ALCHEMY: Data-Free Adversarial Training

Yijie Bai
baiyj@zju.edu.cn
Zhejiang University
Hangzhou, Zhejiang, China

Zhongming Ma
allen191819@whu.edu.cn
Zhejiang University
Hangzhou, Zhejiang, China

Yanjiao Chen
chenyanjiao@zju.edu.cn
Zhejiang University
Hangzhou, Zhejiang, China

Jiangyi Deng
jydeng@zju.edu.cn
Zhejiang University
Hangzhou, Zhejiang, China

Shengyuan Pang
pangpang0093@zju.edu.cn
Zhejiang University
Hangzhou, Zhejiang, China

Yan Liu
bencao.ly@antgroup.com
Ant Group
Hangzhou, Zhejiang, China

Wenyuan Xu
wyxu@zju.edu.cn
Zhejiang University
Hangzhou, Zhejiang, China

ABSTRACT

Machine learning models have become integral to various aspects of daily life, prompting increased vulnerability to adversarial attacks. Adversarial training is one of the most promising and practical methods to enhance model robustness. Existing adversarial training methods, however, assume access to the original training data. But nowadays, more and more users directly download models from the open-source model platforms or tech companies, but the original training datasets are usually unreleased because of commercial interests or privacy. In such scenarios, the user cannot utilize the former adversarial training methods to improve model robustness because of the lack of original training datasets.

Thus, we present the first exploration of a data-free adversarial training framework, ALCHEMY, which seeks to enhance model robustness without requiring access to the original training data. By addressing the notable challenges of reconstructing high-quality training data with robust features and improving the adversarial robustness to the inaccessible original dataset, our approach achieves the goals of both high accuracy maintenance and robustness improvement. Comprehensive experiments on four datasets compared with five baselines, demonstrate ALCHEMY 's high effectiveness. With no access to any training dataset, the average robustness improvement with ALCHEMY is effective in most attack scenarios. Additional evaluations underscore the framework's stability under different settings and discuss future research directions.

CCS CONCEPTS

• **Security and privacy** → *Software and application security*; • **Computing methodologies** → **Artificial intelligence**.

CCS '24, October 14–18, 2024, Salt Lake City, UT, USA

ACM ISBN 979-8-4007-0636-3/24/10
https://doi.org/10.1145/3658644.3670395

KEYWORDS

Data-free, Dataset reconstruction, Adversarial training, Robustness transferability

ACM Reference Format:
Yijie Bai, Zhongming Ma, Yanjiao Chen, Jiangyi Deng, Shengyuan Pang, Yan Liu, and Wenyuan Xu. 2024. ALCHEMY: Data-Free Adversarial Training. In *Proceedings of Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS '24).* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3658644.3670395

1 INTRODUCTION

In recent years, machine learning models have been widely integrated into various aspects of people's lives, providing significant assistance in essential tasks such as facial recognition [27, 54], speech processing [60], autonomous driving [15], natural language understanding [30], and more. These applications hold considerable economic and informational value, therefore attracting the attention of potential attackers [65, 99]. Among these, adversarial example attacks represent one of the most commonly employed attack approaches, wherein attackers introduce imperceptible noise to input data, causing machine learning models to generate erroneous outputs [67, 99]. For instance, in autonomous driving, adversaries can induce recognition errors in self-driving vehicles and consequently cause traffic accidents by introducing minor perturbations to road signs [80]. In order to safeguard machine learning models from adversarial attacks, numerous defense methods against adversarial attacks have been developed. Adversarial training is currently considered one of the most feasible means to enhance model robustness [46, 69], thereby achieving model security.

So far, based on our investigation, nearly all adversarial training work assumes that the defender has access to the model's original training data. However, as elaborated in Figure 1, in some crucial real-world scenarios, participants can only obtain pre-trained models which are fragile against adversarial example attacks, but cannot access the training dataset for further robustness improvement due to data privacy or intellectual property. For example, increasing numbers of individuals tend to acquire and implement the pre-trained models from open-source model platforms [47, 53], but the

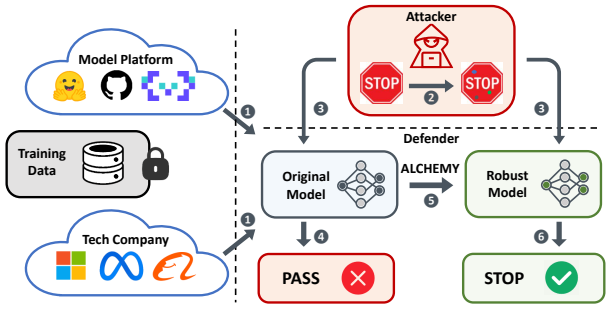


Figure 1: Attack Scenario of Alchemy. The user (1) downloads the open-source model from the model platforms or technology companies, but the training data is inaccessible. The attacker (2) leverages imperceptible perturbations on the input data and (3) tries to plague the user. The downloaded original model (4) is easy to be deceived to make the wrong prediction, causing dangerous troubles. So it's urgent for Alchemy to (5) perform adversarial training on the original model without access to the training dataset and (6) build a robust model to make the right prediction.

training dataset is inaccessible. Most of the models on these platforms have not undergone adversarial training. Taking Hugging Face[1] as an example, as of now, out of approximately 480,000 models available on Hugging Face, only around 350 models claim to be robust or have undergone adversarial training, less than 1‰. At the same time, many technology companies also choose to open-source their models, but they do not open-source their training data for reasons such as commercial interests. Also, most of these models have not undergone adversarial training properly [29, 32, 102]. In these contexts, users directly download models trained on private datasets but cannot access the original training data. Lacking the original training dataset, they cannot rely on previously proposed adversarial training methods to enhance model robustness, which poses significant risks to their use.

The majority of adversarial training methods assume that the model trainer has access to all the training data. Data augmentation based work also requires the trainer to have a portion of the training data to realize augmentation [64, 72, 92]. Although there have been semi-supervised methods and unsupervised methods based on unlabeled data proposed to improve adversarial robustness [2, 13], the semi-supervised methods aim to leverage the unlabelled dataset to supplement the original dataset to further improve adversarial robustness [13, 92] and the unsupervised methods use the unlabelled original dataset to acquire robustness [2]. These methods are all built upon having the labeled or unlabelled original dataset; thus, they can only be used with original dataset samples. Therefore, it is unprecedented and crucial to establish a method to improve model robustness in a totally data-free scenario.

Hence, we propose the first holistic data-free adversarial training framework, dubbed Alchemy. With this approach, users can enhance the robustness of models obtained from open-source communities or technology companies' websites, without needing any information about the original training data. Building data-free adversarial training requires us to address two significant difficulties to achieve high robustness against adversarial attacks while maintaining high accuracy on the original dataset. (1) With no access to the original dataset, the defender has to reconstruct the training data for the adversarial training. The adversarial training task requires the reconstructed data to contain enough robust features and demands high reconstruction quality. To address this problem, we propose the robust sample generation module to generate the substitute dataset with high similarity to the original training dataset and high inner diversity. A novel sample generation method considering the adversarial margin is designed to maintain the robust features for the downstream adversarial training. (2) Given the distribution difference between the generated data and the original training dataset, the adversarial robustness of the generated dataset faces a natural degrading problem when transferred to the original dataset. We propose a new adversarial training framework to address this challenge for better adversarial robustness on the original dataset. We integrate an effective adversarial example construction method into the framework. In the adversarial training module, we utilize the smoothness loss function and weight perturbation to improve the model robustness on both the generated dataset and the unseen original dataset.

We conduct comprehensive experiments to evaluate the performance of Alchemy on four datasets (MNIST [24], CIFAR-10 [50], SVHN [63], CINIC-10 [21]), compared with five baselines. With no access to any training dataset, the average robustness improvement between Alchemy and the original downloaded models is more than 65% in the white-box PGD-20 attack scenario on average. We also evaluate the method effect with different model structures and parameter settings, demonstrating the stability of Alchemy while providing insights from our observations. We further discuss the generator structure selection and combination setting, suggesting more viable research directions in future data-free adversarial training work.

The main contributions are summarized as follows:

- We propose a data-free adversarial training framework, named Alchemy, which realizes effective adversarial robustness improvement on the pre-trained model despite the lack of access to the original training dataset.
- We develop a novel training dataset generation algorithm that can maintain the reconstructed robust features and dataset diversity. We design a combination framework of generalized adversarial example construction and adaptive adversarial training strategy for model smoothness to enhance the robustness transferability.
- We conduct extensive experiments to evaluate the effectiveness of Alchemy under different settings and against different attacks. We also discuss the generator setting and selection for future research directions.

2 BACKGROUND

2.1 Adversarial Attacks and Defenses

Deep learning models have already achieved better performance than humans on many tasks [26, 82]. Unfortunately, well-trained models can be misled by adversarial examples that appear to be

[1]https://huggingface.co/models

similar to natural examples but contain imperceptible adversarial perturbations [81]. Adversarial example attacks pose significant threats to real-life systems, *e.g.*, speech processing [17, 22, 23], automatic driving [25, 49], facial recognition system [88, 110], deepfake forensics [9]. Even the currently booming language models [48, 105] and diffusion models [41] are shown to be affected by adversarial examples [55, 70, 95, 109].

White-box adversarial attacks mainly follow two lines of methods, i.e., gradient-based and optimization-based. Fast Gradient Sign Method (FGSM) is the first gradient-based attack [34], which generates adversarial examples with a single gradient step. R+FGSM [85] adds a random initialization on the input sample to enhance FGSM. Projected Gradient Descent (PGD) is the multi-step variant of FGSM, which is still one of the most efficient and powerful attacks so far [51, 58]. AutoPGD [20] enhances PGD by overcoming failures due to suboptimal step size and problems of the objective function. C&W [12] formulates an optimization problem that minimizes the adversarial perturbations to achieve the attack objective. Extensions to C&W attacks include DeepFool [62], SPSA [86], and FAB [19]. Many black-box adversarial attacks are based on the transferability of adversarial examples, i.e., an adversarial example constructed based on a white-box model is also effective against a different black-box model. Black-box attacks can also be achieved by estimating the gradient of the target model through query [16, 37, 44, 106].

Defense methods against adversarial examples can be divided into two main categories, i.e., sample processing based methods and model transformation based methods [108]. Sample processing methods focus on detecting and purifying adversarial examples. Adversarial examples may be differentiated from benign samples using statistical features or by a binary detection model [36, 74]. After being detected, adversarial examples may be purified by image processing techniques, *e.g.*, JPEG compression [56] and pixel deflection [4, 68], or image reconstruction methods [104]. However, sample processing methods may be circumvented by adaptive attacks [11, 83]. Model transformation methods aim at strengthening the model to improve its endogenous robustness against adversarial example attacks. Knowledge and logic reasoning are utilized to reconstruct the model to better mimic the human thinking process [38, 103]. Model ensemble is used to turn multiple weak models into a strong one [1, 85]. Denoised smoothing [10, 75] originated from randomized smoothing is another way to provide the probabilistic certification from theory perspective. The certified accuracy is usually lower than the experimental results. And the Denoiser also needs to be trained with large dataset of clean images, which is unavailable in the data-free scenario. The smoothing method is mainly used under small norm perturbations, usually l_2 norm. Adversarial training retrains the model with adversarial examples to improve its robust accuracy [7, 31, 76, 85]. Among all defense strategies, adversarial training is by far the most widely used and most successful one, especially against white-box adaptive attacks [5, 84]. In this paper, we focus on enabling adversarial training in a data-free setting.

