TURNING A CURSE INTO A BLESSING: ENABLING DATA-FREE BACKDOOR UNLEARNING VIA STABILIZED MODEL INVERSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Effectiveness of many existing backdoor removal techniques crucially rely on access to clean in-distribution data. However, as model is often trained on sensitive or proprietary datasets, it might not be practical to assume the availability of in-distribution samples. To address this problem, we propose a novel approach to reconstruct samples from a backdoored model and then use the reconstructed samples as a proxy for clean in-distribution data needed by the defenses. We observe an interesting phenomenon that ensuring perceptual similarity between the synthesized samples and the clean training data is *not* adequate to enable effective defenses. We show that the model predictions at such synthesized samples can be unstable to small input perturbations, which misleads downstream backdoor removal techniques to remove these perturbations instead of underlying backdoor triggers. Moreover, unlike clean samples, the predictions at the synthesized samples can also be unstable to small model parameter changes. To tackle these issues, we design an optimization-based data reconstruction technique that ensures visual quality while promoting the stability to perturbations in both data and parameter space. We also observe that while reconstructed from a backdoored model, the synthesized samples do not contain backdoors, and further provide a theoretical analysis that sheds light on this observation. Our evaluation shows that our data synthesis technique can lead to state-of-the-art backdoor removal performance without clean in-distribution data access and the performance is on par with or sometimes even better than using the same amount of clean samples.

1 INTRODUCTION

Deep neural networks have been shown to be vulnerable to backdoor attacks, in which attackers poison training data such that the trained model misclassifies any test input patched with some trigger pattern as an attacker-specified target class (Saha et al., 2020; Li et al., 2020; Zeng et al., 2021). These attacks create a major hurdle to deploying deep networks in safety-critical applications.

Various techniques (Wang et al., 2019; Guo et al., 2019; Liu et al., 2018a) have been developed to remove the effects of backdoor attacks from a poisoned model and turn it into a well-behaved model that does not react to the presence of a trigger. Most of these backdoor removal techniques rely on access to a set of clean samples drawn from the distribution that the poisoned model is trained on. These clean data are needed for synthesizing potential triggers and further fine-tuning the model to let the model unlearn the triggers. However, accessing clean in-distribution samples might not always be feasible. For instance, machine learning models are often trained on proprietary datasets which are not released due to privacy concerns.

There have been a few attempts to lift the requirement on clean in-distribution data, yet suffering unstable performance over different triggers. CLP (Zheng et al., 2022) assumes that the poisoned model contains certain backdoor-related neurons that have a large Lipschitz constant and prunes these neurons to repair the model. However, this assumption does not hold when the trigger induces a large change in the model input. Another line of ideas (Chen et al., 2019) is to reconstruct some data from the model and then use the reconstructed samples as a proxy for clean in-distribution data needed by existing data-reliant defenses. This line has the unique benefit that it can take advantage

of advances of those data-reliant defenses which have already demonstrated remarkable efficacy on various triggers.

In fact, the problem of reconstructing samples from a trained model has been extensively studied in the data privacy literature, known as *model inversion*. The simplest model inversion technique synthesizes an input for a given class by optimizing the likelihood of the model for predicting that class. Running the optimization multiple times with different initialization gives a set of synthesized samples. Chen et al. (2019) utilize this technique to invert samples and further perform backdoor removal. However, the simple model inversion technique is known to fall short in reconstructing high-dimensional input (e.g., RGB images) from a deep neural network and returns a noise-like pattern that does not contain any semantic information about the class. We observe that feeding such reconstructed samples into even the state-of-the-art (data-reliant) backdoor removal technique leads to poor performance. Recently, a series of works has significantly improved model inversion for high-dimensional data (Zhang et al., 2020; Chen et al., 2021; Wang et al., 2021) by performing data synthesis in the latent space of a pre-trained neural network generator. These advanced techniques can often produce synthetic samples that largely retain the class-specific semantics and look perceptually similar to the original training data. A natural question is: Can we improve backdoor removal performance by inserting these higher-quality synthesized samples?

This paper starts by examining the question above. Intriguingly, despite the perceptual similarity between the samples synthesized by these advanced model inversion techniques and the original training data, there is a significant gap in their resulting backdoor removal performance. Particularly, we find two factors that contribute to performance degradation. Firstly, we show that the model predictions at the synthesized samples are unstable to small input perturbations, which misleads downstream backdoor removal techniques to remove these perturbations instead of underlying backdoor triggers. Moreover, unlike clean samples for which the prediction loss of the model converges and thus is stable to local changes on the model parameters, the prediction loss at the synthesized samples is sensitive to small parameter changes. Based on these observations, we introduce a data reconstruction technique that promotes not only perceptual quality but the stability to perturbations in data and parameter space.

In addition, a key question overlooked by existing work that leverages model inversion for backdoor removal is: will the samples reconstructed from a backdoored model contain backdoors? Note that if the synthesized samples for the target class contain triggers, then the existing backdoor removal techniques would be nullified. Empirically, we find that as long as the pre-trained generator leveraged by model inversion is learned from clean data, the reconstructed samples from a poisoned model do not contain triggers. For a commonly used generator in model inversion literature—a generative adversarial network (GAN), we prove that backdoors are not in the range of the generator by analyzing the GAN's equilibrium.

The contributions of the paper are summarized as follows.

- We investigate the connection between model inversion and backdoor removal. We go beyond perceptual quality and reveal the dependence of defense performance on the stability of the inverted samples to input and parameter perturbations. We also provide a theoretical understanding of why pre-trained generator based model inversion does not generate backdoor-contaminated samples.

- We propose a novel bilevel optimization based data reconstruction approach, FRED, which maximizes the stability to input perturbations while encouraging perceptual similarity and the stability to parameter perturbations.

- On a range of datasets and model architectures, employing the synthetic samples produced by FRED can lead to the state-of-the-art data-free backdoor defense performance, which is comparable to or sometimes even better than using the same amount of clean data.

- FRED can be extended to match clean data's features when there is limited access to clean in-distribution samples. In particular, we show that combining just one clean in-distribution point per class with FRED can lead to a better defense performance than directly supplying 20 clean points.

2 PRELIMINARIES

Attacker model. Assume that an attacker performs a backdoor attack against a clean training set D drawn from the distribution $\mathcal{D}$. The attacker injects a set of poisoned samples into D to form a poisoned dataset D_{poi}. We will refer to the model trained on the poisoned dataset as a poisoned model, denoted by $f_{\theta_{\text{poi}}}$. The goal of the attacker is to poison the training set D such that for any clean test input x, adding a pre-defined trigger pattern δ to x will change the output of the trained classifier $f_{\theta_{\text{poi}}}$ to be an attacker-desired target class y_{tar}. A standard technique to poison the dataset is to inject backdoored samples that are labeled as the target class and inject the trigger into their features. The model trained on such a poisoned dataset will learn the association between the trigger and the target class, thereby outputting the target class whenever a test input contains the trigger.

Backdoor Removal. We consider that the defender is given the poisoned model $f_{\theta_{\text{poi}}}$. The goal of the defender is to remove the effects of backdoor triggers from $f_{\theta_{\text{poi}}}$ and obtain a new model f_{θ^*} that is robust to backdoor triggers, i.e., $f_{\theta^*}(x+\delta) = f_{\theta^*}(x)$. Many past backdoor removal techniques (including the state-of-the-art one) are based on the idea of fine-tuning the poisoned model with a set of samples, which will be referred to as the base set; furthermore, past techniques assume that the base set is clean and in-distribution, i.e., each sample there is drawn from $\mathcal{D}$—the distribution generating the clean portion of the data that the poisoned model is trained on. Given the base set $B = \{(x_i, y_i)\}_{i=1}^n$, Zeng et al. (2022) provide a minimax optimization framework that unifies a variety of different backdoor removal techniques (Wang et al., 2019; Chen et al., 2019; Guo et al., 2019):

$$\theta^* = \arg\min_{\theta} \max_{\delta} \frac{1}{|B|} \sum_{i \in B} L(f_\theta(x_i + \delta), y_i), \tag{1}$$

where the inner optimization is aimed at (approximate) *trigger synthesis*, i.e., finding a pattern that causes a high loss for predicting correct labels across all samples in the base set, and the outer optimization performs *trigger unlearning*, which seeks a model that maintains the correct label prediction y_i when the synthesized trigger pattern is patched onto the input x_i. Zeng et al. (2022) proposed I-BAU, which achieves state-of-the-art backdoor removal performance by fine-tuning the poisoned model using mini-batch gradients of the objective in (1). Backdoor removal performance is typically measured by *attack success rate* (ASR), which measures the ratio of the backdoored samples predicted as the target class, and *clean accuracy* (ACC), which measures the ratio of the clean samples predicted as their original class. Despite the promising results, I-BAU shows that the defense performance degrades quickly as the size of available clean in-distribution samples shrinks.

