The Dual Power of Interpretable Token Embeddings: Jailbreaking Attacks and Defenses for Diffusion Model Unlearning

Siyi Chen
University of Michigan
siyich@umich.edu

Yimeng Zhang
Amazon
damonzym@amazon.com

Sijia Liu
Michigan State University
liusiji5@msu.edu

Qing Qu
University of Michigan
qingqu@umich.edu

Abstract

Despite the remarkable generation capabilities of diffusion models, recent studies have shown that they can memorize and create harmful content when given specific text prompts. Although fine-tuning approaches have been developed to mitigate this issue by unlearning harmful concepts, these methods can be easily circumvented through jailbreaking attacks. This implies that the harmful concept has not been fully erased from the model. However, existing jailbreaking attack methods, while effective, lack interpretability regarding why unlearned models still retain the concept, thereby hindering the development of defense strategies. In this work, we address these limitations by proposing an attack method that learns an orthogonal set of interpretable attack token embeddings. The attack token embeddings can be decomposed into human-interpretable textual elements, revealing that unlearned models still retain the target concept through implicit textual components. Furthermore, these attack token embeddings are powerful and transferable across text prompts, initial noises, and unlearned models, emphasizing that unlearned models are more vulnerable than expected. Finally, building on the insights from our interpretable attack, we develop a defense method to protect unlearned models against both our proposed and existing jailbreaking attacks. Extensive experimental results demonstrate the effectiveness of our attack and defense strategies.

1 Introduction

Diffusion models (DMs) have recently emerged as a powerful class of generative models, capable of producing diverse and high-quality content such as images [1], videos [2], and protein structures [3]. Notably, Text-to-Image (T2I) diffusion models [4–8] have gained significant popularity for their ability to generate high-fidelity images from user-provided text prompts. However, the remarkable generative capabilities of these models also raise significant concerns regarding their safe deployment. For example, users can exploit carefully crafted text prompts to induce these models by generating unethical or harmful content, such as nude or violent images, or copyrighted material [9].

To address such safety concerns, *Machine Unlearning* (MU) methods have recently been developed for "erasing" harmful concepts from the models while preserving the generation quality of safe content. For instance, a wide range of methods [10–13] seek to unlearn harmful content in pretrained DMs by fine-tuning the model weights [14]. Although these methods have demonstrated notable progress, unlearning DMs through fine-tuning still leaves them vulnerable to *jailbreaking attacks* [15–19], which enforce unlearned models to regenerate harmful content. For instance, UnlearnDiff

[15] crafts adversarial discrete text prompts, and CCE [16] leverages textual inversion [20] to execute jailbreaking attacks in embedding space. These jailbreaking attack methods reveal that existing unlearned models remain vulnerable and can be used to evaluate the robustness of unlearned models. They also highlight the pressing need to address the emerging safety challenge of *defending* unlearned diffusion models, which aims to enhance their robustness against attacks.

However, prior jailbreaking attack approaches rely on discrete or continuous optimization, without considering the interpretability of the resulting attack prompts. Consequently, they offer limited insights into the underlying causes of the deficiencies in current unlearning methods, nor do they explore the potential for defense. To the best of our knowledge, the defense of unlearned models is an underexplored problem in the field. A recent work, RECE [21], targets a specific unlearned model (i.e., UCE [11]), and focuses on defending it against adversarial attacks (i.e., UnlearnDiff). Yet, defending a broader range of unlearned models against other types of attacks remains a challenging problem. This leads us to pose the **question**: *Can we design interpretable and effective jailbreaking attacks, and leverage the resulting insights to develop defenses for existing unlearned models?*

To address the above challenge, we introduce *a subspace attack method* that is interpretable, effective, and transferable, which further motivates an effective *subspace-based defense strategy* applicable to various unlearned models and attacks. Inspired by the hidden-language interpretability of DMs [22], we analyze the token embeddings of the text encoder in unlearned diffusion models, and discover that a diverse set of orthogonal token embeddings can be learned–each capable of regenerating the same harmful concept. These embeddings achieve greater or comparable attacking effectiveness on unlearned models compared to prior methods, while exhibiting stronger transferability across text prompts, initial noises, and unlearned models, establishing them a reliable tool for evaluating model robustness. Importantly, each attack embedding can be expressed as a nonnegative linear combination of interpretable concepts (Sec. 3.1). We leverage this interpretability to uncover how current diffusion unlearning methods continue to associate the harmful concept with mixtures of other concepts, thus retaining unintended generative capabilities. These insights motivate the design of new defense solutions. We propose a concrete defense mechanism that mitigates the harmful concept by removing the learned attack token embeddings through orthogonal subspace projection (Sec. 3.2), and outline additional future directions in App. J. Our defense strategy can be seamlessly integrated into various unlearned models, improving robustness against different jailbreaking attacks while preserving higher generation quality than the baseline defense method [21]. For a comprehensive discussion of related works, see our discussion in **App. A**. In summary, this work makes the following **contributions**:

- **Interpretable jailbreaking attack.** We propose a subspace attack method whose token embeddings can be interpreted in a bag-of-words fashion, revealing that while explicit associations with the target concept are weakened in unlearned diffusion models, implicit associations still persist, providing insights for defending unlearned models.
- **Effective and transferrable attack.** Our attack method consistently achieves strong attack performance across various unlearned models and concepts, providing a reliable metric for evaluating unlearning robustness. Furthermore, these embeddings transfer effectively across initial noise, text prompts, and unlearned models, highlighting the vulnerability of current unlearned models.
- **Subspace defense inspired by subspace attack.** Our investigation into interpretable jailbreaking attacks further motivates a subspace-based defense strategy that mitigates adversarial influence by orthogonally projecting out attack embeddings. This defense approach offers more reliable and flexible protection for unlearned models against diverse jailbreaking attacks, while preserving model utility more effectively than prior defense methods.

