FAST: Feature Aware Similarity Thresholding for Weak Unlearning in Black-Box Generative Models

Subhodip Panda, Prathosh AP

IEEE Transaction on Artificial Intelligence
(TAI, 2025)

April 15, 2026

Table of Contents

Introduction

Introduction

- Motivation
  1. In order to regulate the output of generative models, machine unlearning emerged as a possible solution.
  2. However, modern *machine unlearning* approaches typically assume access to model parameters and architectural details during unlearning, which is not always feasible.
  3. In a multitude of downstream tasks, these models function as black-box systems, with inaccessible pre-trained parameters, architectures, and training data.
- Objectives
  - *What are the feasible remedies that can effectively suppress the display of undesired outputs when the model is entirely opaque, operating as a black-box system?*
  - In a black-box scenario, filtering where the task involves screening out undesired outputs while retaining other outputs, is a viable practical solution. This research addresses the formidable challenge of effectively filtering undesired outputs generated by black-box GAN.

Contributions

- This work presents a weak unlearning framework to block undesired features within black-box Generative Adversarial Networks (GANs).
- Theoretical analysis establishes that, in the context of black-box generative models, the processes of filtering and weak unlearning are equivalent.
- We introduce an innovative family of filtering mechanisms, collectively referred to as *Feature Aware Similarity Thresholding (FAST)*, meticulously designed to proficiently detect and eliminate outputs featuring undesired attributes while inherently possessing an awareness of the representation of these undesired features.
- We assess and validate the performance of our proposed method across various GAN architectures, including DC-GAN and Style-GAN2, employing datasets such as MNIST and CelebA-HQ, thereby showcasing its effectiveness and versatility in different scenarios and datasets.

Problem Setup

- $\Theta :=$ the space of all parameters of generative models, $\mathcal{X} :=$ the space of all data points, $\mathcal{Z} :=$ space of all latent vectors of generative models
- The pre-trained generative model is denoted as $G_{\theta^{init}}$, with parameters $\theta^{init} \in \Theta$ trained on dataset $D = \{x_i\}_{i=1}^{m}$.
- Based on the outputs of the generative model, the user designates a portion of the data space as undesired, referred to as $\mathcal{X}_f$. Therefore, $\mathcal{X} = \mathcal{X}_r \bigcup \mathcal{X}_f$. We denote the distributions over $\mathcal{X}_f$ and $\mathcal{X}_r$ as $P_{\mathcal{X}_f}$ and $P_{\mathcal{X}_r}$, respectively.
- The objective is to create a mechanism in which the generative model does not produce outputs within the domain $\mathcal{X}_f$.
- Consequently, any retraining mechanism $(\mathcal{R}_{\mathcal{T}})$ results in a new model with parameters θ^r whose output distribution P_{θ^r} seeks to match the distribution $P_{\mathcal{X}_r}$.

FAST Framework

Definition

(Exact Weak Unlearning) Given a retrained generative model $G_{\theta^r}(.)$, we say the model $G_{\theta^u}(.)$ is an exact weak unlearned model iff $\forall z \sim P_{\mathcal{Z}}, \mathcal{O} \subset \mathcal{X}$ the following holds:

$$\Pr(G_{\theta^r}(z) \in \mathcal{O}) = \Pr(G_{\theta^u}(z) \in \mathcal{O}) \tag{1}$$

Definition

((ϵ, δ) Approximate Weak Unlearning) Given a retrained generative model $G_{\theta^r}(.)$, we say the model $G_{\theta^u}(.)$ is an (ϵ, δ) approximate weak unlearned model for a given $\epsilon, \delta \geq 0$ iff $\forall z \sim P_{\mathcal{Z}}, \mathcal{O} \subset \mathcal{X}$ the following conditions hold:

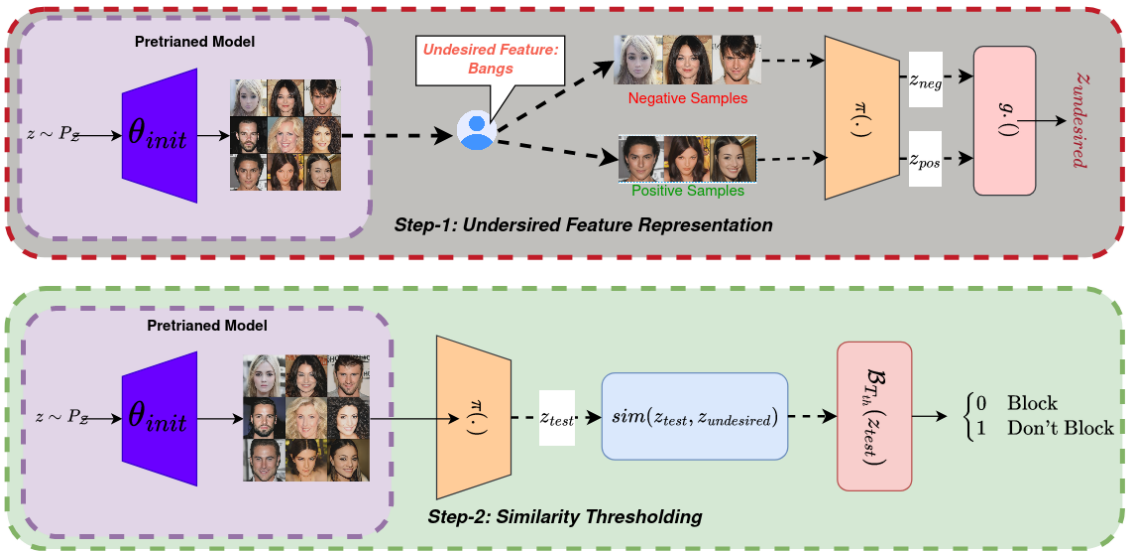
$$\Pr(G_{\theta^r}(z) \in \mathcal{O}) \leq e^{\epsilon} \Pr(G_{\theta^u}(z) \in \mathcal{O}) + \delta \tag{2}$$

$$\Pr(G_{\theta^u}(z) \in \mathcal{O}) \leq e^{\epsilon} \Pr(G_{\theta^r}(z) \in \mathcal{O}) + \delta \tag{3}$$

Theorem (Informal)

If the retrained model follows some regularity constraints, filtering will lead to weak unlearning

Proposed Method



- **Latent Projection Functions (LRFs)**:
  1. *Implicit Latent Space (Imp-LS)* with $\pi(.) = G^{-1}_{\theta^{init}}$.
  2. *Inverted Latent Space (Inv-LS):* Obtained through learning an inverter ($I_{\theta^{init}}$) via reconstruction loss such that $I_{\theta^{init}} : \mathcal{X} \rightarrow \mathcal{Z}_{inv}$.
- **Undesired Representation Functions (URFs)**:
  1. *Mean Difference (MD):*As the difference between the means of latent vectors.
  2. *SVM Normal (Norm-SVM):* As the normal vector of a hyperplane obtained after training an SVM.

Experiments and Results

Table: Accuracy (↑), Recall(↑), AUC(↑), FID (↓) Density(↑) and Coverage(↑) after filtering **MNIST** classes with only **20 pos and 20 neg**. user-feedback

Methods	Class-5						Class-8					
	Acc.	**Rec.**	**AUC**	**FID**	**Dens.**	**Cov.**	**Acc.**	**Rec.**	**AUC**	**FID**	**Dens.**	**Cov.**
Base Classifier	0.64±0.01	0.19±0.01	0.45±0.01	3.60±0.14	0.93±0.01	0.91±0.01	0.70±0.01	0.35±0.02	0.53±0.01	1.42±0.07	**0.99±0.01**	0.99±0.00
Base Classifier + Data Aug.	**0.89±0.00**	0.10±0.00	0.50±0.00	0.80±0.03	0.93±0.00	**0.99±0.00**	**0.94±0.00**	0.10±0.00	0.50±0.00	0.55±0.07	**0.99±0.00**	**1.01±0.00**
Imp-LS + MD	0.60±0.10	**0.69±0.06**	0.69±0.04	2.34±0.97	0.95±0.02	0.96±0.02	0.63±0.05	**0.71±0.03**	**0.72±0.03**	1.78±0.33	0.98±0.01	0.97±0.01
Imp-LS + MD + Latent Aug.	0.61±0.09	**0.69±0.06**	0.69±0.04	2.30±0.94	**0.96±0.02**	0.96±0.01	0.63±0.05	**0.71±0.03**	**0.73±0.03**	1.79±0.36	**0.99±0.01**	0.97±0.01
Imp-LS + Norm-SVM	0.61±0.11	0.67±0.11	0.69±0.03	2.27±1.00	**0.96±0.01**	0.96±0.02	0.62±0.05	0.69±0.04	0.71±0.02	1.90±0.36	**0.99±0.02**	0.97±0.01
Imp-LS + Norm-SVM + Latent Aug.	0.54±0.06	0.56±0.02	0.57±0.05	3.05±0.72	0.92±0.01	0.95±0.01	0.58±0.04	0.60±0.10	0.63±0.06	2.39±0.37	**0.99±0.01**	0.96±0.01
Inv-LS + MD	0.87±0.10	0.11±0.02	0.74±0.04	0.65±0.95	0.93±0.02	**0.99±0.02**	0.92±0.01	0.01±0.01	0.61±0.08	0.64±0.08	**0.99±0.01**	**1.10±0.00**
Inv-LS + MD + Latent Aug.	0.87±0.01	0.12±0.02	0.75±0.03	**0.63±0.97**	0.93±0.02	**0.99±0.01**	0.92±0..02	0.01±0.00	0.59±0.06	0.58±0.08	**0.99±0.01**	0.99±0.01
Inv-LS + Norm-SVM	0.85±0.13	0.55±0.02	**0.83±0.04**	0.72±1.01	**0.96±0.01**	0.97±0.02	0.89±0.06	0.11±0.14	0.68±0.04	0.85±0.33	**0.99±0.00**	0.99±0.01
Inv-LS + Norm-SVM + Latent Aug.	0.79±0.14	0.30±0.02	0.70±0.04	2.22±1.02	0.93±0.01	0.94±0.02	0.92±0.05	0.12±0.14	**0.72±0.05**	**0.48±0.03**	**0.99±0.01**	0.99±0.01