Connection between Data-Free Backdoor Removal and Model Inversion. How to remove backdoors from a given poisoned model without access to clean, in-distribution samples? A natural idea is that as the poisoned model is trained with some clean data, it may memorize the information about the data and therefore one can potentially reconstruct the clean data from the poisoned model. Reconstructing training data from a trained model is intensively studied in the privacy literature, known as *model inversion* (Fredrikson et al., 2014; 2015). To recover training data from a given model f_θ for any class y, the key idea of model inversion is to find an input that minimizes the prediction loss of y:

$$x_{\text{syn}} \in \arg\min_{x} L(f_\theta(x), y) \tag{2}$$

DeepInspect (Chen et al., 2019) solved (2) with gradient descent for multiple times, each of which uses a randomly selected initial value of x; then, the base set was formed by collecting the converged input x_{syn} for each initial value and pairing it with the corresponding label y. However, solving (2) over the high-dimensional space without any constraints generates noise-like features that lack semantic information about corresponding labels. Hence, using the samples synthesized by this way to form the base set gives unsatisfactory backdoor removal performance.

Recently, more advanced model inversion techniques (Zhang et al., 2020; Chen et al., 2021) are proposed to improve the visual quality of synthesized images. Their idea is to optimize over the latent space of a pre-trained GAN:

$$x_{\text{syn}} = G(z^*), \quad z^* \in \arg\min_{z} \underbrace{L(f_\theta(G(z)), y)}_{L_{\text{cl}}(z)} \underbrace{-D(G(z))}_{L_{\text{prior}}(z)} \tag{3}$$

where G and D represent the generator and the discriminator of the GAN, respectively. Zhang et al. (2020) show that the samples synthesized by the GAN-based model inversion technique above can

maintain high visual similarity to the original training data of f_θ. It is natural to ask: Can we apply this more advanced model inversion technique to recover samples from the poisoned model and use them as a substitute for the clean, in-distribution samples needed in backdoor removal? Also, it is critical for the effectiveness of backdoor removal that the target-class samples in the base set do not contain backdoor triggers; otherwise, the trigger unlearning step would reinforce the association between the trigger and the target class, instead of eliminating it. Hence, another critical question is: will model inversion recover backdoor triggers from the poisoned model? We will answer these questions in the following section. And a more detailed discussion of related works on backdoor removal and model inversion can be found in Appendix A.

3 On using model inversion to form the base set

3.1 Visual quality is not enough

We evaluate the performance of I-BAU (Zeng et al., 2022), the state-of-the-art backdoor removal technique, when the underlying base set is formed by the samples synthesized by the pre-trained generator based model inversion technique in Zhang et al. (2020), which will be referred to as generative model inversion (GMI). Specifically, the poisoned model is trained on a traffic sign dataset (Houben et al., 2013). The backdoor attack in Li et al. (2020) is considered and the target class is a randomly chosen class. Figure 1 (a) illustrates the reconstructed samples and the original training data. We find that the samples synthesized by GMI can in general successfully recover the semantics of the clean samples. However, when comparing the backdoor removal performance induced by these synthesized samples with the clean, in-distribution samples of the same size, we find that ACC quickly drops while ASR is two times higher (Figure 1 (b)).

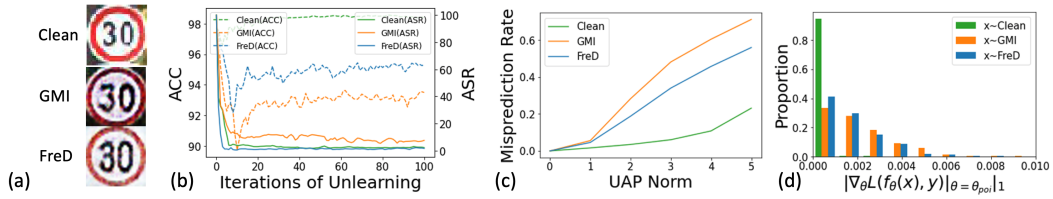

Figure 1: (a) and (b) show the example images and defense performance of the three base sets. (c) is misclassification rate when adding UAP to the three sets respectively. For each set, an optimal UAP is obtained and normalized to 1. We then gradually scale up the three UAPs and test the corresponding misprediction rate. (d) is distribution of samples given its model stability $\|\nabla_\theta L(f_\theta(x), y)|_{\theta=\theta_{\text{poi}}}\|_1$.

Many spurious backdoor triggers exist around the inverted samples. Many backdoor removal techniques, including the state-of-the-art one, rely on trigger synthesis. Recall Eq. 1 which formalizes the backdoor trigger as a universal adversarial perturbation (UAP) that causes a high loss for predicting correct labels across all samples. However, if synthetic samples are sensitive to small UAPs, then these UAPs will be synthesized instead of the actual trigger, leading to poor or unstable defense performance. As shown in Figure 1(c), clean data need to be perturbed with a stronger universal adversarial perturbation (UAP) to reach the same misprediction rate as GMI-synthesized data, implying that clean data is more robust to UAPs.

The poisoned model's prediction on the inverted samples does not exhibit convergence. The poisoned model is usually trained to be optimal on the original training data, meaning that the gradient with respect to the model parameter on the data should be close to zero: $\|\nabla_\theta L(f_\theta(x), y)|_{\theta=\theta_{poi}}\|_1 \approx 0$. However, Figure 1 (d) shows that, while 90% of the clean sample has $\|\nabla_\theta L(f_\theta(x), y)|_{\theta=\theta_{poi}}\|_1 \leq 0.001$, GMI-reconstructed samples distribute more diversely, and the gradient norm based on GMI generated samples are relatively higher.

3.2 Proposed Approach

We propose FreD, an approach to reconstructing the training data from a trained model. FreD differs from recent model inversion techniques in that its synthesis goal not only considers synthetic

data quality and recovery of class-specific semantics, but also addresses the specific challenges of non-converging prediction and small universal perturbation that hinder successful application to backdoor removal.

Specifically, in addition to the two loss terms in (3) that are commonly considered in model inversion literature, we introduce some new loss terms critical to enable the downstream task of backdoor removal. To reconstruct the samples for a given class y from the model $f_{\theta_{\text{poi}}}$, we consider the following loss terms.

- The model-perturbation loss $L_{\text{mp}}(z) = \|\nabla_\theta L(f_\theta(G(z)), y))|_{\theta=\theta_{\text{poi}}}\|_1$, which measures the stability of the prediction for the synthesized sample $G(z)$ to small changes on the parameters of the poisoned model.
- (Optional) The feature consistency loss $L_{\text{con}}(z) = \sum_{(x',y')\in D_{\text{clean}}, y'=y} \|g_{\theta_{\text{poi}}}(G(z)) - g_{\theta_{\text{poi}}}(x')\|_2$, where $g_{\theta_{\text{poi}}}$ represents the feature extractor of the poisoned model $f_{\theta_{\text{poi}}}$, i.e., the output of the penultimate layer. The loss is only used when we extend our approach to the backdoor removal setting where a set of clean in-distribution samples D_{clean} is available. It measures the feature distance between the synthesized sample and the available clean samples.
- The data-perturbation loss $L_{\text{dp}}(z, \delta) = -\text{CosSim}(f_{\theta_{\text{poi}}}(G(z)), f_{\theta_{\text{poi}}}(G(z) + \delta))$, where $\text{CosSim}(\cdot, \cdot)$ stands for cosine similarity. This loss calculates the change of the model output logits when a synthesized sample $G(z)$ is perturbed by δ.