2.2 Adversarial Training

We consider a deep neural network model f_θ that performs an N-class classification task, mapping samples $x \in \mathcal{X}$ to labels $y \in \mathcal{Y}$. Let $\mathcal{B}(x)$ denote the set of adversarial examples of the benign sample x. Adversarial training aims to solve the minmax problem [7, 42, 77, 90]

$$\min_{\theta} \max_{x' \in \mathcal{B}(x)} \mathcal{L}(f_\theta(x'), y). \tag{1}$$

where $\mathcal{L}$ is the loss function, θ is the parameter of f.

Directly solving the minmax optimization problem is difficult. Mainstream adversarial training methods usually adopt a two-step solution [77]. First, adversarial examples are created against the target model using white-box attacks, which approximates the inner maximization problem. Second, the target model is retrained using the generated adversarial examples, which approximates the outer minimization problem. PGD is widely used in adversarial training [52, 58], which greatly improves the robustness of the model compared to prior methods and becomes the mainstream setting of the adversarial training up to now [69].

2.3 Data-Free Knowledge Distillation

Knowledge distillation [35, 61, 89] enables a student model to learn from a well-performed teacher model may be leveraged to train the student model to perform the customized task and meet specific needs. For example, due to the resource constraints of most edge devices like embedding systems, large neural network models cannot be deployed and run. The teacher model needs to be compressed into a small but still functional student model [8]. In the incremental learning scenarios [107], the student model needs to learn new classes of samples without forgetting the teacher knowledge [79]. But due to the privacy issues or the intellectual property issues like medical data, portrait data, and other large-scale datasets, the original training datasets may be impossible to obtain. To solve such problems, data-free learning emerged in recent years [14, 28].

The essential step of data-free learning is constructing training data from the well-trained teacher model. The generated data distribution similarity with the original training data is the main concern for the student further training. The dataset diversity is also important to avoiding the mode collapse problem. The data construction can be categorized into the optimization based methods and generator based methods. The optimization based methods directly optimize the training data in the input space. The early work optimizes the dataset based on the recording activation statistics [57]. The model layer information is utilized to estimate the feature statistics, therefore recovering the training data. The mean and and variance in batch normalization layers are firstly used in [94]. The generator based methods leverage a generator to transform random noise into high-fidelity images. Data-Free Learning (DAFL) [14] proposes an efficient framework for data-free model compression with generative adversarial learning networks (GANs) [33]. In this framework, the well-trained model plays a role of a discriminator and the generator is guided by it to produce synthetic training samples. Contrastive Model Inversion (CMI) [28] attempts to solve the mode collapse problem to increase the diversity of synthetic data. Spaceship-Net proposes to use channel-wise feature exchange

and multi-scale spatial activation region consistency constraint to efficiently synthesize diverse images [98]. A few studies working on data-free knowledge distillation train the student model without reconstructing training dataset, instead, using Gaussian noise [71], but the results are unsatisfactory.

3 THREAT MODEL

First, we elaborate in detail on our motivation, which is also the adversarial scenario where the data-free adversarial training is necessary. Then we portray the threat model in terms of the knowledge, capability, and goal of the attacker and the defender.

3.1 Motivation

As the number of machine learning model users increases and the cost of training models rises, the open-source community for machine learning models is gradually taking shape. More and more model trainers will open-source their trained models to platforms such as Hugging Face, GitHub, and Modelscope. Many tech companies such as Microsoft, Meta, Google, Alibaba, and others also open-source their trained models as technological products for public use. In this scenario, users may download these models and apply them to their critical applications, such as security monitoring, feature recognition, and more. These models may accurately make predictions in normal circumstances. However, due to the high time and economic costs, model trainers often do not conduct standard adversarial training.

Therefore, when faced with attackers using adversarial perturbations, these models are easily induced to make incorrect predictions, causing losses for the users. For resource-limited individuals or companies who want to use open-source pre-trained models, their local dataset may be much smaller than the original training dataset. The local dataset may be able to slightly fine-tune the pre-trained model but is not big enough to equip the model with adversarial robustness. This may hinder the application of the model to critical missions, e.g., healthcare or security. So far, out of all the models exceeding 480,000 on Hugging Face, less than one-thousandth claim to be robust. On the Chinese open-source model platform Modelscope, out of 3000 models, less than one percent claim to be robust. We also conduct a user study in the open-source model community. Out of 50 users who had previously downloaded open-source models, only 4% of them used models that had undergone adversarial training before downloading. However, 86% of the users hoped that the models they used were robust, even though 82% of these users couldn't access the source training data for the downloaded models. Out of 20 users who had previously uploaded models, considering issues such as high costs and unclear demand, 18 users did not want to conduct adversarial training on models before uploading. Therefore, users need a method to customarily perform adversarial training on pre-trained models they acquire without access to the original training dataset.

3.2 Knowledge

We respectively define the attacker and the defender's knowledge. For attacker we consider both the white-box attack and the black-box attack.

(1) *Attacker's knowledge.* For white-box attack, the attacker has full knowledge of the structure and the parameter of the target implemented model. The attacker can leverage the parameter to calculate the adversarial examples. For black-box attack, the attack can not access the parameter of the target model. The attack has a substitute model for generating the adversarial examples.

(2) *Defender's knowledge.* The defender only has knowledge of the pre-trained model f^T trained on the original dataset $\mathcal{X}_{ori}$.

3.3 Capability

We respectively define the attacker and the defender's capability.

(1) *Attacker's capability.* To create disruption, the attacker adds adversarial perturbations on the input data to mislead the model to make wrong predictions. To avoid detection from human beings, the attackers keeps the adversarial example of x_i in the region of

$$\mathcal{B}_\epsilon^p(x_i) = \{x_i' \in \mathcal{X} : \left\|x_i' - x_i\right\|_p \leq \epsilon\}. \tag{2}$$

The p-norm of a vector is defined as the pth root of the sum of the pth powers of the absolute values of its components. Mathematically, for a vector $x = (x_1, x_2, \ldots, x_n)$, the p-norm is given by $\|x\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{\frac{1}{p}}$. The infinite norm, also referred to as the maximum norm, yields the maximum absolute value of the vector's elements. We use the infinite norm as the adversarial setting for the rest of our paper.

(2) *Defender's capability.* The defender performs re-training on f^T to get the model f^T for deployment. But the defender can not access $\mathcal{X}_{ori}$.

3.4 Goal

The attacker and the defender engage in an adversarial game. For input x_i and its ground truth label y, the attacker generates adversarial example $x_i' \in \mathcal{B}_\epsilon^p(x_i)$ and aims at deceiving the deployment model f^S by $\arg\max_i f^S(x)_i \neq y$. Conversely, the defender aims at making correct prediction by f^S, which is $\arg\max_i f^S(x)_i = y$.

4 SYSTEM DESIGN

Alchemy's aim is to conduct adversarial training on a pre-trained model f^T without reliance on any original training data $\mathcal{X}_{ori}$. An overview of Alchemy is shown in Figure 2.

Our framework comprises three primary modules: robust sample generation, generalized adversarial example, and smooth adversarial training. The robust sample generation module addresses the critical challenge of lacking adversarial training datasets by generating robust sample datasets $\mathcal{X}_{sub}$ with the aid of f^T. The robust features benefit the downstream adversarial training. The disparities between the generated dataset and the original dataset present a pronounced challenge for the adversarial robustness transferability. So the adversarial example construction produces generalized and effective adversarial examples $\mathcal{X}'_{sub}$ against f^S. The smooth adversarial training further use $\mathcal{X}'_{sub}$ to improve the model smoothness, accomplishing the robustness improvement on $\mathcal{X}_{ori}$. We summarize the whole brief algorithm pseudocode in Algorithm 1.

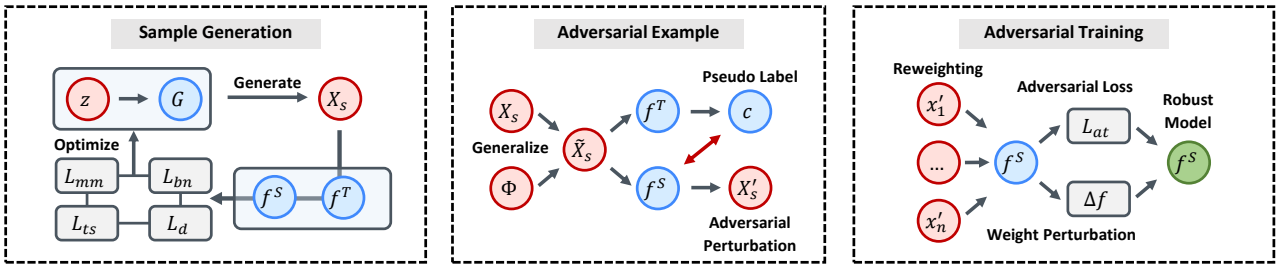
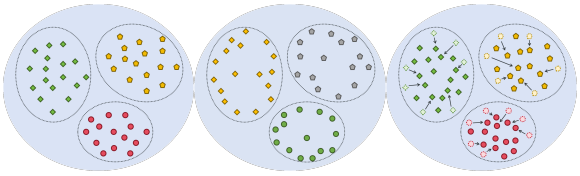


Figure 2: Overview of ALCHEMY. ALCHEMY consists of the robust sample generation, the generalized adversarial example module, and the smooth adversarial training module. The sample generation module leverages adversarial margin, batch normalization, T-S disagreement and memory bank to generate the training samples with robust features for adversarial training. Then we use generalized convolution and max-margin target loss to generate adversarial noise. The adversarial training module leverages smoothness loss function and weight perturbation to enhance the student model.