2 Preliminaries and Problem Statement

Overview of LDM. T2I diffusion models have recently gained popularity for their ability to generate desired images from user-provided text prompts. Among these various T2I models, Latent Diffusion Model (LDM) [4] is the most widely deployed DM, which current machine unlearning methods majorly focus on. In this work, we first introduce an attack method, and then leverage the insights gained from it to develop a defense strategy. For a given text prompt p, LDM first encodes p using a pretrained CLIP text encoder $f(\cdot)$ to obtain the text embedding $c = f(p)$. Then, the generation process begins by sampling a random noise $z_T \sim \mathcal{N}(0, 1)$ in the latent space. After that, LDM progressively denoises z_T conditioned on the context c until the final clean latent z_0 is achieved. Specifically, for each timestep $t = T, T-1, \ldots, 1$, its denoising UNet, $\epsilon_\theta(z_t \mid c)$, predicts and removes the noise to obtain a cleaner latent representation z_{t-1}. The clean latent z_0 is then decoded
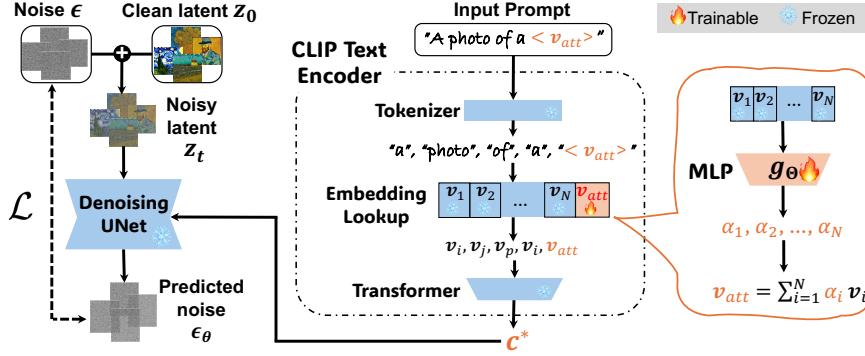
Figure 1: **Learning one interpretable attack token embedding.** The learning process of one attack token embedding v_{att} for the concept "Van Gogh" is visualized. Blue parts represent the frozen unlearned LDM, where, for simplicity, we omit the image encoder and decoder. In orange parts, it illustrates the learning mechanism for optimizing an MLP network to produce v_{att}, which is a linear combination of the existing token embeddings.

to an image with a pretrained image decoder. To train the denoising UNet $\epsilon_\theta(z_t \mid c)$ in LDM, we minimize the denoising error:

$$\mathcal{L} = \mathbb{E}_{(z,c),t,\epsilon \sim \mathcal{N}(0,1)}\left[\|\epsilon - \epsilon_\theta\left(z_t \mid c\right)\|_2^2\right], \tag{1}$$

where z is the clean image latent encoded by a pretrained image encoder and c is its corresponding text embedding encoded by a pretrained CLIP text encoder [23]. Here, $z_t = \sqrt{\alpha_t}z + \sqrt{1-\alpha_t}\epsilon$ is the noisy image latent at timestep t, and $\alpha_t > 0$ is a pre-defined constant.

CLIP text encoder and the token embedding space. To control the generation process, a key component of LDM is the pretrained CLIP text encoder $f(\cdot)$. As illustrated in **Fig. 1**, the CLIP text encoder consists of three main components:

- **Tokenizer:** This module splits the text prompt p into a sequence of tokens, which can be words, sub-words, or punctuation marks. Each token is assigned a unique token ID from the CLIP text encoder's predefined vocabulary.
- **Token Embeddings:** These token IDs [i, j, ...] are then mapped to corresponding token embeddings $v_i \in \mathbb{R}^d$ stored in the token embedding table. This process generates a sequence of token embeddings $[v_i, v_j, \dots]$.
- **Transformer Network:** This network processes the sequence of token embeddings and encodes them into the final text embedding c that can guide the image generation process in LDMs.

Through optimizing Eq. (1), LDM learns to associate activations in the text encoder with concepts in the generated images. Prior research has explored controlling generated content through manipulating activations in the text encoder. In particular, it has been identified that the token embedding space v plays a vital role in content personalization, where a single text embedding can represent a specific attribute [20] and the token embedding space can be utilized for linear decomposition of concepts [22]. Inspired by the expressiveness and interpretability of the token embedding space, this work proposes both jailbreaking attack and defense mechanisms, as detailed in Sec. 3.

Problem statement: jailbreaking attack and defense on unlearned LDMs. Existing MU for LDMs [10, 11, 13] often rely on heuristic fine-tuning of the denoising UNet of LDM, and the resulting models typically *lack* robustness. Jailbreaking attacks aim to evaluate unlearned models' robustness, while defenses aim to improve their robustness under attacks.

Given a prompt p = "a photo of a [*target concept*] ...", an unlearned LDM originally can not generate this target concept. ***Jailbreaking attack*** treats the *unlearned LDM* as a *victim model*, and manipulates the prompt to lead the victim model to regenerate unwanted concepts. There are majorly two kinds of attack setup: (*i*) Adversarial jailbreaking attacks [15, 17–19] optimize an *adversarial text prompt* p_{att} which is appended to p. The combined prompt is then used to trigger the generation of the target concept. (*ii*) CCE [16] follows a different approach by learning an *attack token embedding* v_{att}, which is assigned to a new token, $< v_{att} >$ following [20]. This new token is attached to the existing token set $\{v_i\}$ and the prompt is modified by replacing the [target concept] with $< v_{att} >$ to induce generation of the target concept. Our attack setup is similar to CCE in that it learns attack token embeddings, but it is designed to provide interpretability and achieve stronger attack performance.