If the synthesized samples are sensitive to small universal perturbation, the trigger synthesis part of backdoor removal (i.e., inner maximization of (1)) will synthesize these small universal perturbations, instead of patterns similar to ground-truth trigger and hence the ground-truth trigger does not get to be unlearned. To improve the effectiveness of backdoor removal, it is critical to maximize the robustness of the synthesized samples to universal perturbations.

We propose a bilevel-optimization algorithm to find the most potent universal perturbation for B synthesized samples:

$$\delta^* = \arg\max_{\theta} \sum_{i=1}^{B} L_{\text{dp}}(z_i^*(\delta), \delta) \tag{4}$$

$$\text{s.t.} \quad z_i^*(\delta) = \arg\min_{z_i} L_{\text{prior}}(z_i) + \lambda_1 L_{\text{cl}}(z_i) + \lambda_2 L_{\text{mp}}(z_i) + \lambda_3 L_{\text{con}}(z_i) + \lambda_4 L_{\text{dp}}(z_i, \delta), \quad \forall i \in \{1, \ldots, B\} \tag{5}$$

However, this bilevel optimization could be computationally expensive as the inner optimization at any δ requires synthesizing a batch of samples. To tackle this challenge, we adopt an online approximation algorithm (Shu et al., 2019; Madaan et al., 2021) to update z and δ alternatively through a single optimization loop. At the output of this online algorithm, we will get both the optimal perturbation and a batch of synthesized samples robustified against the perturbation. The pseudo-code is summarized in Algorithm 1.

λ_1 to λ_4 are the weights associated with each loss to balance their scales in Equation 5. While there are four hyperparameters in the optimization objective, the hyperparameter tuning is lightweight. In our experiments, we choose $\lambda_1 = 1000$ following the prior works on model inversion. We find $\lambda_3 = 1000, \lambda_4 = 1$ work well across different datasets and models. The best value for λ_2 is task-dependant, and for each task, we use grid search to find the best value λ_2 that yields the smallest value of $L_{\text{prior}} + L_{\text{cl}} + L_{\text{mp}} + L_{\text{dp}}$.

4 MI DOES NOT RECOVER BACKDOORS

4.1 EMPIRICAL STUDY

We perform experiments on the GTSRB dataset (Houben et al., 2013) to study whether the synthesized samples would contain backdoors. Specifically, we train a poisoned model under L_0 invisible (L_0 inv) attack (Li et al., 2020), and apply FRED to reconstruct a set of samples from this model. We generate 100 images for each class. To detect whether or not our synthesized data contain the backdoor trigger, we train a binary trigger detection classifier on clean GTSRB training set and its

ALGORITHM 1: Algorithm of FRED.

Input : Generator G, target model T, batch size B, clean data x (optional), max iterations N, learning rate α_1, α_2.

for *each class* $y \in (1, K)$ **do**
  Initialize z: $z^{(1)} \sim \mathbb{N}(0, I)$.
  Initialize δ: $\delta^{(1)} = \mathbf{0}^{1\times d}$ where d is the dimension of synthesized sample $G(z)$.
  for *each iteration* $i \in (1, N)$ **do**
    Temporary Update z: $\hat{z}^{(i)} = z^{(i)} - \alpha_1 \frac{1}{B} \sum_{b=1}^{B} \nabla_z L_{dp}(z^{(i)}, \delta^{(i)}|y, x)$.
    Update δ: $\delta^{(i+1)} = \delta^{(i)} + \alpha_2 \frac{1}{B} \sum_{b=1}^{B} \nabla_\delta L_{dp}(\hat{z}^i, \delta^{(i)})$.
    Update z: $z^{(i+1)} = z^{(i)} - \alpha_1 \frac{1}{B} \sum_{b=1}^{B} \nabla_z L_{total}(z^{(i)}, \delta^{(i+1)}|y, x)$.
  end
  $z_y = z^{(N)}$
end
return $z_1, \ldots, z_K$

poisoned correspondence. The trained trigger classifier has 100% accuracy on a held-out test set. Applying this trigger classifier to our synthesized data, we get that no images are detected to contain the L_0 inv trigger. However, it is still possible that backdoored data are within the support of the GAN generated images but just not discovered by our synthesis technique as the underlying optimization does not directly lead the synthesized data to recover the backdoored samples. To empirically verify whether the backdoored data are not found by our optimization or they are not in the support of the GAN, after regular optimization as shown in 1, we continue to optimize these synthesized data to minimize the mean square error (MSE) between them and their poisoned version. However, even after the optimization, backdoor detection rate is still 0.

Above experiments are all based on the assumption that the auxiliary dataset D_{aux} used for training GAN is clean. What if some poisoned data is mixed into GAN's training data? We further poison D_{aux} with different ratio and test the trigger detection rate respectively. As shown in Figure 3, detection rate remains 0 when using 1% and 2% poison rate, which is typically used for backdoor attacks. Even when the poison rate to an uncommonly large ratio (i.e., 50%), the detection rate remains low at 0.02. Bau et al. (2019) show similar findings of the wholesale omission of GAN on generating some objects. These objects are either complex patterns that include many pixels (e.g., large human figure), or are under-represented in the GAN training distribution.

4.2 THEORETICAL JUSTIFICATION BASED ON EQUILIBRIUM IN GAN

Here, we provide the theoretical justification of why reconstructing samples from a poisoned model based on a (clean) pre-trained GAN does not recover backdoored samples. We require the following two assumptions.

Assumption 1. *The generator G is L-Lipschitz in latent vector input $h \in \mathbb{R}^p$.*

Assumption 2. *For all $h \in \mathbb{R}^p$, if $\|h\| \geq B$ for some $B > 0$, then $G(h)$ has no semantic meaning.*

The Assumption 5 is justified by Figure 3 (b) in Appendix B: We observe an apparent quality degradation of the generated images when increasing the norm of the latent vector h. When the norm is 10^6 larger, the generated images do not contain semantic meaning. Note that both clean and backdoor data are considered to have semantic meaning.

We are interested in whether the range of the generator $\text{range}(G)$ contains backdoored data. Since the backdoored images still have semantic meaning by definition, if $\text{range}(G)$ contains backdoored data they will be within the high-density region where the corresponding latent vector h has $\|h\| \leq B$. By the Lipschitzness (which implies continuity) of G, it means the density of the distribution induced by the generator $(G(h), h \sim \mathcal{N}(0, I))$ on the backdoored data points > 0.[1] Thus, we reduce the question of "*whether the range of G contains backdoored data*" to "*whether the generator distribution has a non-zero density on backdoored data points*."

[1]Note that the set of latent vectors with semantic meaning $\{h : \|h\| < B\}$ is an open set.

We proceed by formulating GAN training as a two-player game between the generator and discriminator. The game terminates only when the two players reach a min-max solution where neither party has the incentive to deviate from the current state. Such a min-max solution is called *pure Nash equilibrium*. Based on the game-theoretic framework, we show the following result, and the proof is outlined in Appendix B.

Theorem 3 (Informal)**.** *When the generator learns a distribution with non-negligible density on backdoored data, the generator and discriminator cannot achieve pure equilibrium.*

This result implies that, no backdoored data point can appear in $\mathrm{range}(G)$ when the GAN is trained properly where the generator and discriminator reaches equilibrium. Thus, no matter how we search over $G(h)$ for different latent vector h during the model inversion step, it is impossible to find an h such that $G(h)$ is a backdoored image.

5 EVALUATION

The primary goal of our evaluation is to assess the effectiveness of FRED to enable data-free backdoor removal. We will also investigate the benefits of using FRED when there is a very limited amount of clean samples available, which alone cannot support effective backdoor removal. In addition, we will go beyond backdoor removal and study the potential application of FRED in adversarial fine-tuning (Jeddi et al., 2020b), where the goal is to robustify a pre-trained model against adversarial examples (Goodfellow et al., 2014). At last, we will perform ablation study on several design choices of FRED, including different loss terms and the number of synthesized samples.

5.1 EXPERIMENTAL SETUP

Data. We evaluate datasets built for different prediction tasks, including face recognition, traffic sign classification, and general object recognition. For each task, we choose two datasets, one used for training the poisoned model and another for learning a pre-trained GAN. Detailed usage of the datasets is shown in Table 5.