Algorithm 1: ALCHEMY: data-free adversarial training

Input : A pre-trained model f^T
Output : A model f^S with adversarial robustness

Generation Dataset $\mathcal{B} \leftarrow \emptyset$;
Initialize the generator G;
for *i in [0, ..., N]* **do**
  /* Robust Sample Generation */
  $z \leftarrow \mathcal{N}(0, 1)$; ▷ Initialization
  for g_{step} **do**
    $x \leftarrow G(z)$; ▷ Sample generated
    $z \leftarrow z - \eta_z \nabla_z \mathcal{L}_G(x, f^T)$;
    $G \leftarrow G - \eta_G \nabla_G \mathcal{L}_G(x, f^T)$; ▷ Samples optimized
  end
  $\mathcal{B}, l \leftarrow \mathcal{B} \cup G(z), 0$; ▷ Training dataset construction
  for x *in* $\mathcal{B}$ **do**
    /* Generalized Adversarial Example */
    $x', y \leftarrow G_{ae}(x), f^T(x), f^S$; ▷ Adversarial examples
    /* Smooth Adversarial Training */
    $f^S \leftarrow f^S + \Delta f$;
    $l \leftarrow l + \mathcal{L}_{at}(f^S, x, x', y)$; ▷ Adversarial loss
    $f^S \leftarrow f^S - \Delta f$; ▷ Weight perturbation
  end
  $f^S \leftarrow f^S - \eta_s * \nabla l$; ▷ Model robustness updating
end



(a) Original Dataset (b) Generated Dataset (c) Adversarial Margin

Figure 3: Generated dataset distribution illustration with adversarial margin principle. Normal generated dataset contains boundary samples with non-robust features. Adversarial margin loss leads the generator to generate robust samples which benefit the adversarial training.

4.1 Robust Sample Generation

Different from the sample generation used in the knowledge distillation [28, 57] or incremental learning [79], sample generation in ALCHEMY serves the main target of enhancing the adversarial robustness. Based on the adversarial training theories [6, 45, 87], robust models after adversarial training focus more on robust features as humans rather than the non-robust features, which are considered as the source of the imperceptible adversarial examples. Models trained on datasets with more robust features also show more robustness than the non-robust features. So in ALCHEMY, we encourage the sample generator to generate samples with more robust features instead of non-robust features.

To be detailed, we leverage a generator model $\mathcal{G}$ to transform the random Gaussian noise z to $\mathcal{G}(z)$ of the same dimension with the original training sample and get the training samples $x = \mathcal{G}(z)$. To construct the substitute dataset $\mathcal{X}_{sub}$, at each generation epoch, we optimize $\mathcal{G}$ and z to minimize the reconstruction loss, which significantly contributes to the data reconstruction process following the prior literature.

To generate training samples with robust features, we design the objective function with adversarial max-margin generation loss. The former sample generation methods only optimize the target label possibility. Due to the nonlinearity of deep learning models, many samples with non-robust features can also induce the model to output high target label possibility. Also, these samples usually are typically distributed around the decision boundary of the teacher model, so they will produce misleading effects in adversarial training. To address this problem, we propose to use max-margin loss to encourage the samples to reduce the predicted probability values for other labels while increasing the target label probability. As illustrated in our diagram Figure 3, regular generated dataset contains samples closer to the decision boundary, which have non-robust features and degrade the adversarial training. By keeping sample distance to the boundary, the generate samples contain

more robust features. So we leverage max-margin loss to encourage the sample to keep the distance to the boundary and enforce the teacher model to predict the target label as follows:

$$\mathcal{L}_{mm}(x_i) = -f^T(c|x_i) + \max_{c' \neq c} f^T(c|x_i) + \mathcal{L}_{ce}(f^T(x_i), c) \quad (3)$$

Also for the best reconstruction effect, we leverage the batch normalization loss $\mathcal{L}_{bn}$ to reinforce the samples with the model's memory saved in the batch norm layers, teacher-student model disagreement loss $\mathcal{L}_{ts}$ to enhance the learning effectiveness and sample diversity loss $\mathcal{L}_{cr}$ to improve training sample diversity.

- ***Batch normalization loss.*** The batch normalization layers maintain the mean and variance statistic information of the features on the layers to normalize the input data distribution. The statistic information in batch normalization layers servers as natural and effective restraints for the data construction as:

$$\mathcal{L}_{bn}(x_i) = \sum_l D\left(\mu_l(x_i), \sigma_l^2(x_i), \mu_l, \sigma_l^2\right) \quad (4)$$

- ***Teacher-Student disagreement loss.*** Larger disagreement between the teacher model and the student model is usually considered to be beneficial to knowledge distillation. So we maximize the disagreement by the loss function:

$$\mathcal{L}_{ts}(x_i) = -KL(f^T(x_i), f^S(x_i)) \quad (5)$$

- ***Sample diversity loss.*** We also leverage the memory bank from [28] to encourage the data diversity as generation goes on to avoid data collapse. We use the contrastive learning loss to encourage the generation model to generate samples different from the former samples. With history samples as x_j^-, transformed samples as x_j^+, and an instance discriminator $h(\cdot)$, we use the contrastive loss function as [28]:

$$\mathcal{L}_{cr}(\mathcal{X}, h) = \mathbb{E}_{x_i \in \mathcal{X}}\left[\log \frac{\sum_j \exp\left(\text{sim}\left(x_i, x_j^-, h\right)\right)}{\exp\left(\text{sim}\left(x_i, x_i^+, h\right)\right)}\right] \quad (6)$$

So the construction loss $\mathcal{L}_G$ composes of the aforementioned loss function as:

$$\mathcal{L}_G = \theta_{mm} * \mathcal{L}_{mm} + \theta_{bn} * \mathcal{L}_{bn} + \theta_{ts} * \mathcal{L}_{ts} + \theta_d * \mathcal{L}_d. \quad (7)$$

For each epoch, the random input z and generator model $\mathcal{G}$ are optimized by the loss function $\mathcal{L}_G$ to meet the several optimization goals. The sample generation can be adapted to other domains like audio or NLP with minor modifications. For robust sample generation, L_{mm}, L_{ts}, and L_{cr} can be directly applied to any classification models. L_{bn} provides statistical prior knowledge which can be replaced by pre-trained language prediction model or human speech features.

4.2 Generalized Adversarial Example

After generating training samples in each epoch, we produce adversarial examples $\mathcal{X}'_{sub}$ on the substitute dataset $\mathcal{X}_{sub}$ for the further adversarial training on the student model f^S. Different from the former adversarial training where the training dataset $\mathcal{X}_{train}$ and the test dataset $\mathcal{X}_{test}$ can be considered to be independent and identically distributed, $\mathcal{X}_{sub}$ and $\mathcal{X}_{test}$ are in fact differently distributed. So the robustness on the $\mathcal{X}'_{sub}$ can not be naturally transferred to the original test dataset $\mathcal{X}_{test}$. So we proposed the methods to improve adversarial examples' efficacy and generalization.

For effective adversarial example generating, we first address the label lacking on dataset $\mathcal{X}_{sub}$ with the model f^T. We choose the highest possibility in the output $f^T(x)$ as the label c for sample x. Then we leverage max-margin loss as the target loss to craft adversarial noise. To be detailed, we construct adversarial examples x' to increase the max-margin loss:

$$x'_i = \underset{x \in \mathcal{B}_\epsilon^p(x_i)}{\text{argmin}} \left(f^S(c|x_i) - \max_{c' \neq c} f^S(c|x_i)\right). \quad (8)$$

The max-margin loss not only decreases the target label probability but also increases the probability of the other classes. It is known to generate stronger attacks compared to the normal cross-entropy loss and KL disagreement. To be noted, the max-margin loss doesn't conflict with the one used in Sec. 4.1. The two objective function are optimized based on f^T and f^S respectively. The training sample generation is optimized on f^T to generate robust feature samples from the teacher model and the adversarial examples are constructed to find the vulnerabilities of the student model. We also leverage data augmentation to generalize the adversarial robustness to the unseen original test dataset. As the generation and training progress, we gradually introduce transformation to the examples to compromise non-robust features and enhance the model's learning of transferable features. We leverage augmentation convolution layers Φ to transform the input data from $\mathcal{X}_{sub}$. $\widetilde{x_i}$ is the linear combination of the original samples and the augmented samples with θ_g, which is dynamically adjusted as the training progresses.

4.3 Smooth Adversarial Training

For adversarial training in ALCHEMY, it's more important to improve robustness that can be transferred to the target test dataset than robustness on the generated dataset $\mathcal{X}_{sub}$. For better transferability, we propose smooth adversarial training to improve the model's endogenous smoothness against input variations.

4.3.1 Smoothness loss function. To be detailed, for the adversarial example x', we calculate the Kullback-Leibler divergence inspired by [91] between the student model output $f^S(x')$ and the clean output $f^S(x)$, which can be quanlified as:

$$\mathcal{L}_{kl} = \sum_{k=1}^{K} f_k^S(\mathrm{x}_i) \log \frac{f_k^S(\mathrm{x}_i)}{f_k^S(\hat{\mathrm{x}}'_i)}. \quad (9)$$

K is the number of sample classes. We leverage the $\mathcal{L}_{kl}$ as the main adversarial training loss accompanied by the cross entropy loss of the adversarial example x' and the target label c with sample reweighting. Our goal is to ensure that the model exhibits minimal output variation when subjected to adversarial noise, achieving the effect of smoothness. So the adversarial training is optimized with the loss function as follows:

$$\mathcal{L}_{at} = \mathcal{L}_{kl} + \theta_{ce} * \mathcal{L}_{ce} \quad (10)$$

θ_{ce} is used to balance the model smoothness goal and the robustness on the generated dataset.

The generated samples exhibit a wide variation in their generative effects, and there are discernible differences in the impact of various adversarial samples on adversarial training. Consequently,

Table 1: Overall Performance of ALCHEMY compared with baselines against PGD-20.

DS	Metrics	OD†	DAFL	DI	SSN	DDPM	ALCHEMY
MN	T.ACC‡	98.56%	77.96%	10.16%	9.50%	97.87%	**98.84%**
	G.ACC		92.55%	10.26%	4.72%	99.06%	97.96%
	T.B.R	98.21%	58.81%	10.16%	9.50%	**97.18%**	96.94%
	G.B.R		73.47%	10.26%	4.72%	99.02%	94.26%
	T.W.R	96.13%	11.84%	10.16%	9.50%	93.53%	**93.66%**
	G.W.R		30.13%	10.26%	4.72%	98.64%	75.94%
CF	T.ACC	81.71%	16.00%	52.88%	60.64%	50.43%	**73.56%**
	G.ACC		93.13%	96.42%	21.76%	75.25%	90.06%
	T.B.R	79.68%	13.58%	47.56%	54.94%	49.46%	**66.26%**
	G.B.R		92.91%	95.70%	19.26%	74.69%	80.44%
	T.W.R	60.44%	8.28%	15.74%	46.50%	20.69%	**58.02%**
	G.W.R		51.59%	87.98%	14.86%	55.32%	60.72%
SV	T.ACC	93.16%	74.49%	60.32%	6.80%	75.04%	**83.90%**
	G.ACC		94.93%	93.68%	5.94%	94.46%	94.62%
	T.B.R	88.55%	32.94%	52.62%	6.80%	70.54%	**71.14%**
	G.B.R		82.47%	93.58%	5.94%	94.31%	86.70%
	T.W.R	66.48%	6.03%	20.40%	6.80%	31.07%	**50.54%**
	G.W.R		68.81%	87.70%	5.94%	79.53%	72.94%
CN	T.ACC	74.09%	10.44%	21.44%	45.60%	40.08%	**59.26%**
	G.ACC		79.76%	74.68%	16.00%	76.05%	75.88%
	T.B.R	73.54%	9.53%	21.34%	44.84%	39.85%	**57.44%**
	G.B.R		22.83%	73.78%	15.92%	76.15%	72.54%
	T.W.R	53.97%	5.84%	9.36%	33.76%	14.33%	**39.78%**
	G.W.R		32.66%	46.62%	10.34%	60.98%	34.88%

† OD: the performance of model trained on the original data.

