.94	75.01
		Ours	**63.11**	**67.50**	**64.37**	**61.02**	**65.65**	**64.33**
	ResNet-18	EM	87.24	88.62	89.66	81.77	79.87	85.43
		REM	65.23	48.16	40.65	82.38	48.39	58.96
		Ours	**37.78**	**23.12**	**19.30**	**72.42**	**18.67**	**34.26**
	ResNet-50	EM	87.57	89.17	89.83	82.64	90.68	87.98
		REM	51.88	44.27	37.79	82.01	42.09	51.61
		Ours	**49.33**	**39.95**	**36.50**	**79.69**	**41.57**	**49.41**
	DenseNet-121	EM	87.59	84.51	85.57	82.76	85.68	85.22
		REM	67.30	69.62	66.42	60.51	72.09	67.19
		Ours	**61.41**	**58.77**	**58.55**	**58.66**	**63.38**	**60.15**
CIFAR-100	VGG-16	EM	57.33	63.55	65.44	53.45	68.23	61.60
		REM	41.13	52.00	51.77	48.92	56.05	49.97
		Ours	**36.67**	**45.82**	**46.45**	**45.52**	**48.59**	**44.61**
	ResNet-18	EM	56.94	63.43	66.43	53.52	68.27	61.72
		REM	58.07	27.35	26.03	56.63	27.71	39.16
		Ours	**55.05**	**23.00**	**21.47**	**52.25**	**20.14**	**34.38**
	ResNet-50	EM	56.82	64.19	66.93	54.51	68.56	62.20
		REM	54.61	35.50	30.43	54.26	35.11	41.98
		Ours	**52.57**	**26.17**	**29.38**	**52.19**	**25.91**	**37.24**
	DenseNet-121	EM	57.39	63.73	66.37	54.62	68.43	62.11
		REM	47.22	41.89	45.49	41.15	50.66	45.28
		Ours	**38.15**	**34.70**	**34.46**	**37.84**	**32.30**	**35.49**

Table 5. Test accuracy (%) of different models adversarially trained on CIFAR-10 and CIFAR-100 generated by different noise generators.

25% compared to state-of-the-art (SOTA) methods, and the average accuracy on CIFAR-100 is reduced by roughly 4% to 10% compared to SOTA methods. These results indicate that our approach possesses superior generalizability, and we have successfully proposed a generalizable method rather than a specific noise.

4.5. Protective effects on different protection percentages

In a more realistic and challenging scenario, only a portion of the data is protected, while the rest remains clean.

Specifically, we randomly select a subset of the training data from the entire set and introduce unlearnable noise to it. Subsequently, we conduct adversarial training with ResNet-18 on the combined noisy and clean data. The unlearnable perturbation radius for each noise is set to $8/255$, while the adversarial perturbation radius ρ_a of REM [8], Entf [34], and our method is set to $4/255$. The difference between the test accuracies on mixed data and clean data reflects the knowledge gained from the protected training data. The accuracies on clean test data are reported in Table 6.

Table 6 illustrates that as the percentage of data protection decreases, the performance of the trained model improves, indicating that the model can still learn from clean data. Furthermore, Table 6 reveals that in all cases, our method offers superior data protection compared to other approaches. This observation confirms that the unlearnable noise generated by our method is more effective, even when combined with clean data. Additionally, it suggests that our methods are capable of concealing more information.

When the protection ratio is relatively low, the protective effects of all methods are not particularly pronounced, which may be associated with the composition of the dataset. For example, in the CIFAR-10 [15] training set, each category contains 5,000 samples. Even if unlearnable noise is introduced to a small portion of the data, a significant amount of clean data remains. The remained clean data is sufficient for the model to acquire ample knowledge, so the addition of a small number of unlearnable examples does not lead to substantial accuracy changes. However, when the proportion of unlearnable examples increases, our method exhibits a noticeable performance improvement.

5. Conclusion and Discussion

Conclusion. In this paper, we have systematically reviewed existing availability attacks that aim to safeguard data from unauthorized usage by generating unlearnable noise and analyzed their limitations. Drawing from prior research and our experiments, we argue that a robust surrogate model, trained from scratch, is crucial for generating robust unlearnable noise capable of withstanding the detrimental effects of adversarial training. Moreover, we recognize that the current