Backdoor Attacks. We evaluate nine different kinds of backdoor attacks in all-to-one settings (the target model will misclassify all other classes' samples patched with the trigger as the target class),

including the hidden trigger backdoor attack (Hidden) (Saha et al., 2020), input-aware backdoor (IAB) attack (Nguyen & Tran, 2020), WaNet (Nguyen & Tran, 2021), L_0 invisible (L_0 inv) (Li et al., 2020), L_2 invisible (L_2 inv) (Li et al., 2020), the frequency invisible smooth (Smooth) attack (Zeng et al., 2021), trojan watermark (Troj-WM) (Liu et al., 2018b), trojan square (Troj-SQ) (Liu et al., 2018b), and blend attack (Chen et al., 2017). The implementation details of the attacks are deferred to Appendix C.2.

Baselines. We compare FRED with five baselines, where the first four baselines differ in what kind of samples are contained in the base set and share the same downstream backdoor removal technique, namely, I-BAU, which achieves the state-of-the-art backdoor removal performance given a clean base set. 1) Clean: The base set is formed by clean samples drawn from the original training data of the poisoned model. 2) Out-of-the-distribution (OOD): The base set consists of the OOD samples that are used for learning the pre-trained GAN. 3) Naive: The base set contains samples synthesized by the MI adopted in (Chen et al., 2019) which directly optimizes in the pixel space. 4) GMI: The base set is formed by the synthetic samples from GMI (Zhang et al., 2020). The comparison between FRED and GMI will demonstrate the effectiveness of our designed loss terms. 5) CLP (Zheng et al., 2022): The last baseline is a recent data-free backdoor removal technique that does not utilize the idea of data synthesis. Instead, it prunes the neurons directly based on corresponding Lipschitz constants.

Protocol. We generate 20 samples per class for PubFig and GTSRB; 40 samples per class for CIFAR-10. A detailed study of choosing the number of samples to be generated for each class is shown in Section 5.2. We use the same amount of samples for all the methods for a fair comparison. For the hyperparameters, we fix $\lambda_1 = 1000, \lambda_3 = 1000, \lambda_4 = 1$, set $\lambda_2 = 10$ for PubFig and GTSRB, and $\lambda_2 = 100$ for CIFAR-10. The defense performance is averaged over three random-initialized runs of I-BAU.

	L_0 inv							L_2 inv						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.97	0.98	0.88	0.88	0.93	**0.95**	0.94	0.97	0.98	0.9330	0.95	0.94	**0.94**	0.94
ASR	1.0	0.03	0.02	0.08	0.09	**0.02**	0.02	0.998	0.06	0.832	0.06	0.48	**0.01**	0.01
	Smooth							**Wanet**						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.97	0.98	0.83	0.82	0.96	**0.97**	0.18	0.98	0.94	0.26	0.17	0.81	**0.94**	0.01
ASR	0.998	0.1	0.40	0.05	0.03	**0.02**	0.02	0.99	0.05	0.34	0.97	0.16	**0.05**	0.99
	IAB							**Troj-Sq**						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.94	0.97	0.81	0.45	0.89	**0.91**	0.92	0.98	0.96	0.40	0.45	0.86	**0.94**	0.81
ASR	1.0	0.02	0.10	0.11	0.11	**0.10**	0.10	1.0	0.01	0.06	0.16	0.10	**0.06**	0.23
	Troj-Wm							**Blend**						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.98	0.96	0.47	0.62	0.84	**0.87**	0.87	0.98	0.96	0.47	0.68	0.82	**0.92**	0.75
ASR	1.0	0.01	0.30	0.09	0.30	**0.09**	0.12	1.0	0.08	0.51	0.55	0.48	**0.22**	0.85

Table 1: Results of FRED boosted backdoor unlearning on GTSRB.

5.2 RESULTS

Data-Free Backdoor Defense. Table 5.1 shows that FRED outperforms naive MI, OOD, GMI and CLP against various backdoor attacks on GTSRB. Results for the other datasets can be found in Appendix D and FRED remains the best. Figure 6 visualizes the samples synthesized by Naive, GMI, and FRED. GMI, and FRED can generate samples with better visual quality, whereas Naive generates merely noise-like samples. This visualization explains the significant defense performance improvement achieved by GMI and FRED upon Naive. The performance of FRED is mostly on par with Clean. Interestingly, FRED achieves a higher ACC and comparable ASR than baseline utilizing clean data when defending against the IAB attack performed on the CIFAR-10 dataset. This may be because the model is overfitted to the clean training samples, and samples generated by FRED reduce the degree of overfitting by providing more abundant features. Note that CLP fails defensing against Smooth, Wanet, and Blend attack: Either the ACC drops to close to zero (Smooth and Wanet), or ASR remains high (Wanet and Blend), or both occurs (Wanet). As the above triggers have large norm and hence induce large changes in the model input, CLP's assumption that the poisoned model contains certain backdoor-related neurons have a large Lipschitz constant does not hold. To better interpret the data synthesis process of FRED, we show a series of samples generated at different iterations in Figure 5. We observe that the appearance of the generated images varies significantly over the first ten optimization iterations and stabilizes afterwards.

Data-Free Adversarial Fine-Tuning. Given the promising results on backdoor removal, we consider a related application of the synthesized samples—adversarial fine-tuning, where the goal is to enhance the robustness of a trained model against evasion attacks by fine-tuning with adversarial examples. Specifically, we use FRED-synthesized samples to perform adversarial fine-tuning (FT) (Jeddi et al., 2020a). A detailed experiment setting can be found in Appendix D. We compare FRED with Clean, OOD, Naive, and GMI with the same number of samples, and evaluate the defense performance using two metrics. The first is accuracy on the original, untampered data (Clean Acc); The second is the prediction accuracy under evasion attacks. Table 5.2 presents the results on GTSRB, and we leave the results on CIFAR-10 to Appendix D. Remarkably, FRED outperforms Clean on both clean and robust accuracy. This could be explained by the fact that our specially designed data-perturbation loss facilitates synthesis of larger perturbation during adversarial fine-tuning, hence improving robustness. This observation coincides with the one made in Sehwag et al. (2021) that the synthetic samples from generative models help improve robustness.

Data-Limiated Backdoor Defense. Here, we evaluate the benefits of FRED when there exists a tiny amount of clean samples. Particularly, we consider a stress test with 1 sample. With a single sample, even the state-of-the-art data-reliant backdoor removal technique works poorly as shown in the CIFAR-10 and PubFig83 results in Table 5.2). To evaluate FRED, we use FRED with the proposed feature consistency loss L_{con} to generate 20 additional samples for each class, and the final result (FRED-Booster) is obtained by using the combination of both 1 clean sample and 20 generated samples for each class. Table 5.2 shows that FRED can significantly

	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**
Clean ACC	93.8	91.4	32.3	91.2	91.5	**91.9**
PGD (8/255)	9.0	16.2	9.2	10.1	19.2	**22.8**
PGD (10/255)	4.1	8.8	6.1	5.4	9.2	**14.9**
PGD (16/255)	0.1	1.6	4.0	1.2	1.4	**4.1**
AutoAttack (8/255)	10.0	15.4	8.1	10.2	15.3	**21.7**
AutoAttack (10/255)	4.2	7.4	6.8	5.2	7.2	**14.6**

Table 2: Results of FRED boosted FT on GTSRB. All numbers are accuracies given in %.

boost the defense performance compared to solely using the available clean sample(s). Moreover, using 20 samples from FRED plus one clean sample gives better defense performance than 20 clean samples. As a final note, compared to FRED, CLP, the data-free backdoor removal baseline based on model pruning, cannot be benefited from additional clean samples.

	GTSRB Smooth			CIFAR-10 IAB			PubFig83 Troj-wm		
	Clean(20)	Clean(1)	FRED-Booster	Clean(20)	Clean(1)	FRED-Booster	Clean(20)	Clean(1)	FRED-Booster
ACC	0.98	0.93	**0.98**	0.82	0.52	**0.85**	0.86	0.44	**0.86**
ASR	0.01	1	**0**	0.03	0.18	**0.01**	0.03	0.35	**0**

Table 3: Backdoor unlearning performance with a small amount of clean data and generated samples.