‡ T.ACC: clean data accuracy on the original dataset. G.ACC: clean data accuracy on the generated dataset. T.B.R: robustness on the original dataset under the black-box attack. G.B.R: robustness on the generated dataset under the black-box attack. T.W.R: robustness on the original dataset under the white-box attack. G.W.R: robustness on the generated dataset under the white-box attack. The perturbation ranges are 0.1, 8/255, 8/255, 8/255 for MNIST, CIFAR-10, SVHN, and CINIC-10.

a reweighting strategy can be tailored to increase the weights of samples more conducive for robustness transferring after a period of training. The adversarial noise intensity can be leveraged as the penalty term to increase the weight of the samples with small adversarial noise. For generation effectiveness, the various generated samples can be reweighted with the generation quality.

4.3.2 Weight Perturbation. To better improve the adversarial training smoothness against overfitting on the generated dataset, we leverage the weight perturbation methods to flatten the weight loss landscape [93]. Before each epoch, we perturb the weight with Δf^S based on the adversarial parameter perturbation. The weight perturbation is computed using the gradient of the loss function with respect to the weights to determine the direction in which the weights should be perturbed to maximize the loss. Then we optimize the perturbed model with the adversarial training loss L_{at} and update the parameters. At the end of the epoch, we restore the model weight by removing the model perturbation Δf^S added at the beginning. Weight Perturbation is proven to bring flatter weight loss landscape and benefit the model smoothness instead of only the robustness on the generated dataset [93], which is suitable to enhance the intrinsic smoothness of the model in the data-free scenario.

Table 2: Performance of ALCHEMY on max-margin loss.

DS	θ_{mm}	T.ACC	G.ACC	T.B.R	G.B.R	T.W.R	G.W.R
MN	0.00	94.58%	50.86%	93.08%	43.64%	85.22%	13.46%
	0.05	98.62%	92.64%	97.54%	80.66%	**94.82%**	37.14%
	0.10	98.84%	97.96%	96.94%	94.26%	93.66%	75.94%
	0.15	98.51%	99.34%	96.36%	98.22%	88.95%	85.20%
	0.20	98.62%	99.44%	94.22%	99.42%	72.36%	99.40%
CF	0.00	71.62%	81.04%	64.86%	71.68%	48.02%	56.76%
	0.05	74.24%	90.66%	66.50%	82.70%	57.66%	66.46%
	0.10	73.56%	90.06%	66.26%	80.44%	**58.02%**	60.72%
	0.15	73.96%	91.86%	65.94%	83.24%	55.32%	63.10%
	0.20	73.96%	93.50%	66.08%	86.84%	57.68%	70.42%
SV	0.00	79.97%	86.72%	68.24%	76.00%	42.60%	55.46%
	0.05	81.95%	90.54%	70.91%	77.92%	45.41%	60.74%
	0.10	83.90%	94.62%	71.14%	86.70%	**50.54%**	72.94%
	0.15	81.04%	88.90%	66.64%	64.30%	43.32%	39.60%
	0.20	80.19%	86.59%	66.20%	65.83%	41.06%	36.47%
CN	0.00	53.52%	62.60%	52.52%	59.94%	34.19%	30.88%
	0.05	56.46%	67.13%	56.91%	65.83%	39.05%	32.91%
	0.10	59.26%	75.88%	57.44%	72.54%	**39.78%**	34.88%
	0.15	53.54%	81.58%	51.56%	79.92%	34.82%	43.92%
	0.20	44.38%	67.34%	36.50%	55.58%	30.24%	49.14%

5 EVALUATION

5.1 Experiment Setup

5.1.1 Dataset. We evaluate ALCHEMY on four datasets with different neural network architectures. Specifically, we choose four popular datasets, i.e., MNIST [24], CIFAR- 10 [50], SVHN [63], CINIC-10 [21].

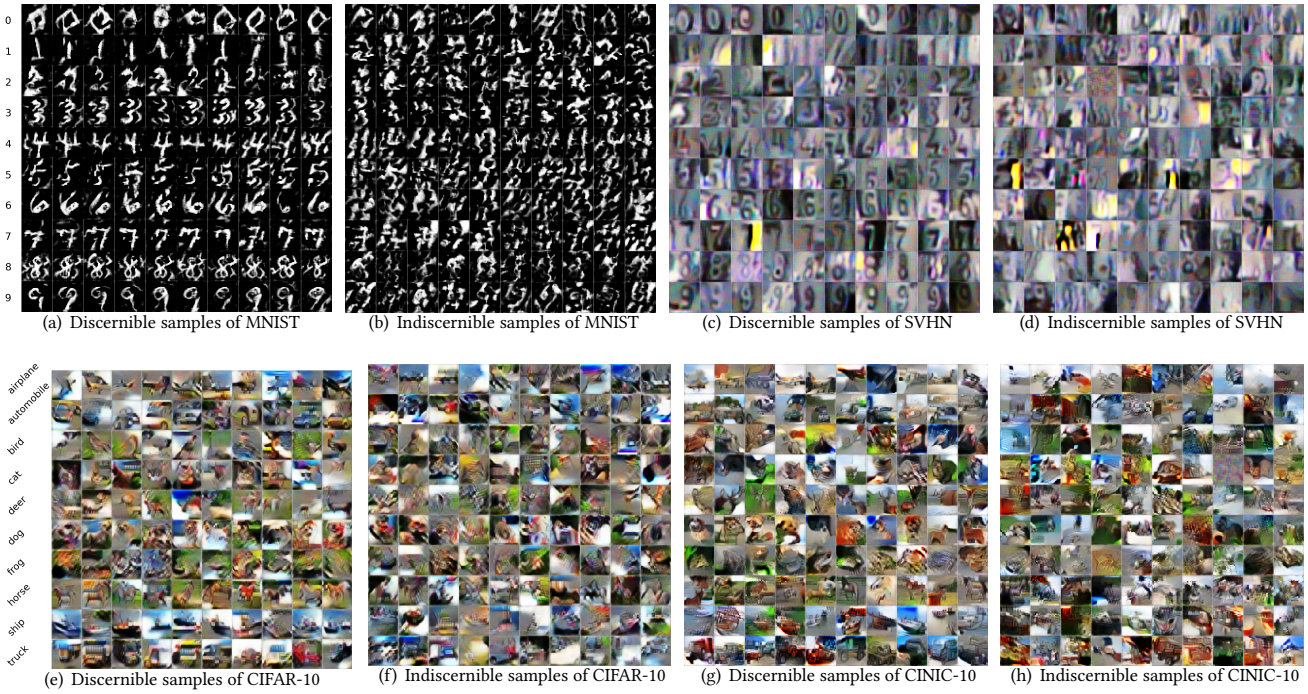
MNSIT. MNIST (MN) contains 70,000 28×28 gray-scale images of handwritten digits 0 to 9 in 10 classes, with 7,000 images per class. The training set contains 60,000 samples, and the remaining 10,000 are used as the test set.

CIFAR-10. CIFAR-10 (CF) consists of 60,000 32×32 RGB colour images in 10 classes, with 6,000 images per class. The training set and test set contain 50,000 and 10,000 samples respectively.

SVHN. SVHN (SV) includes 99,289 32×32 RGB colour images with labels of printed digits 0 to 9 from the real world. We randomly select 73,257 labeled samples for training, and 26,032 labeled samples for testing.

CINIC-10. CINIC-10 (CN) has a total of 270,000 32×32 real-world colour images in 10 classes. The dataset is evenly split into three subsets: training set, validation set, and test set, each of which contains 90,000 samples.

5.1.2 Baselines. Since there are no adversarial learning frameworks in the existing research works, we compare the accuracy and robustness performance of ALCHEMY with baselines adapted from data-free learning frameworks to the widely used adversarial framework TRADES [101], which is the pivotal AT framework that integrates many technical improvements including early stopping, augmentation, and tuning and the base of many recent frameworks. As shown in Table 1, we compare ALCHEMY with four baselines. Data-Free Learning (DAFL) [14] is one of the most popular data-free learning frameworks with the generative network in the knowledge distillation and model compression. Besides the one-hot loss to encourage the outputs to be close to the target label and information entropy loss to have balanced generated images, DAFL also



(a) Discernible samples of MNIST (b) Indiscernible samples of MNIST (c) Discernible samples of SVHN (d) Indiscernible samples of SVHN

(e) Discernible samples of CIFAR-10 (f) Indiscernible samples of CIFAR-10 (g) Discernible samples of CINIC-10 (h) Indiscernible samples of CINIC-10

Figure 4: Generated sample visualization. For each dataset, we elaborate both the samples with/without discernible category characteristics. The corresponding target categories of generation samples are on the far left side. MNIST and SVHN share the same categories. CIFAR-10 and CINIC-10 share the same categories.

optimizes the generator and the input z to have a higher activation value in the feature maps. DeepInversion (DI) [94] is one of the state-of-the-art works in non-generator knowledge distillation. DI first utilizes batch normalization layer information to impose constraints on the generated sample mean and variance. SpaceShipNet (SSN) [98] is also a generator-based method, which proposes Channel-wise Feature Exchange to leverage the history feature channel data to obtain the mixed features for crafting new diverse training samples. The generation is more efficient by the feature mixture. Also, SSN aligns the spatial activation region of the teacher network and the student model to alleviate the influence of unwanted noises in diverse synthetic images on distillation learning. The Denoising Diffusion Probabilistic Model (DDPM) [41] is utilized in improving adversarial robustness in a lot of the methods[64, 72, 92]. But these methods are supposed to have full access to the original training dataset. So we adapt the DDPM methods from [72] to have access to a small part (each class 100 samples) of the original dataset to approximate our data-free scenario.