"nude"	$\boldsymbol{v}_i$	α_i	"Van Gogh"	$\boldsymbol{v}_i$	α_i	"church"	$\boldsymbol{v}_i$	α_i
	undergraduate	0.08		blu	0.09		eton	0.12
	sumptuous	0.07		staring	0.09		peter	0.09
	hips	0.07		tweeted	0.09		elim	0.09
	shading	0.06		vaugh	0.08		royalwedding	0.09
	nips	0.06		blue	0.08		pilgrimage	0.08
	belle	0.05		tinted	0.07		prayers	0.08

Figure 2: **Interpreting the attack token embeddings for concept "nudity", "Van Gogh", and "church".** Tokens with the largest α_i are words associated with the target concept. For example, top tokens for "church" are activities conducted in the church, or names from the Bible.

Defense, in contrast, seeks to protect an *unlearned LDM* from certain jailbreaking attacks. Once a defense strategy is applied, it should make the model more resistant to regenerating harmful concepts under attack, while preserving its ability to generate harmless content. For example, RECE [21] further modifies the denoising UNet of the unlearned model UCE [11] to defend against UnlearnDiff [15]. In this work, we propose a defense strategy that builds upon existing unlearned LDMs to further enhance their robustness through safeguarding the token embedding space. Our objective is to develop a broadly applicable defense that can be simply integrated into diverse existing unlearned models and remains resilient against jailbreaking attacks having different setups.

Notations. Before introducing our method, we define the following projection operators. Specifically, given vector $\boldsymbol{z}$, for a vector $\boldsymbol{v}$, let $\mathrm{Proj}_{\boldsymbol{v}}(\boldsymbol{z})$ denote the projection of $\boldsymbol{z}$ onto $\boldsymbol{v}$. For a matrix $\boldsymbol{V}$, let $\mathrm{Proj}_{\boldsymbol{V}}(\boldsymbol{z})$ denote the projection of $\boldsymbol{z}$ onto the subspace spanned by the columns of $\boldsymbol{V}$. Formally, these operators are given by

$$\mathrm{Proj}_{\boldsymbol{v}}(\boldsymbol{z}) := \frac{\boldsymbol{v}\boldsymbol{v}^\top}{\|\boldsymbol{v}\|_2^2}\boldsymbol{z}, \ \ \mathrm{Proj}_{\boldsymbol{V}}(\boldsymbol{z}) := \boldsymbol{V}(\boldsymbol{V}^\top\boldsymbol{V})^{-1}\boldsymbol{V}^\top\boldsymbol{z}.$$

3 Subspace Attacking and Defending Methods

This section introduces our subspace attacking and defending methods for LDMs. In Sec. 3.1, we explore the token embedding space to develop an interpretable and transferable attack method (SubAttack) by learning a sequence of attack token embeddings that form a low-dimensional subspace. SubAttack reveals the vulnerability of unlearned models and inspires us to propose a defense strategy (SubDefense) in Sec. 3.2, by orthogonal subspace projection of learned attack token embeddings, which can effectively defend against various jailbreaking attacks.

3.1 Subspace Attacking: *SubAttack*

Before we introduce our subspace attacking (SubAttack) method, let us build some intuition of how to learn a single-token embedding attack $\boldsymbol{v}_{\mathtt{att}} \in \mathbb{R}^d$ first. Based on this, we will then show how to iteratively learn a sequence of orthogonal attack token embeddings through *deflation*, i.e., removing already computed embeddings.

3.1.1 A Single-Token Embedding Attack

Specifically, inspired by [22], we learn a single-token embedding $\boldsymbol{v}_{\mathtt{att}} \in \mathbb{R}^d$ through a non-negative linear representation of existing token embeddings $\boldsymbol{v}_i$ in the CLIP vocabulary $\mathcal{V}$ as follows:

$$\boldsymbol{v}_{\mathtt{att}} = \sum_{i=1}^{N} \alpha_i \boldsymbol{v}_i, \ \ \alpha_i = g_{\boldsymbol{\Theta}}(\boldsymbol{v}_i) \geq 0, \tag{2}$$

where N is the total size of the original CLIP vocabulary, and $\boldsymbol{v}_i, i = 1, 2, \ldots, N$, are original CLIP token embeddings within $\mathcal{V}$. Non-negative α_i are parameterized via a multi-layer perceptron (MLP) network $g_{\boldsymbol{\Theta}}(\cdot) : \mathbb{R}^d \mapsto \mathbb{R}^+$ with ReLU activation.

To learn $\boldsymbol{v}_{\mathtt{att}}$, we optimize the loss $\mathcal{L}$ in (1) with respect to the parameter $\boldsymbol{\Theta}$ of the MLP, while freezing all the other components. As illustrated in **Fig. 1**, during training we enforce the training data pairs $(\boldsymbol{z}, \boldsymbol{c}^*) \sim \mathcal{D}$ to satisfy the following constraints: (*i*) $\boldsymbol{z}$ is the latent image containing the target harmful concept. (*ii*) $\boldsymbol{c}^*$ is the text embedding for the text prompt $\boldsymbol{p}$, and $\boldsymbol{p}$ contains the new special token $< \boldsymbol{v}_{\mathtt{att}} >$ whose token embedding is $\boldsymbol{v}_{\mathtt{att}}$.