Ablation of Loss Terms. We proposed two loss terms 1) model-perturbation loss L_{mp} and 2) data-perturbation loss L_{dp} to improve the utility of the synthesized samples in the data-free backdoor defense setting. Table 5.2 presents an ablation study of the two losses on a poisoned model trained on GTSRB under the trojan square attack as well as a model trained on Celeba under the trojan watermark attack. We observe that l_{dp} improves the ASR more than l_{mp} while l_{mp} is a more critical driver of maintaining the ACC compared to l_{dp}. This observation aligns with our design objectives. Recall that l_{dp} is designed to enable effective synthesis of backdoor trigger and thus directly related to the reduction of ASR. On the other hand, l_{mp} encourages the stability of prediction to small parameter changes, which in turn mitigates catestrophic forgetting during unlearning; hence, it is directly related to maintaining the clean accuracy.

		Initial	GMI	L_{mp}	L_{dp}	$L_{mp} + L_{dp}$
GTSRB	**ACC(%)**	98.8	86.52	92.70	89.32	94.49
	ASR(%)	100.0	10.06	8.63	8.25	5.90
PubFig	**ACC(%)**	92.21	70.52	72.32	70.61	83.53
	ASR(%)	100.0	26.06	3.25	1.63	2.88

Table 4: Ablation study of proposed model-perturbation loss L_{mp} and data-perturbation loss L_{dp}.

Ablation of the Base Set Size. We study the impact of the number of the synthesized samples used for creating the base set. We choose the number of samples for each class to be $[1, 5, 10, 15, 20, 25, 30]$ and evaluate the defense performance by averaging over 3 runs of the defense. To better interpret the performance of FRED, we also compare with Clean, Naive and GMI using the same amount of samples. Note that this experiment excludes the OOD baseline: we use the poisoned model to generate pseudo-labels for the OOD samples but because the label space of the OOD samples and that of the poisoned model may not overlap, the labeled OOD samples are insufficient or even none for some classes. Figure 2 shows that the defense performance keeps increasing as the number of samples increases and converges to optimum when the number of samples for each class is above 20. FRED, GMI, and Clean can maintain a high ACC when a larger number of generated samples are used, but Naive suffers a significant ACC drop. We also observe during the experiments that performance of Naive has a large variance when evaluated over base set with different size. The variance could induce from the inconsistent quality among generated samples, or instability of the samples against input perturbation, leading to synthesizing inaccurate triggers. On the other hand, the variance of FRED is similar to the variance of Clean, indicating good generalizability of FRED-enabled defenses. Another interesting finding is that when performing unlearning with a small amount of samples (i.e., 1 or 5 per class), FRED even achieves higher ACC than Clean.

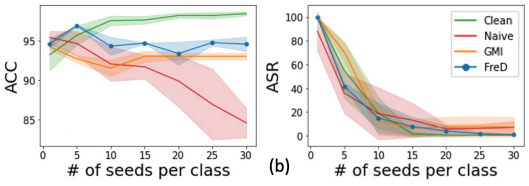

Figure 2: Ablation study of the number of samples used on backdoor defenses on the GTSRB with L_0 inv attack.

6 CONCLUSIONS

We present a FRED to generate synthetic samples that can be used as a substitute for clean data to support backdoor removal. FRED can also be used to boost the defense performance when only limited clean data are available. This work sets a foundation towards developing highly effective data-free backdoor defenses. In particular, one can potentially supply our synthetic data to other future defenses to enable their data-free mode of usage or improve their performance in the limited data setting.

REFERENCES

David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing what a gan cannot generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4502–4511, 2019.

Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728*, 2018.

Huili Chen, Cheng Fu, Jishen Zhao, and Farinaz Koushanfar. Deepinspect: A black-box trojan detection and mitigation framework for deep neural networks. In *IJCAI*, volume 2, pp. 8, 2019.

Si Chen, Mostafa Kahla, Ruoxi Jia, and Guo-Jun Qi. Knowledge-enriched distributional model inversion attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16178–16187, 2021.

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

Edward Chou, Florian Tramèr, and Giancarlo Pellegrino. Sentinet: Detecting localized universal attacks against deep learning systems. In *Deep Learning and Security Workshop*, 2020. URL https://arxiv.org/abs/1812.00292.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pp. 2206–2216. PMLR, 2020.

Bao Gia Doan, Ehsan Abbasnejad, and Damith C Ranasinghe. Februus: Input purification defense against trojan attacks on deep neural network systems. In *Annual Computer Security Applications Conference*, pp. 897–912, 2020.

Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pp. 1322–1333, 2015.

Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An {End-to-End} case study of personalized warfarin dosing. In *23rd USENIX Security Symposium (USENIX Security 14)*, pp. 17–32, 2014.

Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C. Ranasinghe, and Surya Nepal. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*, ACSAC '19, pp. 113–125, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450376280. doi: 10.1145/3359789.3359790. URL https://doi.org/10.1145/3359789.3359790.

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

Wenbo Guo, Lun Wang, Xinyu Xing, Min Du, and Dawn Song. Tabor: A highly accurate approach to inspecting and restoring trojan backdoors in ai systems. *arXiv preprint arXiv:1908.01763*, 2019.

Sebastian Houben, Johannes Stallkamp, Jan Salmen, Marc Schlipsing, and Christian Igel. Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark. In *International Joint Conference on Neural Networks*, number 1288, 2013.

LinLin Huang. Chinese traffic sign database. URL http://www.nlpr.ia.ac.cn/pal/trafficdata/recognition.html.

Ahmadreza Jeddi, Mohammad Javad Shafiee, and Alexander Wong. A simple fine-tuning is all you need: Towards robust deep learning via adversarial fine-tuning. *arXiv preprint arXiv:2012.13628*, 2020a.

Ahmadreza Jeddi, Mohammad Javad Shafiee, and Alexander Wong. A simple fine-tuning is all you need: Towards robust deep learning via adversarial fine-tuning. *arXiv preprint arXiv:2012.13628*, 2020b.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/koh17a.html.

Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). URL http://www.cs.toronto.edu/~kriz/cifar.html.

Shaofeng Li, Minhui Xue, Benjamin Zhao, Haojin Zhu, and Xinpeng Zhang. Invisible backdoor attacks on deep neural networks via steganography and regularization. *IEEE Transactions on Dependable and Secure Computing*, 2020.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*, pp. 273–294. Springer, 2018a.

Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *25nd Annual Network and Distributed System Security Symposium, NDSS*. The Internet Society, 2018b.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 2015.

Divyam Madaan, Jinwoo Shin, and Sung Ju Hwang. Learning to generate noise for multi-attack robustness. In *International Conference on Machine Learning*, pp. 7279–7289. PMLR, 2021.

Anh Nguyen and Anh Tran. Wanet–imperceptible warping-based backdoor attack. *arXiv preprint arXiv:2102.10369*, 2021.

Tuan Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack. *Advances in Neural Information Processing Systems*, 33:3454–3464, 2020.

Nicolas Pinto, Zak Stone, Todd Zickler, and David Cox. Scaling up biologically-inspired computer vision: A case study in unconstrained face recognition on facebook. In *CVPR 2011 WORKSHOPS*, pp. 35–42. IEEE, 2011.

Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 11957–11965, 2020.

Vikash Sehwag, Saeed Mahloujifar, Tinashe Handina, Sihui Dai, Chong Xiang, Mung Chiang, and Prateek Mittal. Robust learning meets generative models: Can proxy distributions improve adversarial robustness? *arXiv preprint arXiv:2104.09425*, 2021.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. *Advances in neural information processing systems*, 32, 2019.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. In *Advances in Neural Information Processing Systems*, pp. 8000–8010, 2018.

Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 707–723. IEEE, 2019.

Kuan-Chieh Wang, Yan Fu, Ke Li, Ashish Khisti, Richard Zemel, and Alireza Makhzani. Variational model inversion attacks. *Advances in Neural Information Processing Systems*, 34, 2021.

Yi Zeng, Won Park, Z Morley Mao, and Ruoxi Jia. Rethinking the backdoor attacks' triggers: A frequency perspective. In *ICCV*, 2021.

Yi Zeng, Si Chen, Won Park, Zhuoqing Mao, Ming Jin, and Ruoxi Jia. Adversarial unlearning of backdoors via implicit hypergradient. In *International Conference on Learning Representations*, 2022.

Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 253–261, 2020.

Runkai Zheng, Rongjun Tang, Jianze Li, and Li Liu. Data-free backdoor removal based on channel lipschitzness. *arXiv preprint arXiv:2208.03111*, 2022.