5.1.3 Model and Attack settings. For the overall performance, we utilize the most popular model settings as the pre-trained model f^T for each dataset. We evaluate the accuracy and white-box adversarial robustness against PGD-20 attacks for f^T elaborated in Table A1. For MNIST, we utilize CNN [43] with four convolutional layers and two fully-connected layers as the teacher model structure. The model accuracy is 99.57% and the white-box robustness is 2.23%. For CIFAR-10, we utilize ResNet34 [39] as the teacher model structure. The model accuracy is 93.04% and the white-box robustness is 0.01%. For SVHN, we utilize ResNet34 as the teacher model structure and the model accuracy is 95.00% and the white-box robustness is 1.81%. For CINIC-10, we utilize ResNet34 as the teacher model structure and the model accuracy is 89.64% and the white-box robustness is 0.01%. From the above previous settings, we can see that the teacher models all have no adversarial robustness against adversarial attacks. We will test different model structures in the hyperparameter experiments. To test the methods' effect, we evaluate the models' accuracy and the model accuracy under black-box attack and white-box attack. The white-box attack is evaluated with PGD-20, assuming that the attacker has full knowledge of the student model. The black-box attack [3, 66] is evaluated with PGD-20, assuming that the attacker has a surrogate model of the same structure as the student model. We use the teacher model f^T as the surrogate model.

The experiments are carried out on our workstations equipped with Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz, running Ubuntu 18.04 system. We use one NVIDIA GeForce RTX 4090 Graphics Card for all the model training.

5.2 Overall Performance

To evaluate the overall performance, we conduct Alchemy on the above four datasets, compared with other baselines. To better evaluate how the generated samples generalization to the original training data, we both test the accuracy, black-box accuracy, and white-box accuracy on the original test dataset, as **T.ACC**, **T.B.R**, **T.W.R**,

and generated test dataset, as **G.ACC**, **G.B.R**, **G.W.R**. We show the model effectiveness in Table 1. The results on the original dataset are the most important, so we highlight the best results among all the results on the original training data. So from the table, we can see that Alchemy achieves the best results on almost all the datasets and metrics. We also elaborate the adversarial training results with the original dataset. From the results, we can see that for MNIST and CIFAR-10, the robustness results are very close to the results trained on the original data. For SVHN and CINIC-10, the results are around 15 percent from the original dataset results, which is much better than the baselines. For all the baselines, the DDPM method shows relatively effective results. But on CIFAR-10, SVHN, and CINIC-10, the white-box robustness results have an average deviation of 30 percentage points compared to Alchemy. DDPM method relies on 100 original training samples for each class and Alchemy uses none. Also, the adversarial training is prone to instability when applied to generated data. As we can see, the SSN method shows an instable model on the MNIST and SVHN results and the DI method also shows an instable model on the MNIST. Also, the Alchemy accuracy on the clean dataset is the highest compared with the other baselines, showing that Alchemy has the least influence on the normal test dataset accuracy.

Observation 1

Without access to the training dataset, Alchemy attains high adversarial robustness **(T.B.R and T.W.R)** on PGD-20 without degrading the accuracy **(T.ACC)** excessively.

Furthermore, we also illustrate the generalization gap existing between the outcomes derived from the generated data and those from the original training data. When juxtaposed with other baselines, we observe that the generalization gap achieved by our approach is notably minimized. Specifically, for CIFAR-10, the white-box robustness on the original dataset closely aligns with the robustness observed on the generated dataset, signifying successful transferability of robustness from the generated dataset to the original dataset. In the case of MNIST and CINIC-10, the white-box adversarial robustness on the original dataset slightly surpasses the robustness on the generated dataset. However, for SVHN, the generalization gap registers at approximately 23% with regard to white-box robustness. It is evident that for alternative baseline methods such as DAFL, DI, and DDPM, the accuracy, as well as the black-box and white-box robustness, is markedly higher on the generated dataset. Nevertheless, when assessed on the original dataset, these methods exhibit considerable declines, underscoring the critical role of generalization in achieving adversarial robustness on the original dataset. For SSN method, the robustness is obviously higher on the original dataset, but the generated dataset robustness is much lower than the other methods and faces training non-convergence phenomenon in MNIST and CINIC-10.

To keep the generated dataset balanced, we assign the target label c evenly in L_{mm}. The diversity loss L_{cr} also contributes to the sample balancing. In our experiment settings, it takes 5-20 minutes to generate balanced datasets in each epoch. Since adversarial training is usually conducted offline, the overall time to generate the whole dataset is not particularly restrictive for training. Alchemy is scalable to more complicated datasets.

Table 3: Performance of Alchemy on different generation steps.

DS	g_{step}	T.ACC	G.ACC	T.B.R	G.B.R	T.W.R	G.W.R
MN	50	64.94%	82.98%	59.84%	73.24%	47.54%	33.18%
	100	96.19%	97.16%	94.21%	91.44%	89.17%	59.46%
	300	98.20%	99.32%	94.50%	99.34%	84.58%	99.28%
	500	98.84%	97.96%	96.94%	94.26%	**93.66%**	75.94%
	800	95.09%	96.44%	90.67%	94.99%	82.19%	69.20%
CF	50	62.18%	80.90%	56.62%	74.82%	45.46%	63.76%
	100	67.26%	82.54%	58..43%	73.62%	46.16%	66.33%
	300	69.38%	86.62%	65.31%	79.18%	46.99%	67.32%
	500	73.56%	90.06%	66.26%	80.44%	58.02%	60.72%
	800	74.21%	90.19%	69.44%	76.89%	**58.40%**	64.17%
SV	50	77.19%	74.88%	63.07%	59.61%	44.78%	42.64%
	100	73.36%	70.18%	63.46%	58.90%	41.08%	39.42%
	300	80.30%	92.76%	68.48%	81.00%	40.58%	57.58%
	500	83.90%	94.62%	71.14%	86.70%	**50.54%**	72.94%
	800	83.22%	94.36%	73.64%	83.09%	50.22%	76.14%
CN	50	31.49%	52.95%	38.01%	47.21%	31.61%	30.67%
	100	40.70%	48.96%	40.12%	48.06%	25.52%	23.26%
	300	55.48%	65.30%	53.88%	62.36%	39.52%	32.10%
	500	59.26%	75.88%	57.44%	72.54%	**39.78%**	34.88%
	800	56.26%	49.44%	49.16%	68.75%	38.24%	34.21%

Observation 2

Alchemy achieves high robustness transferability from the generated dataset to the original dataset **(T.W.R - G.W.R)** while maintaining the local robustness **(G.W.R)**.

5.3 Generated Sample Visualization

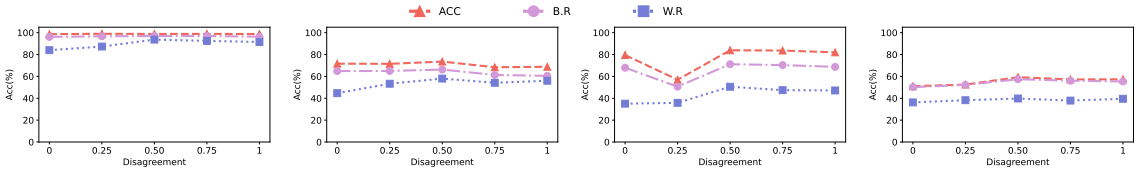
In this section, we visualize the data generated during the sample generation phase in Figure 4. For comprehensiveness, we show both discernible and indiscernible samples in the generated datasets. As shown in the discernible samples, we can clearly observe the characteristics of the categories. Indiscernible samples poses less characteristics to human eyes but also have learnable information for the models. It can also be observed from the results that the data we generated has a high diversity for further learning. Whether adversarial training on indiscernible samples will benefit the robustness on the original dataset is unclear.

Observation 3

Diversified generation by Alchemy produces both discernible and indiscernible samples. How to select from generated data for adversarial training remains a challenging issue.

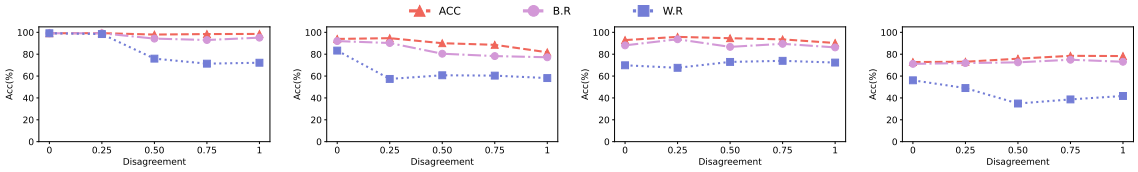
5.4 Impact of Hyperparameters

In order to affirm the efficacy of Alchemy in enhancing robustness, we subject it to rigorous testing across a comprehensive array of hyperparameter configurations. Our evaluation encompasses different structural settings, different max-margin loss parameters, different generation step adjustments, and different disagreement loss settings on all four datasets. Through hyperparameter tuning, we also summarized the techniques for adversarial training in data-free scenarios. In each subsection, we have also specified the default heperparameter settings we used. For page limit, we place the different structural setting evaluation in Appendix A.1.



(a) MNIST (b) CIFAR-10 (c) SVHN (d) CINIC-10

Figure 5: ALCHEMY results on different teacher-student model disagreement on the original training dataset.



(a) MNIST (b) CIFAR-10 (c) SVHN (d) CINIC-10

Figure 6: ALCHEMY results on different teacher-student model disagreement on the generated substitute dataset.

Table 4: Performance of ALCHEMY on different model structures.

DS	Model Structure	T.ACC	G.ACC	T.B.R	G.B.R	T.W.R	G.W.R
MN	CNN1†	98.84%	97.96%	96.94%	94.26%	93.66%	75.94%
	CNN2	99.22%	96.82%	97.44%	92.06%	**94.48%**	75.98%
	ResNet8	98.07%	86.50%	96.58%	78.58%	90.60%	46.08%
	ResNet14	98.92%	79.76%	97.96%	74.26%	94.20%	37.94%
CF	ResNet18	74.40%	92.14%	64.22%	76.90%	53.34%	45.18%
	ResNet34†	73.56%	90.06%	66.26%	80.44%	**58.02%**	60.72%
	W.ResNet34-2‡	61.85%	81.30%	58.90%	76.12%	45.13%	55.44%
	W.ResNet34-5	66.92%	86.22%	63.60%	78.48%	53.97%	58.86%
SV	ResNet18	79.60%	97.82%	57.42%	77.14%	36.06%	44.42%
	ResNet34†	83.90%	94.62%	71.14%	86.70%	50.54%	72.94%
	W.ResNet34-2	81.10%	84.62%	72.21%	69.38%	**57.06%**	49.24%
	W.ResNet34-5	84.59%	93.62%	73.03%	82.80%	45.51%	59.34%
CN	ResNet18	44.38%	67.34%	36.50%	55.58%	30.24%	49.14%
	ResNet34†	59.26%	75.88%	57.44%	72.54%	**39.78%**	34.88%
	W.ResNet34-2	45.72%	58.72%	44.90%	56.58%	29.92%	28.64%
	W.ResNet40-2	44.80%	72.72%	44.08%	70.50%	28.58%	33.30%

† Model structures used in the overall performance setting.
‡ W.ResNet34-2: Wide-ResNet-34-2. W.ResNet34-5: Wide-ResNet-34-5. W.ResNet40-2: Wide-ResNet-40-2.