Remarks. The optimized $\boldsymbol{v}_{\mathtt{att}}$ is the "hidden word" within the unlearned LDM representing the target concept, and prompts such as "a photo of $< \boldsymbol{v}_{\mathtt{att}} >$" can trigger the unlearned model to

(a) SD	(b) ESD	(c) FMN	(d) UCE	(e) SPM

Figure 3: **Interpreting the subspace of attack token embeddings for concept "nudity" across different models.** (a) The original LDM (i.e., SD) majorly relates it to **explicit** synonyms. (b-e) Unlearned LDMs more heavily associate it with **implicit** concepts.

regenerate the target concept. Adopting the constraint in (2) following [22] further enables the "hidden word" to be interpretable in a bag-of-words manner. The target concept can be viewed as a combination of top-weighted (i.e., having largest α_i) concepts in $\mathcal{V}$. As illustrated in **Fig. 2**, our method can identify human-interpretable tokens that are associated with different harmful concepts (e.g., nudity, Van Gogh, church) in the unlearned LDM. This aligns with prior findings [24] that "negative concepts are not as interpretable as positive concepts." We discuss such interpretability in a broader context, including the LLM linear representation hypothesis [25] in App. A. Moreover, we will ultimately learn an orthogonal set of "hidden words" for the target concept, and the set of words can be interpreted together to more comprehensively understand how the unlearned models still memorize the target concept, as shown in **Fig. 3**, with a more detailed discussion in Sec. 4.2. Now, we first introduce how such a set of "hidden words" are learned.

3.1.2 Subspace Token Embedding Attacks

Instead of learning a single attack token embedding $\boldsymbol{v}_{\text{att}}$, it is more powerful to learn a set of diverse attacks $\{\boldsymbol{v}_{\text{att},k}\}_{k=1}^{K}$ ($m \leq d$) that can attack the same harmful concept, as outlined in **Algorithm 1**. We enforce orthogonality on $\{\boldsymbol{v}_{\text{att},k}\}_{k=1}^{K}$ to increase the diversity of the attacks so that they span a low-dimensional subspace.

We learn such a set of orthogonal token embeddings $\{\boldsymbol{v}_{\text{att},k}\}_{k=1}^{K}$ through deflation, sharing similar ideas with classical methods in numerical linear algebra such as orthogonal matching pursuit [26]. Specifically, suppose the first attack token embedding $\boldsymbol{v}_{\text{att},1}$ is identified following Sec. 3.1.1 by optimizing a MLP g_{Θ_1}, we then "remove" the harmful concept $\boldsymbol{v}_{\text{att},1}$ from the whole vocabulary $\mathcal{V}$ via orthogonal projection:

$$\boldsymbol{v}_{i,2} \;=\; \boldsymbol{v}_{i,1} - \text{Proj}_{\boldsymbol{v}_{\text{att},1}}(\boldsymbol{v}_{i,1}), \quad \forall\, i \in [N]. \tag{3}$$

Here, for the ease of presentation, we let $\boldsymbol{v}_{i,1} = \boldsymbol{v}_i \in \mathcal{V}$ for all $i \in [N]$. (3) makes *all* the updated $\boldsymbol{v}_{2,i}, \ldots, \boldsymbol{v}_{2,N}$ orthogonal to $\boldsymbol{v}_{\text{att},1}$. With the new $\mathcal{V}_2 = \{\boldsymbol{v}_{2,i}\}_{i=1}^{N}$, we can learn a second attack token embedding $\boldsymbol{v}_{\text{att},2} = \sum_{i=1}^{N} \alpha_{i,2} \boldsymbol{v}_{i,2}$, $\alpha_{i,2} = g_{\Theta_2}(\boldsymbol{v}_{i,2}) \geq 0$, then $\boldsymbol{v}_{\text{att},2}$ **is ensured to be orthogonal to** $\boldsymbol{v}_{\text{att},1}$. Here, g_{Θ_2} is another MLP optimized in the same way as g_{Θ_1}. As such, we can repeat the procedure for K times to learn and construct a set of orthogonal attack token embeddings $\{\boldsymbol{v}_{\text{att},k}\}_{k=1}^{K}$ for the same harmful concept. In practice, during attacking, we choose $K = 5$, which already has a strong attack performance while keeping the optimization efficient.

3.2 Subspace Defending: *SubDefense*

Our SubAttack reveals that combinations of related hidden concepts can represent the target concept in an unlearned LDM through a linear composition. This insight motivates us to design a defense strategy within the same linear framework. Our intuition is to remove these identified concept representations from unlearned models through orthogonal projection, thereby making them more robust to various jailbreaking attacks. Concretely, because linearly composed concepts become more difficult to recover, this is achieved by projecting onto the null space of the learned subspace attacks.

More specifically, suppose we have learned a set of attack token embeddings $\{\boldsymbol{v}_{\text{att},k}\}_{k=1}^{K}$ for a target concept through SubAttack outlined in Sec. 3.1, then let us rewrite

$$\boldsymbol{V}_{\text{att}} = [\boldsymbol{v}_{\text{att},1} \quad \boldsymbol{v}_{\text{att},2} \quad \cdots \quad \boldsymbol{v}_{\text{att},K}] \in \mathbb{R}^{d \times K}.$$

This $\boldsymbol{V}_{\text{att}}$ is learned in an unlearned diffusion model whose CLIP token embedding vocabulary is $\mathcal{V} = \{\boldsymbol{v}_i\}_{i=1}^{N}$. The proposed defense will "block" the subspace spanned by $\boldsymbol{V}_{\text{att}}$ through orthogonal
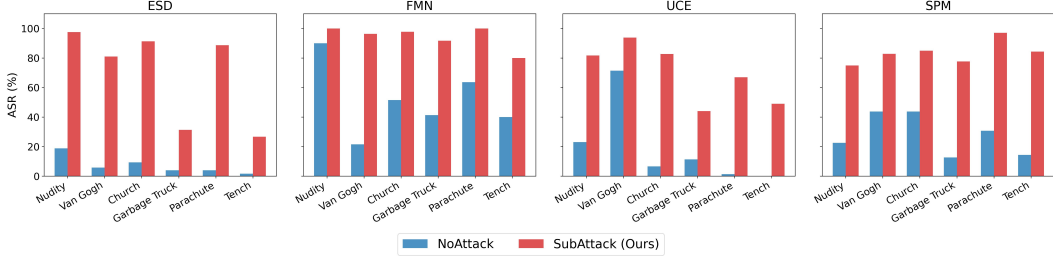
Figure 4: **SubAttack jailbreaks various concepts (NSFW, style, objects) across different unlearned models (ESD, FMN, UCE, SPM).** It consistently reveals the residual vulnerabilities in these models.
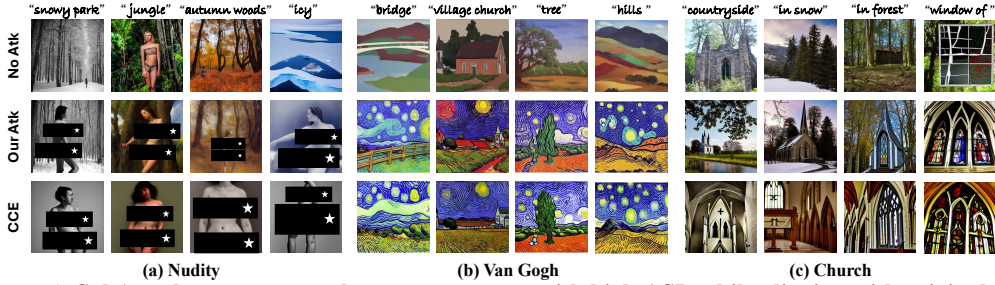