A Related Works

Backdoor defenses. Backdoor defenses normally can be performed on two levels: data-level and model-level. For data-level detection or cleaning, the defender aims to identify (Gao et al., 2019; Chen et al., 2018; Tran et al., 2018; Koh & Liang, 2017; Chou et al., 2020; Zeng et al., 2021) or purify (Doan et al., 2020) the poison input in the training set, thus requiring the access to training data. Model-level detection or cleaning, instead, aims to detect if a pre-trained model is poisoned or mitigate vulnerabilities of the models. In this paper, we focus on model-level cleaning. Most of the prior works on this line (Wang et al., 2019; Chen et al., 2019; Guo et al., 2019; Zeng et al., 2022) require a small set of clean data to synthesize triggers and further perform unlearning. Among them, I-BAU (Zeng et al., 2022) has achieved the state-of-the-art defense performance against a wide range of existing attacks. However, the performance of I-BAU degrades as the number of clean samples reduces. We aim to enable I-BAU to function effectively without any clean data. A recent work (Zheng et al., 2022) proposes a method to perform backdoor removal without using clean data. They identify model channels with high Lipschitz constants, which are directly calculated from the weight matrices, as backdoor related channels; and do simple pruning to repair the model. However, their method only applies to backdoor scenarios and cannot benefit from clean samples if available. Our method, by contrast, also applies to evasion attacks and is able to leverage available clean samples to further boost defense performance. Above all, the performance of our method is more favorable.

Model Inversion. The goal of model inversion (MI) is similar to ours. But from an attack perspective, MI aims to divulge sensitive attributes in the training data, and to achieve this goal, the generated data should have good visual quality. Fredrikson *et al.* (Fredrikson et al., 2015) follows the maximum likelihood principle and performs model inversion by searching over the image space for a sample with highest likelihood under the given target model. DeepInspect (Chen et al., 2019) employs this simple model inversion to generate a surrogate training set for backdoor unlearning and achieves good results on MNIST and GTSRB. However, we find that samples generated by this naive model inversion approach have bad visual quality and usually fail in downstream defenses on high-dimensional datasets (e.g., PubFig and CIFAR-10). In this paper, we build upon the idea of recent MI works (Zhang et al., 2020; Chen et al., 2021) that search for a synthetic sample in the latent space of a pre-trained GAN instead of the image space. Even when the GAN is not trained on the in-distribution data, this idea can greatly help improve the visual quality of synthesized samples. The key innovations that set our work apart from the MI attacks is that we go beyond the traditional "high-likelihood" assumption made in all existing model inversion works about clean data and further formalize other plausible assumptions, especially those related to data- and model-stability. We show that enforcing the synthetic data to satisfy these assumptions can significantly improve their utility for defenses.

B Why is backdoored data not on GAN's range?

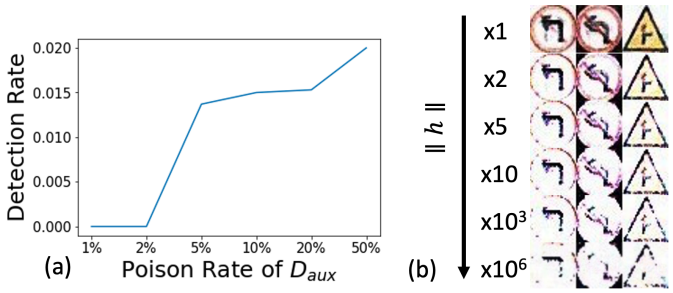

Figure 3: (a) Trigger detection rate when increasing data poison rate of auxiliary dataset D_{aux}. (b) Example of images generated from latent code h with different scales of norm.

Notation. Throughout the section, we use d for the dimension of samples, and p for the dimension of the latent vector. We denote discriminator $D : \mathbb{R}^d \to [0, 1]$, generator $G : \mathbb{R}^p \to \mathbb{R}^d$. We use $\mathcal{P}_{\text{real}}$ to denote the real distribution the GAN aims to learn. The generator G defines a distribution

$\mathcal{P}_G$ as follows: generate latent vector h from p-dimensional spherical standard Gaussian distribution, and then apply G on h and generate a sample $x = G(h)$. We denote the class of discriminators as $\mathcal{D} = \{D\}$ and the class of generators $\mathcal{G} = \{G\}$. Ideally, $\mathcal{D}$ is the class of all 1-Lipschitz functions. For a distribution $\mathcal{P}$, we use $\mathcal{P}(x)$ to denote the density of $\mathcal{P}$ on x, and $\mathcal{P}(S)$ denotes the Lebesgue integration $\int I[x \in S] d\mathcal{P}(x)$. We use $\mathbb{E}_{\mathcal{P}}[D]$ as an abbreviation for $\mathbb{E}_{x\sim\mathcal{P}}[D(x)]$. We use $\text{supp}(\cdot)$ to denote the support of distribution. We use $d_{\mathbb{W}}(\cdot,\cdot)$ to denote 1-Wasserstein distance with ℓ_2-norm, i.e., the Earth Mover distance.

We require the following two assumptions.

Assumption 4. *The generator G is L-Lipschitz in latent vector input $h \in \mathbb{R}^p$.*

Assumption 5. *For all $h \in \mathbb{R}^p$, if $\|h\| \geq B$ for some $B > 0$, then $G(h)$ has no semantic meaning.*

The Assumption 5 is justified by Figure 3 (b): We observe a apparent quality decrease of the generated image When scalping up norm of an optimized seed h. When the norm is $1e^6$ larger, the generated image does not contain semantic meaning. Note that both clean and backdoor data are considered to have semantic meaning.

To formally state our theorem, we formulate the training of GAN as a game between generator and discriminator.

Definition 6 (Payoff)**.** *For a class of generators $\mathcal{G} = \{G\}$ and a class of discriminators $\mathcal{D} = \{D\}$, we define the* payoff $F(D, G)$ *of the game between generator and discriminator as*

$$F(D,G) = \mathop{\mathbb{E}}_{x\sim\mathcal{P}_{\text{real}}}[D(x)] - \mathop{\mathbb{E}}_{x\sim\mathcal{P}_G}[D(x)] \tag{6}$$

The generator and discriminator aims at reaching a min-max solution, i.e., the *pure Nash equilibrium*, where neither party has the incentive to deviate from the current state..

Definition 7 (pure equilibrium)**.** *A pair of strategy (D^*, G^*) a pure equilibrium if for some value V,*

$$\forall D \in \mathcal{D}, F(D, G^*) \leq V$$
$$\forall G \in \mathcal{G}, F(D^*, G) \geq V$$

However, such an equilibrium may not be achievable for a pure strategy setting. We introduce a natural relaxation for quantifying the extent of equilibrium between a pair of generator/discriminator.

Definition 8 (ε-approximate pure equilibrium)**.** *A pair of strategy (D^*, G^*) is an ε-approximate pure equilibrium if for some value V,*

$$\forall D \in \mathcal{D}, F(D, G^*) \leq V + \varepsilon$$
$$\forall G \in \mathcal{G}, F(D^*, G) \geq V - \varepsilon$$

We are now ready to state our main results.