5.4.1 Impact of max-margin loss. For evaluating the max-marginal loss effect, we conduct the ablation study on the max-marginal loss parameter θ_{mm} and we show the results in Table 2. We adjusted the parameters within the range of 0 to 0.2 and measured the metric outcomes for each parameter. When the parameter is set to 0, it is equivalent to not applying the max-margin loss method. The experimental results indicate that the max-margin method significantly enhances the model's robustness, with the potential to improve by up to 10 percentage points. Additionally, the results also demonstrate that excessively large max-margin loss parameters can lead to a decline in model training performance. After a parameter value exceeds 0.1, the white-box robustness on each dataset starts to decrease. This could be attributed to the excessive emphasis on the target classes of generated samples, which may suppress the diversity of generated samples, among other factors. To be noted, the default θ_{mm} setting we use is 0.1.

We also calculate the inter-class FID (Fréchet Inception Distance) [40] values for the data generated by ALCHEMY without max-margin loss, the data generated by ALCHEMY, the data generated by the diffusion model, and the original data training set on CIFAR-10. As shown in Figure 7, ALCHEMY with max-margin loss generate samples with higher inter-class FID value, which means the samples are further from the boundary, containing more robust features. Without robust generation by max-margin loss, the generated samples are more clustered around the decision boundary with non-robust features, making it easy for the model disruptions under adversarial perturbations. The FID results elaborate how the robust generation benefits the adversarial training.

Observation 4

Robust generation with max-margin loss facilitates the $\mathcal{X}_{sub}$ with robust features, benefiting the adversarial training.

5.4.2 Impact of generation steps. We adjusted the upper limit of generation steps g_{step} and conducted tests. The results are shown in Table 3. The results indicate a noticeable improvement with increased generation steps across the MNIST, CIFAR-10, and CINIC-10 datasets, with a similar trend observed for SVHN. Interestingly, even at 50 generation steps, SVHN exhibited promising results, possibly due to the dataset's inherently lower image clarity. It's worth noting that while increasing generation steps enhances performance, it also prolongs the generation time. Moreover, the performance at 800 steps is comparable to that at 500 steps. Thus, selecting a more suitable value can effectively save time while maintaining training efficacy. The default g_{step} we use is 500.

5.4.3 Impact of teacher-student disagreement. We conduct the ablation study on the teacher-student disagreement on the disagreement parameter θ_{ts}. We elaborate the disagreement results in Figure 5 and Figure 6. From the plots, we observe that employing disagreement results in generated data that better serves adversarial training compared to data generated without disagreement. This improvement enhances the robustness of adversarial training, demonstrating the effectiveness of utilizing teacher-student disagreement in enhancing sample generation. However, similar to the previous

max-margin loss parameter, excessive use of disagreement may lead to a slight performance decrease, as evidenced in the SVHN dataset. The default θ_{ts} we use is 0.50.

5.4.4 Impact of model structure. We show the results of different model structures in Table 4. The detailed setting is elaborated in appendix. From Table 4, Alchemy shows high generalization to several different model structures. For different model structures, the model can all acquire robustness through data-free adversarial training without losing the accuracy on the normal training dataset. However, it is noticeable that the model exhibits varying performance across different datasets. For CINIC-10 and CIFAR-10 datasets, ResNet34 models show the best robustness effect, but for SVHN, Wide-ResNet34-2 is the most robust model, showing the robustness of 57.06%, far better than the rest. This shows that the nature of the dataset itself also influenced the final results. The structure of the model, in addition to affecting the effectiveness of data recovery, also influences the effectiveness of adversarial training on the dataset. Nevertheless, based on the findings in Table A1, it can be inferred that if the original model demonstrates superior performance on the original dataset, it is more likely to exhibit enhanced robustness in data-free adversarial scenarios.

5.5 Robustness against different attack methods

We also measure the adversarial trained models with Alchemy under different attack methods. As shown in Table A2, we measure the popular and general methods widely used in adversarial training. First, we test the gradient-based methods with another step number of PGD-10 (Projected Gradient Descent with a maximum perturbation step of 10). Next, we have optimization-based methods, including AutoPGD (Auto Projected Gradient Descent) [20], DeepFool [62], CW attack [12], and SPSA attack [86]. We show the results in the appendix. We can see that the results of adversarial training demonstrate overall improvements in robustness against various attack methods. This to some extent confirms that our algorithm possesses general robustness improvement, beyond just defending against PGD-20 attacks. We also test experiments on ensemble attacks with Auto Attack [20] and Adaptive Auto Attack and show the results on CIFAR-10 in the appendix. We use the standard setting in Auto Attack and set the perturbation range to different values. These methods employ different optimization techniques to iteratively find the best perturbations that can deceive the model. Within a certain perturbation scope, the robustness accuracy is above 40%, elaborating the robustness improvement against Auto Attack. But adaptive attacks like Auto Attack with large perturbations still introduce a drop in robustness to the model, showing that achieving a robust model oriented towards multiple ensemble and adaptive attacks under data-free conditions needs more sophisticated dataset reconstruction. In our future work, we will continue to explore more effective solutions for more situations.

6 DISCUSSION

6.1 Diffusion as Generator

In recent years, the diffusion model has shown promising results in image generation. It has been used as a powerful data augmentation method in adversarial training [72]. However, as a data augmentation method, the training performance in the absence of original data is extremely poor. Nevertheless, we still believe that data-free adversarial training can be performed with diffusion model. There are many public SOTA diffusion models available on the internet, such as stable-diffusion [73],a latent text-to-image diffusion model capable of generating photo-realistic images given any text input. We attempt to use stable-diffusion-v1-5 [73] to perform data-free adversarial training on CIFAR10. "A photo of {*class name*}" is used as the text prompt to guide the stable-diffusion model to generate images of the corresponding class. In order to generate data that is more relevant to the target dataset, we introduce a well-trained classifier in the denoising process and fine-tune the generated latent variables multiple times through gradients. However, the performance of data-free adversarial training with these generated images is very terrible. The model achieved very high prediction accuracy and robustness on the generated training data, but its generalization capability on the test set was very poor.

In Figure 7, we also measure the inter-class FID [40] values for the data generated by the diffusion model. The inter-class FID of the data generated by the diffusion model is larger than the others. Combined with the experimental results mentioned above, it can be analysed that the diffusion model tends to generate the most representative samples of a given class. However, the diversity of data is not easily guaranteed. In future work, the effect can be improved by adjusting the input more finely, which can be used for data-free adversarial training.

Observation 5

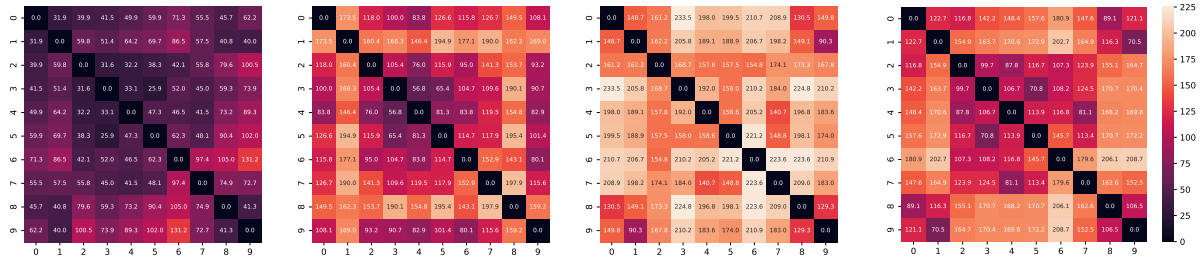
When in the data-free scenario, only utilizing the diffusion model to generate dataset for adversarial training still needs to adjust the input more finely to enhance data diversity.

6.2 Multiple Generators

For most generative tasks, increasing the number of generators is a feasible way to enhance sample diversity and improve the generalization performance of training. Thus, in our research, we also attempted to use multiple models for sample generation. We experimented with 2 to 5 generators, but we found that using multiple generators did not significantly improve our training and adversarial training outcomes. In fact, there was even a decrease in training performance with 4 to 5 generators. Additionally, employing multiple generators significantly increased the training and adversarial training time for our generative models. We speculate that the varying effectiveness of multiple generators in reconstructing the original training dataset and the differences in the distribution of generated samples could be the reasons behind this outcome. Training on data from different distributions poses a challenge of generalization across multiple distributions, requiring the use of different generalization methods. We intend to address this aspect in future research endeavors.

6.3 Efficiency

Compared to regular training, adversarial training itself is time-consuming, and generating samples is also a time-intensive task.



(a) Alchemy w/o robust generation (b) Alchemy (c) Diffusion model (d) Original training dataset

Figure 7: Heatmap of FID score between different classes on CIFAR-10. Each FID value, represented by the depth of color, signifies the distribution difference between two classes. From the results, it can be seen that the classes of generated data are closer when the robust generation is not applied, indicating that the generated samples are closer to the classification boundary and contain a large number of non-robust features. The samples generated by the diffusion model, on the other hand, exhibit a significant difference between classes. The samples generated by Alchemy are the most similar to the relationship between classes in the original dataset.

While this paper has employed some methods to reduce the overall process time, we believe that future research should further explore more efficient adversarial training strategies and sample generation methods to lower the time cost of the entire process. Potential directions could involve improving the convergence speed of adversarial training algorithms, designing more efficient generative models, and utilizing parallel computing to optimize the speed of sample generation. Such efforts will help make the process of adversarial training and sample generation more feasible for real-world applications and more operationally viable for practical deployment.

7 RELATED WORK

7.1 Generative Model for Adversarial Training

Adversarial training of deep neural networks is known to be significantly more data-hungry when compared to standard training. Generative model, which are capable of generating realistic images, is a very powerful data-driven augmentation to improve adversarial robustness. There are a lot of generative models with exquisite design and excellent performance, such as GAN [33], VDVAE [18], DDPM [41]. The generative model is trained with original data, and then a large amount of data is generated by the generative model. Finally, the original data and the generated data are mixed for adversarial training [72]. The diversity and quality of the generated data are critical for improving the adversarial robustness. As a state-of-the-art generative model, generated images from DDPM, improve robustness significantly[72]. So far, which characteristics or standards of the generated data that are most beneficial for adversarial training are still unclear to us.

7.2 Robust Overfitting

Robustness overfitting assumes a catastrophic dimension, engendering an inevitable substantial gap in robust generalization between the model's performance on the training set and that on the test set. Employing early stopping stands out as a straightforward yet impactful method to mitigate the overfitting conundrum encountered in adversarial training, as highlighted in [96]. Insightful research, such as the work by Wu et al. [93], underscores a definitive correlation between the flatness of the weight loss landscape and the discernible robust generalization gap. In addition to early stopping and AWP, alternative approaches have emerged to tackle catastrophic overfitting, including the utilization of activations with low curvature [78] and amplifying the loss associated with low-loss data [97]. These innovative strategies collectively contribute to a more nuanced and comprehensive arsenal for addressing overfitting in the context of adversarial training.