Figure 5: **SubAttack can generate the target concepts with high ASR while aligning with original text prompts**. For example, our attack generates nude women with different backgrounds while CCE fails to generate the correct backgrounds.

projection. Each token embedding v_i in $\mathcal{V}$ will be updated as follows:

$$v_{\texttt{def},i} = v_i - \text{Proj}_{V_{\texttt{att}}}(v_i), \quad \forall\, i \in [N]. \tag{4}$$

For *UnlearnDiff* [15] and *SubAttack*, their learned jailbreaking attack prompts or embeddings are based on the unlearned LDM's vocabulary. Hence, we will update the unlearned LDM by applying Eq. (4) to complete the defense. After that, ***new*** UnlearnDiff and SubAttack attacks can take place on the updated model, but have lower ASR (Sec. 5). For *CCE* [16], which learns an attack token embedding $v_{\texttt{att}}$ with no constraints related to the unlearned LDM's vocabulary $\mathcal{V}$, simply applying Eq. (4) is not enough. Hence, additionally, for ***new*** $v_{\texttt{att}}$ learned by CCE, $v_{\texttt{def}} = v_{\texttt{att}} - \text{Proj}_{V_{\texttt{att}}}(v_{\texttt{att}})$ is applied. In SubDefense, we name K as the number of blocked tokens.

4 Experiments for SubAttack

This section demonstrates that SubAttack is an effective tool for exposing and understanding the vulnerabilities of existing unlearned models. Through extensive experiments, we highlight both the effectiveness and transferability of SubAttack, and leverage its interpretability to provide insights into why current unlearning methods fail.

4.1 Settings

(*i*) **Victim Models.** The domain of diffusion model unlearning is undergoing rapid advancement. To assess the proposed SubAttack, we choose several unlearned LDMs that are widely used in prior jailbreaking attack methods [15, 16], including ESD [10], FMN [12], and UCE [11], together with a recent unlearned model SPM [13]. These unlearned models are capable of unlearning not-safe-for-work (NSFW) concepts, styles, or objects, and perform well on standard unlearning benchmarks while preserving reasonable generation ability. Following [15], the unlearned models used in this work are finetuned from Stable Diffusion (SD) v1.4 [4]. (*ii*) **Concepts and Dataset.** We perform jailbreaking attacks on three categories of concepts commonly targeted in unlearned LDMs: "nudity" for NSFW concept, "Van Gogh" for style concept, and "church", "garbage truck", "parachute", and "tench" for object concept. To facilitate reproducibility, we follow the dataset construction protocol of UnlearnDiff [15], creating for each concept a set of 300-900 (text prompt, seed) pairs. Each pair is verified to produce the target concept with the original SD v1.4. Our dataset is approximately six times larger than that used in UnlearnDiff, enabling more reliable evaluation. Moreover, for each prompt, we construct at least 10 (text prompt, seed) pairs using different seeds to reduce randomness and support the evaluation of attack transferability across different noise initializations. (*iii*) **SubAttack Setup.** By default, we conduct SubAttack to learn $\{v_{\texttt{att},k}\}_{k=1}^{K}$ with $K = 5$ for each concept. For

6

Table 1: **Attack performance of various jailbreaking methods**, measured by ASR (%) over 900 prompts for each concept across various unlearned models, average computation time for attacking one image, and other features. Best results are highlighted in **bold**.

	ASR (%) ↑												Time per Image (s) ↓	Interp-retable	Inspire Defense
Concepts:	Nudity				Van Gogh				Church						
Victim Models:	ESD	FMN	UCE	SPM	ESD	FMN	UCE	SPM	ESD	FMN	UCE	SPM			
NoAttack	18.78	90.00	23.00	22.56	5.78	21.56	71.44	43.78	9.33	51.56	6.55	43.78	NA	NA	NA
UnlearnDiff	51.11	**100.00**	78.22	**83.33**	40.94	**100.00**	**100.00**	53.49	51.74	35.33	61.67	53.67	906.6	✗	✗
CCE	85.11	98.33	77.22	78.33	75.22	93.33	95.67	81.67	82.00	**97.78**	81.89	76.67	**11.4**	✗	✗
SubAttack (Ours)	**97.56**	**100.00**	**81.67**	74.89	**81.00**	96.33	98.33	**82.78**	**91.33**	**97.78**	**82.67**	**84.89**	54.2	✓	✓