Theorem 9 (Formal)**.** *Given any two distributions $\mathcal{P}_1, \mathcal{P}_2$ s.t. for the set $S_{\text{OOD}} = \{x \in \text{supp}(\mathcal{P}_2) : \min_{y\in\text{supp}(\mathcal{P}_1)\cup\text{supp}(\mathcal{P}_{\text{real}})} \|x - y\| \geq 1\}$, we have $\mathcal{P}_2(S_{\text{OOD}}) \geq 1 - q'$ for some $q' \in [0, 1)$. Let $\mathcal{D}^* = \arg\max_{D\in\mathcal{D}} \mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]$ and $D^* \in \mathcal{D}^*$. If G induce a mixture distribution $\mathcal{P}_G = (1-q)\mathcal{P}_1 + q\mathcal{P}_2$ for some $q \in (0, 1)$, then there exists no $D \in \mathcal{D}$ s.t. (D, G) is ε-approximate pure equilibrium for any $\varepsilon < \frac{1}{2}q(\mathbb{E}_{\mathcal{P}_1}[D^*] - q')$. Moreover, when $\mathcal{P}_{\text{real}} = \mathcal{P}_1$, we have $\varepsilon < \frac{1}{2}q(1 - q')$. Further more, given Assumption 4 and 5, we can lower bound q if $\text{range}(G)$ contains backdoored data, which leads to $\varepsilon < \frac{1}{2}(1 - q')\left(\frac{1}{L\sqrt{2}}\right)^p \frac{\exp(-\frac{1}{2}B^2)}{\Gamma(p/2+1)}$, where Γ is the Gamma function.*

To interpret the above theorem statement, one can regard $\mathcal{P}_1$ as a clean distribution (not necessarily $\mathcal{P}_{\text{real}}$), and $\mathcal{P}_2$ as a distribution that contains backdoor data on its support. Since backdoored images are separated from clean image (i.e., out-of-distribution (OOD) data), we can assume that all backdoored images are within the set $S_{\text{OOD}} = \{x \in \text{supp}(\mathcal{P}_2) : \min_{y\in\text{supp}(\mathcal{P}_1)\cup\text{supp}(\mathcal{P}_{\text{real}})} \|x - y\| \geq 1\}$. The above theorem thus states that no equilibrium could be achieved if $\mathcal{P}_G$ has non-negligible density on S_{OOD}.

Remark. *It is also possible that $\mathcal{P}_1$ also supports on $\{x : \min_{y\in\text{supp}(\mathcal{P}_{\text{real}})} \|x - y\| \geq 1\}$, but this leads to vacuous results.*

B.1 PROOF OF THE FORMAL THEOREM

Lemma 10. *Given any two distributions $\mathcal{P}_1, \mathcal{P}_2$, let $\mathcal{D}^* = \arg\max_{D\in\mathcal{D}} \mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]$. For any $D^* \in \mathcal{D}^*$, if G induce a distribution $\mathcal{P}_G = (1-q)\mathcal{P}_1 + q\mathcal{P}_2$, then there exists no $D \in \mathcal{D}$ s.t. (D, G) is ε-approximate pure equilibrium for any $\varepsilon < \frac{1}{2}q(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*])$.*

Proof. We define an alternative generator G^* s.t. $\mathcal{P}_{G^*} = \mathcal{P}_1$. Given any discriminator D, the payoff gain of G by switching strategy to G^* is

$$F(D,G) - F(D,G^*) \tag{7}$$
$$= (1-q)(\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]) + q(\mathbb{E}_{\mathcal{P}_0}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) - (\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) \tag{8}$$
$$= q(\mathbb{E}_{\mathcal{P}_1}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) \tag{9}$$

Given any discriminator D, the payoff gain of D by switching strategy to D^* is

$$F(D^*,G) - F(D,G) \tag{10}$$
$$= \mathbb{E}_{\mathcal{P}_{\text{real}}}[D^*] - (1-q)\mathbb{E}_{\mathcal{P}_1}[D^*] - q\mathbb{E}_{\mathcal{P}_2}[D^*] - (\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - (1-q)\mathbb{E}_{\mathcal{P}_1}[D] - q\mathbb{E}_{\mathcal{P}_2}[D]) \tag{11}$$
$$= \mathbb{E}_{\mathcal{P}_{\text{real}}}[D^*] - \mathbb{E}_{\mathcal{P}_1}[D^*] + q(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*]) - (\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]) - q(\mathbb{E}_{\mathcal{P}_1}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) \tag{12}$$
$$= d_{\mathbb{W}}(\mathcal{P}_{\text{real}}, \mathcal{P}_1) + q(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*]) - (\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]) - q(\mathbb{E}_{\mathcal{P}_1}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) \tag{13}$$

By Definition 8, (D, G) cannot be ε-approximate equilibrium for

$$\varepsilon < \max\left(F(D,G) - F(D,G^*), F(D^*,G) - F(D,G)\right) \tag{14}$$

since otherwise at least one of D and G will gain more than ε by changing its strategy to D^* or G^*. Therefore, we are interested in lower bounding $\min_D \max\left(F(D,G) - F(D,G^*), F(D^*,G) - F(D,G)\right)$. Note that the minimum can only be achieved when $F(D,G) - F(D,G^*) = F(D^*,G) - F(D,G)$, where we have

$$LHS = q(\mathbb{E}_{\mathcal{P}_1}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) \tag{15}$$
$$= d_{\mathbb{W}}(\mathcal{P}_{\text{real}}, \mathcal{P}_1) + q(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*]) - (\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]) - q(\mathbb{E}_{\mathcal{P}_1}[D] - \mathbb{E}_{\mathcal{P}_2}[D]) \tag{16}$$
$$= RHS \tag{17}$$

and we have

$$2LHS = d_{\mathbb{W}}(\mathcal{P}_{\text{real}}, \mathcal{P}_1) + q(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*]) - (\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]) \tag{18}$$
$$\geq q(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*]) \tag{19}$$

where the last inequality is due to $\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D] \leq \sup_{D\in\mathcal{D}} \mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D] = d_{\mathbb{W}}(\mathcal{P}_{\text{real}}, \mathcal{P}_1)$.

Therefore,

$$\min_D \max\left(F(D,G) - F(D,G^*), F(D^*,G) - F(D,G)\right) \geq \frac{1}{2}q\left(\mathbb{E}_{\mathcal{P}_1}[D^*] - \mathbb{E}_{\mathcal{P}_2}[D^*]\right) \tag{20}$$

□

Remark. *This result may be of independent interest.*

Lemma 11. *Consider the set $S_{\text{OOD}} = \{x \in supp(\mathcal{P}_2) : \min_{y\in supp(\mathcal{P}_1)\cup supp(\mathcal{P}_{\text{real}})} \|x - y\| \geq 1\}$. If $\mathcal{P}_2(S_{\text{OOD}}) \geq 1 - q'$ for some $q' \in [0,1]$, then we have $\mathbb{E}_{\mathcal{P}_2}[D^*] \leq q'$.*

Proof. The value of D on $\text{supp}(\mathcal{P}_2) \setminus (\text{supp}(\mathcal{P}_{\text{real}}) \cup \text{supp}(\mathcal{P}_1))$ does not affect $\mathbb{E}_{\mathcal{P}_{\text{real}}}[D] - \mathbb{E}_{\mathcal{P}_1}[D]$, therefore we only need to ensure that D^* satisfies the Lipschitz assumption. It is easy to see that $D^*(x)$ can be 0 for all $x \in S_{\text{OOD}}$. Since $\mathcal{P}_2(S_{\text{OOD}}) \geq 1 - q'$, we know that $\mathbb{E}_{\mathcal{P}_2}[D^*] \leq q'$. □

Therefore, we know that (G, D) is ε-approximate pure equilibrium only for $\varepsilon \geq \frac{1}{2}q\left(\mathbb{E}_{\mathcal{P}_1}[D^*] - q'\right)$. Moreover, when $\mathcal{P}_{\text{real}} = \mathcal{P}_1$, it is impossible to distinguish between $\mathcal{P}_{\text{real}}$ and $\mathcal{P}_1$ and thus $\mathcal{D}^*$ contains function D that output $D(x) = 1$ for all $x \in \text{supp}(\mathcal{P}_{\text{real}})$. Thus $\varepsilon \geq \frac{1}{2}q\left(1 - q'\right)$.

Now we lower bound q, based on Assumption 4 and 5.

Lemma 12. $q \geq \left(\frac{1}{L\sqrt{2}}\right)^p \frac{\exp(-\frac{1}{2}B^2)}{\Gamma(p/2+1)}$

Proof. Suppose for some $h \in \mathbb{R}^p$ we have $G(h) \in S_{\text{OOD}}$, then for all $\|h' - h\|$ we have $\|G(h') - G(h)\| \leq L\|h' - h\| \leq 1$. Therefore, $G(h') \in \text{supp}(\mathcal{P}_2)$. Therefore, h' within the ball centered at h with radius $1/L$ will all have $G(h') \in \text{supp}(\mathcal{P}_2)$, and thus

$$q \geq \frac{1}{(2\pi)^{p/2}} \exp(-B^2/2) \frac{\pi^{p/2}}{\Gamma(p/2+1)} (1/L)^p = \left(\frac{1}{L\sqrt{2}}\right)^p \frac{\exp(-\frac{1}{2}B^2)}{\Gamma(p/2+1)} \tag{21}$$

□

Plugging this lower bound back to the original bound for ε leads to the final result in Theorem 9.