8 CONCLUSION

Our work introduces Alchemy, a novel data-free adversarial training framework designed to enhance model robustness without requiring access to the original training data. Through comprehensive experiments on diverse datasets, Alchemy showcased impressive effectiveness with an average robustness compared to models trained on the original datasets, even in the absence of original training data. The framework not only demonstrated stability and resilience against adversarial attacks in most scenarios but also provided valuable insights for future research directions in data-free adversarial training, particularly in generator selection and parameter settings. This work paves the way for broader access to robust models without the need for original training data, presenting significant potential for enhancing model security across various real-world scenarios.

9 ACKNOWLEDGEMENT

We sincerely thank all the anonymous reviewers for their valuable comments. This work is supported by China NSFC Grant 61925109 and Ant Group. We also thank the undergraduate students Yihui Huang, Yi Xu, Kexu Luo, and Yiran Chen for their assistance. Yanjiao Chen is the corresponding author.

REFERENCES

[1] Mahdieh Abbasi and Christian Gagné. 2017. Robustness to adversarial examples through an ensemble of specialists. *arXiv preprint arXiv:1702.06856* (2017).

[2] Jean-Baptiste Alayrac, Jonathan Uesato, Po-Sen Huang, Alhussein Fawzi, Robert Stanforth, and Pushmeet Kohli. 2019. Are labels required for improving adversarial robustness? *Advances in Neural Information Processing Systems* 32 (2019).

[3] Moustafa Alzantot, Yash Sharma, Supriyo Chakraborty, Huan Zhang, Cho-Jui Hsieh, and Mani B Srivastava. 2019. Genattack: Practical black-box attacks with gradient-free optimization. In *Proceedings of the genetic and evolutionary computation conference*. ACM New York, NY.

[4] Anish Athalye and Nicholas Carlini. 2018. On the robustness of the cvpr 2018 white-box adversarial example defenses. *arXiv preprint arXiv:1804.03286* (2018).

[5] Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*. PMLR.

[6] Muhammad Awais, Fengwei Zhou, Hang Xu, Lanqing Hong, Ping Luo, Sung-Ho Bae, and Zhenguo Li. 2021. Adversarial robustness for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[7] Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. 2021. Recent advances in adversarial training for adversarial robustness. *arXiv preprint arXiv:2102.01356* (2021).

[8] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*.

[9] Nicholas Carlini and Hany Farid. 2020. Evading deepfake-image detectors with white-and black-box attacks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*.

[10] Nicholas Carlini, Florian Tramer, Krishnamurthy Dj Dvijotham, Leslie Rice, Mingjie Sun, and J Zico Kolter. 2022. (Certified!!) adversarial robustness for free! *arXiv preprint arXiv:2206.10550* (2022).

[11] Nicholas Carlini and David Wagner. 2017. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*.

[12] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*.

[13] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. 2019. Unlabeled data improves adversarial robustness. *Advances in Neural Information Processing Systems* 32 (2019).

[14] Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. 2019. Data-free learning of student networks. In *Proceedings of the IEEE/CVF international conference on computer vision*.

[15] Long Chen, Yuchen Li, Chao Huang, Bai Li, Yang Xing, Daxin Tian, Li Li, Zhongxu Hu, Xiaoxiang Na, Zixuan Li, et al. 2022. Milestones in autonomous driving and intelligent vehicles: Survey of surveys. *IEEE Transactions on Intelligent Vehicles* 8, 2 (2022), 1046–1056.

[16] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. 2017. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*.

[17] Yanjiao Chen, Yijie Bai, Richard Mitev, Kaibo Wang, Ahmad-Reza Sadeghi, and Wenyuan Xu. 2021. FakeWake: Understanding and mitigating fake wake-up words of voice assistants. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*.

[18] Rewon Child. 2020. Very deep vaes generalize autoregressive models and can outperform them on images. *arXiv preprint arXiv:2011.10650* (2020).

[19] Francesco Croce and Matthias Hein. 2020. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*. PMLR.

[20] Francesco Croce and Matthias Hein. 2020. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*. PMLR.

[21] Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. 2018. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505* (2018).

[22] Jiangyi Deng, Yanjiao Chen, and Wenyuan Xu. 2022. FenceSitter: Black-box, content-agnostic, and synchronization-free enrollment-phase attacks on speaker recognition systems. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*.

[23] Jiangyi Deng, Fei Teng, Yanjiao Chen, Xiaofu Chen, Zhaohui Wang, and Wenyuan Xu. 2023. V-Cloak: Intelligibility-, naturalness- & timbre-preserving real-Time voice anonymization. In *32nd USENIX Security Symposium (USENIX Security 23)*.

[24] Li Deng. 2012. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine* 29, 6 (2012), 141–142.

[25] Yao Deng, Xi Zheng, Tianyi Zhang, Chen Chen, Guannan Lou, and Miryung Kim. 2020. An analysis of adversarial attacks and defenses on autonomous driving models. In *2020 IEEE international conference on pervasive computing and communications (PerCom)*.

[26] Samuel Dodge and Lina Karam. 2017. A study and comparison of human and deep learning recognition performance under visual distortions. In *2017 26th international conference on computer communication and networks (ICCCN)*. IEEE.

[27] Hang Du, Hailin Shi, Dan Zeng, Xiao-Ping Zhang, and Tao Mei. 2022. The elements of end-to-end deep face recognition: A survey of recent advances. *ACM Computing Surveys (CSUR)* 54, 10s (2022), 1–42.

[28] Gongfan Fang, Jie Song, Xinchao Wang, Chengchao Shen, Xingen Wang, and Mingli Song. 2021. Contrastive model inversion for data-free knowledge distillation. *arXiv preprint arXiv:2105.08584* (2021).

[29] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. 2023. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[30] Jack FitzGerald, Shankar Ananthakrishnan, Konstantine Arkoudas, Davide Bernardi, Abhishek Bhagia, Claudio Delli Bovi, Jin Cao, Rakesh Chada, Amit Chauhan, Luoxin Chen, et al. 2022. Alexa teacher model: Pretraining and distilling multi-billion-parameter encoders for natural language understanding systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[31] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[32] Zhifu Gao, Shiliang Zhang, Ian McLoughlin, and Zhijie Yan. 2022. Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition. *arXiv preprint arXiv:2206.08317* (2022).

[33] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[34] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[35] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision* 129 (2021), 1789–1819.

[36] Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. 2017. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280* (2017).

[37] Chuan Guo, Jacob Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Weinberger. 2019. Simple black-box adversarial attacks. In *International Conference on Machine Learning*. PMLR.

[38] Nezihe Merve Gürel, Xiangyu Qi, Luka Rimanic, Ce Zhang, and Bo Li. 2021. Knowledge enhanced machine learning pipeline against diverse adversarial attacks. In *International Conference on Machine Learning*. PMLR.

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

[40] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems* 30 (2017).

[41] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.

[42] Ruitong Huang, Bing Xu, Dale Schuurmans, and Csaba Szepesvári. 2015. Learning with a strong adversary. *arXiv preprint arXiv:1511.03034* (2015).

[43] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360* (2016).

[44] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. 2018. Black-box adversarial attacks with limited queries and information. In *International conference on machine learning*. PMLR.

[45] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Advances in Neural Information Processing Systems* 32 (2019).

[46] Xiaojun Jia, Yong Zhang, Baoyuan Wu, Ke Ma, Jue Wang, and Xiaochun Cao. 2022. LAS-AT: adversarial training with learnable attack strategy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[47] Wenxin Jiang, Nicholas Synovic, Matt Hyatt, Taylor R Schorlemmer, Rohan Sethi, Yung-Hsiang Lu, George K Thiruvathukal, and James C Davis. 2023. An empirical study of pre-trained model reuse in the hugging face deep learning model registry. *arXiv preprint arXiv:2303.02552* (2023).

[48] Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. 2023. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and individual differences* 103 (2023), 102274.

[49] Zelun Kong, Junfeng Guo, Ang Li, and Cong Liu. 2020. PhysGAN: Generating physical-world-resilient adversarial examples for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[50] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[51] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533* (2016).

[52] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236* (2016).

[53] Chenliang Li, Hehong Chen, Ming Yan, Weizhou Shen, Haiyang Xu, Zhikai Wu, Zhicheng Zhang, Wenmeng Zhou, Yingda Chen, Chen Cheng, et al. 2023. Modelscope-agent: Building your customizable agent system with open-source large language models. *arXiv preprint arXiv:2309.00986* (2023).

[54] Menghan Li, Bin Huang, and Guohui Tian. 2022. A comprehensive survey on 3D face recognition methods. *Engineering Applications of Artificial Intelligence* 110 (2022), 104669.

[55] Bowen Liu, Boao Xiao, Xutong Jiang, Siyuan Cen, Xin He, Wanchun Dou, et al. 2023. Adversarial attacks on large language model-based system and mitigating strategies: A case study on ChatGPT. *Security and Communication Networks* 2023 (2023), 8691095.

[56] Zihao Liu, Qi Liu, Tao Liu, Nuo Xu, Xue Lin, Yanzhi Wang, and Wujie Wen. 2019. Feature distillation: Dnn-oriented jpeg compression against adversarial examples. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[57] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. 2017. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535* (2017).

[58] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* (2017).

[59] EGOR MALYKH. 2016. Tiny ResNet with keras. https://www.kaggle.com/code/meownoid/tiny-resnet-with-keras-99-314.

[60] Juliette Millet, Charlotte Caucheteux, Yves Boubenec, Alexandre Gramfort, Ewan Dunbar, Christophe Pallier, Jean-Remi King, et al. 2022. Toward a realistic model of speech processing in the brain with self-supervised learning. *Advances in Neural Information Processing Systems* 35 (2022), 33428–33443.

[61] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. 2020. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*.

[62] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[63] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*. http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf

[64] Yidong Ouyang, Liyan Xie, and Guang Cheng. 2023. Improving adversarial robustness through the contrastive-guided diffusion process. In *International Conference on Machine Learning*. PMLR.

[65] Qi Pang, Yuanyuan Yuan, Shuai Wang, and Wenting Zheng. 2022. ADI: Adversarial dominating inputs in vertical federated learning systems. *arXiv preprint arXiv:2201.02775* (2022).

[66] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. 2017. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*.

[67] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael P Wellman. 2018. Sok: Security and privacy in machine learning. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*.

[68] Aaditya Prakash, Nick Moran, Solomon Garber, Antonella DiLillo, and James Storer. 2018. Deflecting adversarial attacks with pixel deflection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[69] Zhuang Qian, Kaizhu Huang, Qiu-Feng Wang, and Xu-Yao Zhang. 2022. A survey of robust adversarial training in pattern recognition: Fundamental, theory, and methodologies. *Pattern Recognition* 131 (2022), 108889.