Table: Accuracy (↑), Recall(↑), AUC(↑), FID (↓) Density(↑) and Coverage(↑) after filtering **Celeba-HQ** features with only **20 pos and 20 neg.** user-feedback

Methods	Class-Bangs						Class-Hats					
	Accuracy	Recall	AUC Score	FID	Density	Coverage	Accuracy	Recall	AUC Score	FID	Density	Coverage
Base Classifier	0.41±0.03	0.20±0.01	0.69±0.01	10.58±0.42	**1±0.01**	0.88±0.01	0.64±0.02	0.27±0.01	0.45±0.01	6.37±0.51	**1.08±0.02**	0.99±0.00
Base Classifier + Data Aug.	**0.97±0.01**	0.10±0.00	0.50±0.00	**0.08±0.02**	0.99±0.01	**1±0.00**	0.67±0.01	0.1±0.00	0.5±0.00	6.57±0.04	1.05±0.01	**1±0.00**
Imp-LS + MD	0.78±0.12	0.91±0.05	0.90±0.04	3.71±1.12	0.99±0.02	0.94±0.02	0.76±0.11	0.77±0.06	0.84±0.02	3.55±1.01	0.98±0.01	0.93±0.01
Imp-LS + MD + Latent Aug.	0.77±0.10	0.91±0.07	0.89±0.04	3.69±0.97	0.99±0.02	0.94±0.02	0.77±0.12	0.78±0.06	0.85±0.03	3.50±0.92	0.98±0.01	0.93±0.01
Imp-LS + Norm-SVM	0.78±0.08	**0.93±0.06**	0.92±0.02	2.54±1.00	0.99±0.01	0.96±0.02	0.77±0.07	0.82±0.04	0.86±0.02	3.81±0..93	0.95±0.01	0.93±0.01
Imp-LS + Norm-SVM + Latent Aug.	0.81±0.06	**0.93±0.02**	**0.93±0.05**	2.31±0.72	0.99±0.01	0.97±0.01	0.76±0.05	0.77±0.03	0.84±0.05	3.35±0.65	0.99±0.15	0.94±0.01
Inv-LS + MD	0.71±0.02	0.89±0.01	0.89±0.01	9.91±1.14	0.96±0.01	0.85±0.01	0.78±0.06	**0.86±0.03**	0.88±0.02	9.06±0.72	0.94±0.01	0.83±0.02
Inv-LS + MD + Latent Aug.	0.72±0.02	0.89±0.01	0.89±0.01	9.86±1.04	0.96±0.01	0.85±0.01	**0.79±0.06**	**0.86±0.03**	**0.89±0.02**	9.05±0.70	0.94±0.01	0.83±0.02
Inv-LS + Norm-SVM	0.80±0.01	**0.94±0.02**	**0.93±0.03**	3.61±0.65	**1.01±0.02**	0.96±0.01	0.76±0.05	0.74±0.01	0.85±0.01	5.15±0.81	0.93±0.02	0.90±0.01
Inv-LS + Norm-SVM + Latent Aug.	0.80±0.02	0.92±0.02	**0.93±0.02**	3.83±0.70	1.00±0.01	0.94±0.01	0.78±0.05	0.78±0.01	0.86±0.01	5.48±0.75	0.94±0.02	0.90±0.01

Conclusion

- In this work, we present a methodology to prevent the generation of samples containing undesired features from a pre-trained GAN.
- It is worth mentioning that our method does not assume the availability of the training dataset of the pre-trained GAN so it can generalize to zero-shot settings.
- Despite these advantages, there are some limitations that our methodology can't encompass such as changes in correlated features while unlearning undesired features. Due to high entanglement between the semantics features this kind of impact on other features is visible in the generated outputs