each (text prompt, seed) pair, we perform the attack by replacing the target concept word in the prompt with each $v_{\mathtt{att},k}$. The attack is considered successful if at least one of the $v_{\mathtt{att},k}$ leads to the generation of the target concept. We choose $K = 5$ as it provides strong attack performance while maintaining computational efficiency. Ablations on attack performance versus K are in App. F.1. (*iv*) **Metrics.** Following [15], we utilize pretrained image classifiers to examine whether the target concept is generated in the image, and report attack success rate (ASR). For NSFW concept, NudeNet [27] is employed. For style concept, we use the publicly available classifier finetuned on the WikiArt dataset [28] and report Top-3 since it can better represent the attack results, considering the classifier is overly restrictive as discussed in [15]. For objects, an ImageNet-pretrained ResNet-50 classifier is deployed. (*iv*) **Baselines.** We compare SubAttack with three baselines: NoAttack, UnlearnDiff, and CCE, where NoAttack refers to using the original prompts on unlearned models without specific jailbreaking techniques. By default, UnlearnDiff and CCE are implemented following their original settings, but unified using our dataset. For example, UnlearnDiff will optimize an adversarial attack prompt for each <text prompt, seed> pair. We provide more experiment details in App. C.1.

4.2 On the Effectiveness, Transferability, and Interpretability of SubAttack

SubAttack is an effective global attack. Unlearndiff is a local attack by optimizing an adversarial text prompt for a (prompt, seed) pair. This could be time-consuming since attacking each (prompt, seed) pair takes about 30 minutes on average. In contrast, our SubAttack aligns with CCE and can learn global attack token embeddings to attack any (prompt, seed) pairs, where the learning of each global token embedding takes about 20 minutes on average. As presented in **Fig. 4**, SubAttack's global attack token embeddings learned on different unlearned models can jailbreak various concepts across hundreds of different prompts and seeds. Selecting "nudity", "Van Gogh", and "church" as representative concepts, we compare ASR of SubAttack with baselines in **Tab. 1**. Although, as a local attack, UnlearnDiff performs worse than CCE and SubAttack in many scenarios, such as attacking any unlearned model for the concept "church". Although CCE learns the attack token embedding freely while our SubAttack adds additional constraints to enable interpretability, SubAttack is compatible with CCE, and surpasses CCE in many circumstances. Moreover, as illustrated in **Fig. 5**, our attack follows the text prompts better. For example, our attack fits the nude woman into different backgrounds, such as snowy parks, jungles, and woods, while CCE overly emphasizes "nudity". We provide additional attack visualizations in App. I.

SubAttack can transfer across different unlearned models. The attack token embeddings identified by SubAttack demonstrate strong transferability, even across different unlearned diffusion models. As shown in **Fig. 6 (a)**, embeddings

Table 2: **Transfer attack performance of various jailbreaking methods** from ESD to other models across different concepts, measured by ASR (%).

Concepts:	Nudity			Van Gogh			Church		
Victim Models:	FMN	UCE	SPM	FMN	UCE	SPM	FMN	UCE	SPM
NoAttack	90.00	23.00	22.56	21.56	71.44	43.78	51.56	6.55	43.78
UnlearnDiff	93.33	41.33	38.22	12.78	64.00	47.11	6.19	13.33	58.00
CCE	93.00	18.33	37.56	72.33	43.56	81.33	91.00	70.11	**92.78**
SubAttack (Ours)	**96.89**	**77.00**	**80.44**	**72.67**	**88.89**	**86.89**	**92.89**	**83.77**	92.00

learned via SubAttack on the ESD model are directly transferred to attack FMN, SPM, and UCE. All three concept types, nudity, style, and object, can be successfully transferred to these target models with high ASR. We further compare the transfer ASR of SubAttack against other baselines in **Tab. 2** (more results in Tab. 10 in App. D.2), where we transfer the token embeddings from CCE and the adversarial prompts from UnlearnDiff to other victim models accordingly. SubAttack consistently achieves the highest transfer ASR across different models and concepts. This strong transferability suggests that the learned attack embeddings may either emerge from shared distributional patterns introduced during fine-tuning or be inherited from the original SD model, with our following analysis supporting the latter.

SubAttack token embeddings are inherited from the original SD. We experimentally verify that the learned token embeddings are effective in the original SD. Specifically, we transfer the attack token embeddings from different vic-

tim models (ESD, FMN, UCE, and SPM) back to the original SD, and test their transfer ASR on SD. The transfer ASR turns out to be high, consistently being larger than 80% across all different concepts and models (details in **Tab. 5** in Appendix). We visualize the transfer results in **Fig. 6 (b)**. These

Figure 6: **Transfer attack** token embeddings learned by SubAttack to different unlearned models or to the original diffusion model.

results demonstrate that although effective on unlearning benchmarks, existing machine unlearning methods still preserve certain associations of the target concept that are inherited from the original SD. These inherited associations are likely a key reason unlearned models continue to generate harmful content. Leveraging the interpretability of our method, we subsequently uncover the nature of these residual associations.

SubAttack reveals concepts implicitly associated with the target concept. As shown in **Fig. 2**, one learned token embedding can be interpreted in a bag-of-words manner. Furthermore, the set $\{v_{\text{att},k}\}_{k=1}^{K}$ can be collectively analyzed as in **Fig. 3**. Setting $K = 100$, for each $v_{\text{att},k}$, we extract the top 50 highest-weighted tokens, resulting in 5,000 tokens per victim model. These tokens are stemmed and lemmatized to root forms, and the most frequent root tokens are visualized using WordCloud. The same process is applied to the original SD. As depicted in Fig. 3a, the most frequent stems in the original SD are "nude" and its direct synonyms, such as "bare" and "naked". In contrast, Fig. 3b, Fig. 3d, and Fig. 3e reveal that the top tokens in ESD, UCE, and SPM are instead implicitly related terms such as "slave", "nip", and "babes". This indicates that while these unlearned models reduce explicit associations with the target concept, they still retain implicit associations. Interestingly, this mirrors human associative thinking. Besides, FMN displays a higher presence of explicit terms like "nude" (see Fig. 3c) and produces more nudity-related images than other unlearned models, even with no attacks (see Fig. 4). This supports the notion that weaker unlearning leads to retained explicit associations as well. Additional results and analysis are provided in App. D.1.