Dataset	Adv. Train. ρ_a	Noise Type	Data Protection Percentage 0%	20% Mixed	20% Clean	40% Mixed	40% Clean	60% Mixed	60% Clean	80% Mixed	80% Clean	100%
CIFAR-10	2/255	EM	92.37	92.33	91.30	92.18	90.31	92.00	88.65	92.06	83.37	71.43
		TAP		92.17		91.62		91.32		91.48		90.53
		NTGA		92.41		92.19		92.23		91.74		85.13
		REM		92.23		90.79		88.85		83.70		30.04
		EntF		92.14		91.85		91.02		90.54		92.00
		Ours		**92.03**		**90.34**		**87.98**		**83.32**		**15.38**
	4/255	EM	89.51	89.39	88.17	89.09	86.76	89.41	85.07	89.41	79.41	88.62
		TAP		89.01		88.66		88.40		88.04		88.02
		NTGA		89.56		89.35		89.22		89.17		88.96
		REM		89.71		89.89		89.63		87.17		48.16
		EntF		89.98		88.59		88.56		88.53		89.52
		Ours		**88.79**		**88.36**		**88.25**		**84.84**		**23.12**
CIFAR-100	2/255	EM	68.91	68.68	66.54	68.80	64.21	68.28	58.35	68.70	47.99	68.49
		TAP		68.40		67.93		67.25		67.09		66.91
		NTGA		68.52		68.82		68.36		68.71		66.53
		REM		68.90		68.29		61.42		51.99		16.60
		EntF		69.38		66.93		65.80		66.92		68.94
		Ours		**68.39**		**65.60**		**60.74**		**49.97**		**7.73**
	4/255	EM	64.50	64.65	61.73	63.82	57.61	64.19	53.86	64.32	44.79	63.43
		TAP		64.36		63.35		62.58		63.15		62.39
		NTGA		63.48		63.59		63.64		62.83		62.44
		REM		64.27		64.67		64.99		63.14		27.35
		EntF		64.76		64.06		62.86		61.68		63.94
		Ours		**63.46**		**63.24**		**61.24**		**58.91**		**23.00**

Table 6. Test accuracy (%) on CIFAR-10 and CIFAR-100 with different protection percentages.

optimization process of robust model-based availability attacks is suboptimal, leading to the potential invalidation of their protective effects during adversarial training. To tackle these challenges, we introduce a two-stage **(min-max)-min** optimization procedure for training robust surrogate models from scratch. The inner **min** step employs a robust surrogate model to generate robust unlearnable noise for clean examples, while the outer **min-max** step simulates the adversarial training process of the poisoned model to enhance its robustness, using unlearnable examples as input. Additionally, we propose Differentiable Average Randomness (DAR) to formally define the protective effect of unlearnable examples and constrain the optimization objective during the surrogate model's training phase. Through extensive experiments, we showcase the superior protective performance of our approach, laying a solid foundation for future research.

Limitations and future works. The method proposed in this paper necessitates the incorporation of an adversarial training process to generate robust unlearnable examples, resulting in substantial computational costs when applied to large-scale datasets, such as ImageNet. In our future work, we aim to explore efficient robust methods to accelerate our approach. Furthermore, the current method has not been optimized for situations involving partial protection of data. When unlearnable noise is added to only a fraction of the data, the anti-learning effect is considerably weaker compared to scenarios where protective noise is introduced to the entire dataset. This gap presents a valuable avenue for future research, as it is crucial to develop techniques that can effectively safeguard data privacy even when only a subset of the data is targeted for protection. In future work, we may consider incorporating misleading erroneous high-level semantic information into unlearnable examples, ensuring that any knowledge acquired by the model consists of incorrect information.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62302297, No. 72192821), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102), Shanghai Science and Technology Commission (21511101200), Shanghai Sailing Program (22YF1420300), Young Elite Scientists Sponsorship Program by CAST (2022QNRC001) and YuCaiKe [2023] Project Number: 14105167-2023.

References

[1] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012. 1

[2] Blake Brittain. Ai companies ask u.s. court to dismiss artists' copyright lawsuit. [Online]. site: https://www.reuters.com/legal/ai-companies-ask-us-court-dismiss-artists-copyright-lawsuit-2023-04-19/, 2023. 1

[3] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *CoRR*, abs/1712.05526, 2017. 2

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. 1

[5] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017. 2

[6] Liam Fowl, Ping-yeh Chiang, Micah Goldblum, Jonas Geiping, Arpit Bansal, Wojtek Czaja, and Tom Goldstein. Preventing unauthorized use of proprietary data: Poisoning for secure dataset release. *CoRR*, abs/2103.02683, 2021. 1

[7] Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. Adversarial examples make strong poisons. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 30339–30351, 2021. 1, 2, 5, 6, 3

[8] Shaopeng Fu, Fengxiang He, Yang Liu, Li Shen, and Dacheng Tao. Robust unlearnable examples: Protecting data privacy against adversarial learning. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 1, 2, 3, 4, 5, 6, 7