C Experimental Settings

C.1 Data

Eight datasets built for four different prediction tasks are evaluated in our experiments, including face recognition, traffic sign classification and general object recognition. Detailed usage of the datasets is shown in Table 5.

	Face Recognition	**Traffic Sign Classification**	**General Object Detection**
Poisoned Model	PubFig(Pinto et al., 2011)	GTSRB (Houben et al., 2013)	CIFAR-10 (Krizhevsky et al.)
Pre-trained GAN	CelebA (Liu et al., 2015)	TSRD(Huang)	STL-10 (Coates et al., 2011)

Table 5: Datasets.

C.2 Backdoor Attack Implementation Details

On CIFAR, we adopt test all the nine attacks listed in Section 5. Note that initially, Hidden can only work in one-to-one attack settings where the goal is to fool one class with the trigger, thereby resulting in a low ASR in all-to-one settings. To address this issue, we manually increase the norm bound to $50/255$ with one round of fine-tuning of a pre-trained clean model to achieve an acceptable ASR. However, the ASR of Hidden on GTSRB is still less than 10%, hence, we exclude it from evaluation on GTSRB. On PubFig, we adopt Trojan watermark (Troj-WM), Trojan square (Troj-SQ), and blend attack. The implementations of each attack follow the original works which propose them. The adopted trigger and the target label on each dataset is visualized in Fig. 4.

Figure 4: Datasets and examples of backdoor attacks incorporated. We consider three different datasets in this work: (1) CIFAR-10, (2) GTSRB, and (3) PubFig. Nine different backdoor attack triggers are included in the experimental part as listed. Above, we also show the target label used during the evaluated attacks (e.g., Hidden targeting at label 8 of the CIFAR-10 dataset).

C.3 Adversarial Attacks Implementation Details

Evaluation Metrics. As is customary in the adversarial training literature, we evaluate our techniques against two metrics. The first is accuracy on the original, unaltered data (Clean Acc). The second is accuracy under a PGD based attack, which we call robustness. A robustness value of 0% means every adversarial attack is successful. Note that unlike the metrics in the previous section,

this value is better when higher and worse when lower. As is customary, we consider PGD with $\epsilon = 8/255$ and $\epsilon = 10/255$ - it is well understood that adversarial training on the PGD attack provides robust defenses against many first-order adversarial attacks. Because our paper aims to illustrate a new technique but not provide a novel defense, we set aside new attacks like AutoAttack Croce & Hein (2020) which are designed to mitigate adversarial training.

Attack Settings. We look at models trained on the CIFAR-10 and GTSRB datasets. Our unaltered models - ResNet18 for CIFAR and VGG16 for GTSRB - suffer from accuracy close to 0% when faced with both PGD and AutoAttacks. We do not consider PubFig because the baseline FT approach on the full original PubFig dataset leads to minimal gains in robustness. In fact, even full end-to-end adversarial training from scratch - which would be considered as the "gold standard" to compare against - leads to relatively minor robustness on this dataset. For the FT algorithm, we borrow ideas from the original paper (Jeddi et al. (2020a), specifically gradually increasing the learning rate and then sharply declining. We find that the exact learning rate scheduling proposed in that work does not work for our techniques, so we adjust accordingly.

Baselines. We compare FRED with four baselines: 1) Clean: The base set is formed by clean samples drawn from the original training data of the poisoned model. 2) Out-of-the-distribution (OOD): The base set consists of the OOD samples that are used for learning the pre-trained GAN. 3) Naive: The base set contains samples synthesized by the MI adopted in (Chen et al., 2019) which directly optimizes in the pixel space. 4) GMI: The base set is formed by the synthetic samples from GMI (Zhang et al., 2020). The comparison between FRED and GMI will demonstrate the effectiveness of our designed loss terms.

D FRED BOOSTED DEFENSES RESULTS ON CIFAR-10 AND PUBFIG

Figure 5: Examples of model-specific synthesize by FRED at first 10 iterations for Identity 12 in PubFig dataset. Rightmost in the lower row is the real image of this identity from PubFig.

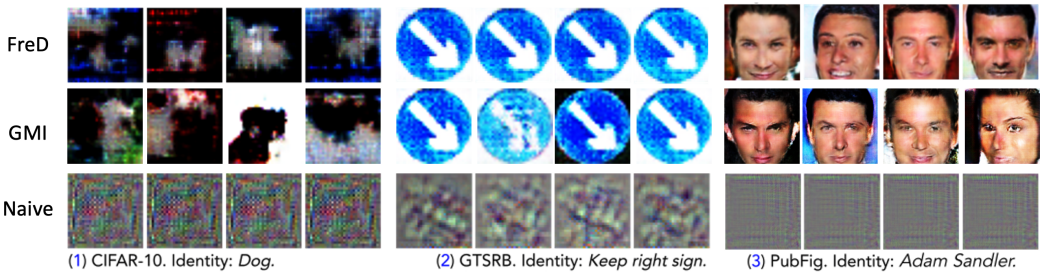

Figure 6: Examples of images obtained by FRED and naive MI. Each subplot represents randomly generated samples for the same class. The upper row shows image generated by FRED and the lower row shows image generated by naive MI.

	L_0 inv							L_2 inv							Smooth						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.93	0.89	0.47	0.44	0.70	**0.76**	0.69	0.94	0.90	0.45	0.87	0.89	**0.90**	0.67	0.93	0.83	0.48	0.54	0.83	**0.83**	0.23
ASR	0.97	0.15	0.07	0.09	0.09	**0.06**	0.06	0.99	0.07	0.12	0.06	0.07	**0.01**	0.05	0.95	0.18	0.02	0.94	0.20	**0.18**	0.86
	Wanet							**IAB**							**Hidden**						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.94	0.91	0.84	0.23	0.79	**0.81**	0.75	0.94	0.82	0.11	0.34	0.84	**0.85**	0.70	0.76	0.89	0.11	**0.66**	0.86	**0.89**	0.35
ASR	0.99	0.01	0.32	0.32	0.03	**0.03**	0.05	0.99	0.03	0	0.08	0.06	**0.05**	0.03	0.88	0.09	0	**0.16**	0.13	**0.09**	0.94
	Troj-Sq							**Troj-Wm**							**Blend**						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.94	0.81	0.51	0.14	0.71	**0.71**	0.71	0.94	0.80	0.50	0.23	0.74	**0.75**	0.66	0.94	0.80	0.85	0.65	0.76	**0.79**	0.46
ASR	1.0	0.02	0.07	0.37	0.28	**0.06**	0.05	1.0	0.04	0.11	0.22	0.02	**0.01**	0.58	0.99	0.05	0.71	0.05	0.06	**0.06**	0.28

Table 6: Results of FRED boosted backdoor unlearning on CIFAR-10.

	Troj-Wm							Troj-Sq							Blend						
	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**	**Initial**	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**	**CLP**
ACC	0.92	0.86	0.06	0.13	0.83	**0.83**	0.01	0.92	0.84	0.02	0.18	0.74	**0.78**	0.01	0.91	0.88	0.06	0.12	0.83	**0.84**	0.03
ASR	1.0	0.03	0.04	0.23	0.10	**0.03**	0.82	1.0	0.04	0.002	0.01	0.06	**0.06**	0.89	1.0	0.44	0.06	0.78	0.82	**0.52**	0.93

Table 7: Results of FRED boosted backdoor unlearning on PubFig.

	Initial	**Clean**	**OOD**	**Naive**	**GMI**	**FreD**
Clean ACC	92.2	85.4	45.3	90.1	90.3	**90.5**
PGD (8/255)	5.2	42.3	13.1	6.0	21.2	**23.6**
PGD (10/255)	3.1	32.8	8.1	4.2	15,8	**18.3**
PGD (16/255)	0.6	13.1	1.2	1.3	9.9	**10.0**
AutoAttack (8/255)	7.3	23.8	12.0	7.2	14.4	**14.9**
AutoAttack (10/255)	4.7	20.1	10.3	5.4	12.1	**12.3**

Table 8: Results of FRED boosted FT on CIFAR-10. All numbers are accuracies given in %.