5 Experiments for SubDefense

Having demonstrated the effectiveness of our attack method, we now turn to evaluating the defense mechanism it motivates. This section evaluates the robustness of SubDefense by integrating it into existing unlearned models. Experimental results demonstrate that SubDefense provides a more versatile and resilient defense strategy than the baseline, while better preserving generation quality on safe prompts.

5.1 Settings

(*i*) **Basics.** SubDefense is plugged into UCE, ESD, FMN, and SPM for concepts "nudity", "Van Gogh", and "church" using our constructed dataset by default. To compare with baseline RECE, we apply Sub-Defense with 20 blocked tokens. In all other cases, we use the default setting of 100 blocked tokens.

(*ii*) **Metrics.** To assess defense effectiveness, various jailbreaking attacks are conducted before and after applying defenses, and the corresponding ASR is reported. SubAttack with $K = 5$ is used consistently before and after defense to ensure a fair comparison. Addition-

Figure 7: **Defending UCE** using RECE or SubDefense across various concepts.

ally, the generative quality of the defended unlearned models is evaluated on the MSCOCO-10k dataset [29, 30] using FID and CLIP scores [31]. Further details are provided in App. C.2.

5.2 Performance of SubDefense

SubDefense surpasses the defense baseline. Defending unlearned diffusion models against jailbreaking attacks remains a largely underexplored area—particularly against strong attacks such as

Table 3: **SubDefense is more robust than baseline RECE in defending three concepts on UCE against UnlearnDiff or our SubAttack, while preserving better generative quality.**

Metrics:	UnlearnDiff ASR ↓		SubAttack ASR ↓		COCO-10k FID ↓		COCO-10k CLIP ↑	
Scenarios:	SubDefense	RECE	SubDefense	RECE	SubDefense	RECE	SubDefense	RECE
Nudity	**73.55%**	76.44%	**34.11%**	62.44%	**17.51**	17.57	**30.70**	30.07
Van Gogh	**52.78%**	61.67%	**29.44%**	84.44%	**16.64**	17.11	**30.94**	30.08
Church	**39.78%**	50.78%	**5.22%**	80.33%	**17.41**	17.41	**30.86**	30.07

CCE. Recently, RECE [21] was proposed to defend UCE from adversarial attacks like UnlearnDiff and serves as our baseline. We compare SubDefense with RECE in defending UCE against both UnlearnDiff and our SubAttack in **Tab. 3**. SubDefense achieves lower ASR under both attacks, demonstrating its superior robustness. Moreover, it attains lower FID and higher CLIP scores on COCO-10k, indicating better preservation of generative quality. Visualization results for RECE and

SubDefense are shown in **Fig. 7**. Visualizations on image generation quality are provided in **Fig. 8**, and additional results are provided in App. H, where we also verify SubDefense's ability to retain generation quality for safe concepts related to the removed harmful ones.

Figure 8: **Safe image generation** after applying RECE or SubDefense.

SubDefense can defend unlearned models against various attacks. Taking ESD and "nudity" as an example, **Tab. 4** shows that SubDefense is effective against a wide range of jailbreaking attacks. While different attack methods impose distinct constraints when learning adversarial prompts or token embeddings, they all depend on the unlearned model's residual ability to generate the target concept. By disrupting this capability through "hidden words" removal, SubDefense can reduce the ASR of multiple attack types, taking a step toward a more versatile defense strategy. However, we observe that NoAttack, UnlearnDiff, and SubAttack achieve lower

Table 4: **SubDefense can defend ESD against different kinds of attacks.**

Metrics:	Nudity ASR				CLIP	FID
	NoAttack	UnlearnDiff	CCE	SubAttack		
ESD	18.11%	51.11%	85.11%	97.56%	30.13	18.23
ESD+SubDefense	0.0%	4.56%	75.67%	42.33%	29.58	19.20

ASR than CCE after defense. This suggests that the current defense is less effective against CCE, a challenging problem remaining underexplored in the literature. We provide a more detailed analysis of defenses against CCE in App. F.2, showing that blocking more tokens improves robustness but comes at the cost of reduced utility. Designing effective defense strategies against CCE while preserving model utility is a promising direction for future research. Additionally, extended results on other datasets (e.g., I2P dataset for NSFW concept) and unlearned models (e.g., FMN, SPM) are available in App. E.2 and App. E.3.

6 Conclusion

This paper introduces a new jailbreaking attack method that learns token embeddings capable of effectively guiding unlearned diffusion models to regenerate harmful concepts. As an interpretable method, it reveals that there still remains a large and diverse subspace within unlearned diffusion models. The subspace embeds the target concept with human-interpretable words that are implicitly associated with it. The proposed attack exhibits strong transferability across text prompts, noise inputs, and unlearned models, underscoring critical limitations in current unlearning approaches, which are more vulnerable than previously assumed. Leveraging the interpretability and diversity of the attack, we design a plug-and-play defense mechanism that can be integrated into existing unlearned models to defend against various jailbreaking attacks while maintaining generation quality. In summary, our findings introduce a novel attack strategy that highlights the pressing need for more robust unlearning techniques, and propose a new defense approach that enhances the safety of generative diffusion models, offering actionable insights for future research.

References

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

[2] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15954–15964, 2023.

[3] Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Ryan J. Ragotte, Laura F. Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 618(7962):512–518, 2023.

[4] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[5] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *ArXiv*, abs/2204.06125, 2022.

[6] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[7] Yimeng Zhang, Tiancheng Zhi, Jing Liu, Shen Sang, Liming Jiang, Qing Yan, Sijia Liu, and Linjie Luo. Id-patch: Robust id association for group photo personalization. *arXiv preprint arXiv:2411.13632*, 2024.