[9] Matt Growcoot. Midjourney founder admits to using a 'hundred million' images without consent. [Online]. site: https://petapixel.com/2022/12/21/midjourny-founder-admits-to-using-a-hundred-million-images-without-consent/, 2022. 1

[10] Hao He, Kaiwen Zha, and Dina Katabi. Indiscriminate poisoning attacks on unsupervised contrastive learning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. 1, 3

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 5, 6, 3

[12] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016. 5, 6

[13] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 1, 2, 3, 4, 5, 6

[14] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 125–136, 2019. 1

[15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5, 6, 7, 3

[16] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 2

[17] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. Reflection backdoor: A natural backdoor attack on deep neural networks. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part X*, pages 182–199. Springer, 2020. 2

[18] Zhuoran Liu, Zhengyu Zhao, and Martha A. Larson. Image shortcut squeezing: Countering perturbative availability poisons with compression. *CoRR*, abs/2301.13838, 2023. 2

[19] 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. 5, 3

[20] Tuan Anh Nguyen and Anh Tuan Tran. Input-aware dynamic backdoor attack. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 2

[21] OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023. 1

[22] Jonas Oppenlaender. The creativity of text-to-image generation. In *25th International Academic Mindtrek conference, Academic Mindtrek 2022, Tampere, Finland, November 16-18, 2022*, pages 192–202. ACM, 2022. 1

[23] Tianyu Pang, Min Lin, Xiao Yang, Jun Zhu, and Shuicheng Yan. Robustness and accuracy could be reconcilable by (proper) definition. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, pages 17258–17277. PMLR, 2022. 4

[24] Jie Ren, Han Xu, Yuxuan Wan, Xingjun Ma, Lichao Sun, and Jiliang Tang. Transferable unlearnable examples. In *The Eleventh International Conference on Learning Representa-*

tions, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net, 2023. 1, 3

[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE, 2022. 1

[26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015. 5, 6

[27] Vinu Sankar Sadasivan, Mahdi Soltanolkotabi, and Soheil Feizi. CUDA: convolution-based unlearnable datasets. *CoRR*, abs/2303.04278, 2023. 1, 2

[28] Pedro Sandoval Segura, Vasu Singla, Jonas Geiping, Micah Goldblum, Tom Goldstein, and David Jacobs. Autoregressive perturbations for data poisoning. In *NeurIPS*, 2022. 1, 2

[29] 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. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6106–6116, 2018. 2

[30] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *J. Big Data*, 6: 60, 2019. 5

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5, 6

[32] Lue Tao, Lei Feng, Jinfeng Yi, Sheng-Jun Huang, and Songcan Chen. Better safe than sorry: Preventing delusive adversaries with adversarial training. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 16209–16225, 2021. 1

[33] Jason Vermes. Why regulators in canada and italy are digging into chatgpt's use of personal information. [Online]. site: https://www.cbc.ca/news/world/openai-chatgpt-data-privacy-investigations-1.6804205, 2023. 1

[34] Rui Wen, Zhengyu Zhao, Zhuoran Liu, Michael Backes, Tianhao Wang, and Yang Zhang. Is adversarial training really a silver bullet for mitigating data poisoning? In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. 1, 2, 3, 5, 6, 7

[35] Shutong Wu, Sizhe Chen, Cihang Xie, and Xiaolin Huang. One-pixel shortcut: On the learning preference of deep neural networks. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. 1, 2

[36] Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. Availability attacks create shortcuts. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 2367–2376. ACM, 2022. 2

[37] Chia-Hung Yuan and Shan-Hung Wu. Neural tangent generalization attacks. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, pages 12230–12240. PMLR, 2021. 1, 2, 5, 6, 3

[38] Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 6022–6031. IEEE, 2019. 2

[39] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*. BMVA Press, 2016. 5, 6

[40] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 2

[41] Jiaming Zhang, Xingjun Ma, Qi Yi, Jitao Sang, Yugang Jiang, Yaowei Wang, and Changsheng Xu. Unlearnable clusters: Towards label-agnostic unlearnable examples. *CoRR*, abs/2301.01217, 2023. 1

[42] Tianyi Zheng, Bo Li, Shuang Wu, Ben Wan, Guodong Mu, Shice Liu, Shouhong Ding, and Jia Wang. Mfae: Masked frequency autoencoders for domain generalization face anti-spoofing. *IEEE Transactions on Information Forensics and Security*, pages 1–1, 2024. 1

[43] Tianyi Zheng, Qinji Yu, Zhaoyu Chen, and Jia Wang. Famim: A novel frequency-domain augmentation masked image model framework for domain generalizable face anti-spoofing. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4470–4474, 2024. 1