[70] Shilin Qiu, Qihe Liu, Shijie Zhou, and Wen Huang. 2022. Adversarial attack and defense technologies in natural language processing: A survey. *Neurocomputing* 492 (2022), 278–307.

[71] Piyush Raikwar and Deepak Mishra. 2022. Discovering and overcoming limitations of noise-engineered data-free knowledge distillation. *Advances in Neural Information Processing Systems* 35 (2022), 4902–4912.

[72] Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. 2021. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946* (2021).

[73] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-Resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[74] Kevin Roth, Yannic Kilcher, and Thomas Hofmann. 2019. The odds are odd: A statistical test for detecting adversarial examples. In *Proceedings of the 36th International Conference on Machine Learning*. PMLR.

[75] Hadi Salman, Mingjie Sun, Greg Yang, Ashish Kapoor, and J Zico Kolter. 2020. Denoised smoothing: A provable defense for pretrained classifiers. *Advances in Neural Information Processing Systems* 33 (2020), 21945–21957.

[76] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. 2019. Adversarial training for free! *Advances in Neural Information Processing Systems* 32 (2019).

[77] Uri Shaham, Yutaro Yamada, and Sahand Negahban. 2018. Understanding adversarial training: Increasing local stability of supervised models through robust optimization. *Neurocomputing* 307 (2018), 195–204.

[78] Vasu Singla, Sahil Singla, Soheil Feizi, and David Jacobs. 2021. Low curvature activations reduce overfitting in adversarial training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

[79] James Smith, Yen-Chang Hsu, Jonathan Balloch, Yilin Shen, Hongxia Jin, and Zsolt Kira. 2021. Always be dreaming: A new approach for data-free class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[80] Ruoyu Song, Muslum Ozgur Ozmen, Hyungsub Kim, Raymond Muller, Z Berkay Celik, and Antonio Bianchi. 2023. Discovering adversarial driving maneuvers against autonomous vehicles. In *32nd USENIX Security Symposium (USENIX Security 23)*.

[81] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).

[82] Mohammad Mustafa Taye. 2023. Understanding of machine learning with deep learning: Architectures, workflow, applications and future directions. *Computers* 12, 5 (2023), 91.

[83] Florian Tramer. 2022. Detecting adversarial examples Is (nearly) as hard as Classifying Them. In *Proceedings of the 39th International Conference on Machine Learning*. PMLR.

[84] Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. 2020. On adaptive attacks to adversarial example defenses. *Advances in Neural Information Processing Systems* 33 (2020), 1633–1645.

[85] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. 2017. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204* (2017).

[86] Jonathan Uesato, Brendan O'Donoghue, Pushmeet Kohli, and Aaron van den Oord. 2018. Adversarial risk and the dangers of evaluating against weak attacks. In *Proceedings of the 35th International Conference on Machine Learning*. PMLR.

[87] Pratik Vaishnavi, Kevin Eykholt, and Amir Rahmati. 2022. Transferring adversarial robustness through robust representation matching. In *31st USENIX Security Symposium (USENIX Security 22)*.

[88] Fatemeh Vakhshiteh, Ahmad Nickabadi, and Raghavendra Ramachandra. 2021. Adversarial attacks against face recognition: A comprehensive study. *IEEE Access* 9 (2021), 92735–92756.

[89] Lin Wang and Kuk-Jin Yoon. 2021. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE transactions on pattern analysis and machine intelligence* 44, 6 (2021), 3048–3068.

[90] Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. 2021. On the convergence and robustness of adversarial training. *arXiv preprint arXiv:2112.08304* (2021).

[91] Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. 2020. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*.

[92] Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. 2023. Better diffusion models further improve adversarial training. *arXiv preprint arXiv:2302.04638* (2023).

[93] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. 2020. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems* 33 (2020), 2958–2969.

[94] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. 2020. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[95] Ziyi Yin, Muchao Ye, Tianrong Zhang, Tianyu Du, Jinguo Zhu, Han Liu, Jinghui Chen, Ting Wang, and Fenglong Ma. 2023. VLAttack: Multimodal adversarial attacks on vision-language tasks via pre-trained models. *arXiv preprint arXiv:2310.04655* (2023).

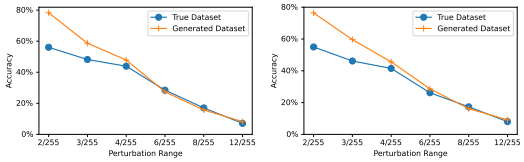
[96] Chaojian Yu, Bo Han, Li Shen, Jun Yu, Chen Gong, Mingming Gong, and Tongliang Liu. 2022. Understanding robust overfitting of adversarial training and beyond. In *International Conference on Machine Learning*. PMLR.
[97] Chaojian Yu, Bo Han, Li Shen, Jun Yu, Chen Gong, Mingming Gong, and Tongliang Liu. 2022. Understanding robust overfitting of adversarial training and beyond. In *Proceedings of the 39th International Conference on Machine Learning*. PMLR.
[98] Shikang Yu, Jiachen Chen, Hu Han, and Shuqiang Jiang. 2023. Data-free knowledge distillation via feature exchange and activation region constraint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
[99] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. 2019. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems* 30, 9 (2019), 2805–2824.
[100] Sergey Zagoruyko and Nikos Komodakis. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146* (2016).
[101] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. 2019. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*. PMLR.
[102] Hui Zhang, Tian Yuan, Junkun Chen, Xintong Li, Renjie Zheng, Yuxin Huang, Xiaojie Chen, Enlei Gong, Zeyu Chen, Xiaoguang Hu, et al. 2022. Paddlespeech: An easy-to-use all-in-one speech toolkit. *arXiv preprint arXiv:2205.12007* (2022).
[103] Jiawei Zhang, Linyi Li, Ce Zhang, and Bo Li. 2023. CARE: Certifiably robust learning with reasoning via variational inference. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*.
[104] Shudong Zhang, Haichang Gao, and Qingxun Rao. 2021. Defense against adversarial attacks by reconstructing images. *IEEE Transactions on Image Processing* 30 (2021), 6117–6129.
[105] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223* (2023).
[106] Zhengli Zhao, Dheeru Dua, and Sameer Singh. 2017. Generating natural adversarial examples. *arXiv preprint arXiv:1710.11342* (2017).
[107] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. 2021. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
[108] Hong Zhu, Shengzhi Zhang, and Kai Chen. 2023. AI-Guardian: Defeating adversarial attacks using backdoors. In *2023 IEEE Symposium on Security and Privacy (SP)*.
[109] Haomin Zhuang, Yihua Zhang, and Sijia Liu. 2023. A pilot study of query-free adversarial attack against stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
[110] Alon Zolfi, Shai Avidan, Yuval Elovici, and Asaf Shabtai. 2021. Adversarial mask: Real-world adversarial attack against face recognition models. *arXiv preprint arXiv:2111.10759* (2021).

Table A1: Original training and adversarial training results of different structure models.

Dataset	Model Structure	Original Results		Original AT Results		
		T.ACC	T.W.R	T.ACC	T.B.R	T.W.R
MNIST	CNN1	99.57%	2.23%	98.56%	98.21%	96.13%
	CNN2	99.58%	2.18%	99.13%	98.74%	**97.57%**
	ResNet8	99.41%	0.00%	96.03%	95.47%	89.43%
	ResNet14	99.67%	1.20%	98.26%	97.91%	94.98%
CIFAR-10	ResNet18	94.00%	0.00%	79.04%	76.76%	58.47%
	ResNet34	93.04%	0.01%	81.71%	79.68%	**60.44%**
	W.ResNet40-2	94.87%	0.01%	67.65%	66.94%	41.64%
	W.ResNet34-5	94.21%	0.00%	78.09%	76.96%	56.74%
SVHN	ResNet18	94.88%	1.88%	91.18%	84.64%	64.57%
	ResNet34	95.00%	1.81%	93.16%	88.55%	66.48%
	W.ResNet40-2	95.80%	0.69%	91.45%	88.83%	68.64%
	W.ResNet34-5	95.96%	0.96%	91.62%	89.90%	**69.29%**
CINIC-10	ResNet18	88.54%	0.03%	66.54%	66.06%	48.02%
	ResNet34	89.64%	0.01%	74.09%	73.54%	**53.97%**
	W.ResNet40-2	88.30%	0.03%	65.85%	65.47%	46.35%
	W.ResNet34-2	83.67%	0.00%	52.73%	52.47%	31.59%

Table A2: The adversarial robust models trained with ALCHEMY evaluation results under different attack methods.

Dataset	Metrics	DeepFool	PGD-10	AutoPGD	C&W	SPSA
MNIST	B.R.	98.59%	96.94%	91.12%	92.23%	92.62%
	W.R.	98.90%	92.60%	85.26%	90.21%	
CIFAR-10	B.R.	72.86%	65.72%	56.02%	64.62%	44.05%
	W.R.	73.88%	57.29%	48.65%	51.76%	
SVHN	B.R.	81.48%	69.31%	62.47%	53.08%	47.07%
	W.R.	83.23%	49.48%	38.24%	43.57%	
CINIC-10	B.R.	59.06%	56.90%	56.43%	62.28%	22.66%
	W.R.	59.34%	39.02%	36.54%	27.06%	



(a) Auto Attack (b) Adaptive Auto Attack

Figure 8: Robustness results of ALCHEMY under Auto Attack and Adaptive Auto Attack with different adversarial perturbation range on CIFAR-10 dataset.

A APPENDIX

A.1 Impact of model structure

We show the results of different model structures in Table 4. We use different structures of pretrained model as the teacher model for data-free adversarial training. For MNIST, we evaluate the performance of ALCHEMY on four different models, denoted as CNN1, CNN2, ResNet8, and ResNet14. CNN1 and CNN2 have 4 and 6 convolutional layers respectively, followed by 2 fully-connected layers. ResNet8 and ResNet14 [59] are two mini-ResNet with depths of 8 and 14, respectively. Different numbers of network layers or depths determine the decision boundaries of classification with different complexity and refinement. Besides, the decision boundary of ResNet is quite different from simple CNN's. Experiments on these types of models can explore the adversarial regularization performance of DFAT for models with decision boundaries of varying complexity. For CIFAR-10, we evaluate ResNet18, ResNet34, Wide-ResNet-34-2, and Wide-ResNet-34-5 as the teacher model structures. Wide-ResNet-34-2 is an extension of the Wide Residual Network (Wide-ResNet) architecture [100], specifically designed for deep neural network models. It consists of 34 layers and employs a widening factor of 2, denoting an increased width of feature maps compared to its parent architecture. Wide-ResNet-34-5 configuration comprises 34 layers and incorporates a widening factor of 5, denoting a substantial increase in the width of the network compared to its base architecture. We elaborate the original model results (without adversarial training) and adversarial training results on the original training dataset. The results are shown in the Table 4 for comparison. All the models show no white-box robustness on the original trained models. We also note the default model structure we used.