[8] Yihua Zhang, Yimeng Zhang, Yuguang Yao, Jinghan Jia, Jiancheng Liu, Xiaoming Liu, and Sijia Liu. Unlearncanvas: A stylized image dataset to benchmark machine unlearning for diffusion models. *arXiv e-prints*, pages arXiv–2402, 2024.

[9] Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[10] Rohit Gandikota, Joanna Materzyńska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. In *Proceedings of the 2023 IEEE International Conference on Computer Vision*, 2023.

[11] Rohit Gandikota, Hadas Orgad, Yonatan Belinkov, Joanna Materzyńska, and David Bau. Unified concept editing in diffusion models. *IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024.

[12] Eric Zhang, Kai Wang, Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Forget-me-not: Learning to forget in text-to-image diffusion models, 2023.

[13] Mengyao Lyu, Yuhong Yang, Haiwen Hong, Hui Chen, Xuan Jin, Yuan He, Hui Xue, Jungong Han, and Guiguang Ding. One-dimensional adapter to rule them all: Concepts, diffusion models and erasing applications. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[14] Thanh Tam Nguyen, Thanh Trung Huynh, Zhao Ren, Phi Le Nguyen, Alan Wee-Chung Liew, Hongzhi Yin, and Quoc Viet Hung Nguyen. A survey of machine unlearning, 2024.

[15] Yimeng Zhang, Jinghan Jia, Xin Chen, Aochuan Chen, Yihua Zhang, Jiancheng Liu, Ke Ding, and Sijia Liu. To generate or not? safety-driven unlearned diffusion models are still easy to generate unsafe images... for now. *European Conference on Computer Vision (ECCV)*, 2024.

[16] Minh Pham, Kelly O. Marshall, Niv Cohen, Govind Mittal, and Chinmay Hegde. Circumventing concept erasure methods for text-to-image generative models. In *The Twelfth International Conference on Learning Representations*, 2024.

[17] Zhi-Yi Chin, Chieh-Ming Jiang, Ching-Chun Huang, Pin-Yu Chen, and Wei-Cheng Chiu. Prompting4debugging: red-teaming text-to-image diffusion models by finding problematic prompts. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

[18] Yu-Lin Tsai, Chia-Yi Hsu, Chulin Xie, Chih-Hsun Lin, Jia You Chen, Bo Li, Pin-Yu Chen, Chia-Mu Yu, and Chun-Ying Huang. Ring-a-bell! how reliable are concept removal methods for diffusion models? In *The Twelfth International Conference on Learning Representations*, 2024.

[19] Haomin Zhuang, Yihua Zhang, and Sijia Liu. A pilot study of query-free adversarial attack against stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2385–2392, June 2023.

[20] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2023.

[21] Chao Gong, Kai Chen, Zhipeng Wei, Jingjing Chen, and Yu-Gang Jiang. Reliable and efficient concept erasure of text-to-image diffusion models, 2024.

[22] Hila Chefer, Oran Lang, Mor Geva, Volodymyr Polosukhin, Assaf Shocher, Michal Irani, Inbar Mosseri, and Lior Wolf. The hidden language of diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024.

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[24] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[25] Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Causal Representation Learning Workshop at NeurIPS 2023*, 2023.

[26] Joel A. Tropp and Anna C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.

[27] Platelminto. NudeNetClassifier: A classifier for nsfw content detection. `https://github.com/platelminto/NudeNetClassifier`, 2024. Accessed: 2025-05-09.

[28] Babak Saleh and Ahmed Elgammal. Large-scale classification of fine-art paintings: Learning the right metric on the right feature, 2015.

[29] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

[30] Yimeng Zhang, Xin Chen, Jinghan Jia, Yihua Zhang, Chongyu Fan, Jiancheng Liu, Mingyi Hong, Ke Ding, and Sijia Liu. Defensive unlearning with adversarial training for robust concept erasure in diffusion models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[31] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *ArXiv*, abs/2104.08718, 2021.

[32] Huiwen Chang, Han Zhang, Jarred Barber, Aaron Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Patrick Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 4055–4075. PMLR, 23–29 Jul 2023.

[33] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *ArXiv preprint arXiv:2310.04378*, 2023.

[34] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, page 89–106, Berlin, Heidelberg, 2022. Springer-Verlag.

[35] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *ArXiv preprint arXiv:2204.06125*, 2022.

[36] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation. *Transactions on Machine Learning Research*, 2022. Featured Certification.

[37] Xingqian Xu, Zhangyang Wang, Eric Zhang, Kai Wang, and Humphrey Shi. Versatile diffusion: Text, images and variations all in one diffusion model, 2024.

[38] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022.

[39] Zhi-Yi Chin, Chieh-Ming Jiang, Ching-Chun Huang, Pin-Yu Chen, and Wei-Chen Chiu. Prompting4debugging: Red-teaming text-to-image diffusion models by finding problematic prompts. In *International Conference on Machine Learning (ICML)*, 2024.

[40] Yijun Yang, Ruiyuan Gao, Xiaosen Wang, Tsung-Yi Ho, Nan Xu, and Qiang Xu. Mma-diffusion: Multimodal attack on diffusion models, 2024.

[41] Antonio A. Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. *Making AI forget you: data deletion in machine learning*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[42] Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. In *ICCV*, 2023.

[43] Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *The Twelfth International Conference on Learning Representations*, 2024.

[44] Chi-Pin Huang, Kai-Po Chang, Chung-Ting Tsai, Yung-Hsuan Lai, Fu-En Yang, and Yu-Chiang Frank Wang. Receler: Reliable concept erasing of text-to-image diffusion models via lightweight erasers. In *ECCV*, page 360–376, Berlin, Heidelberg, 2024. Springer-Verlag.

[45] Alvin Heng and Harold Soh. Selective amnesia: A continual learning approach to forgetting in deep generative models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[46] Yuchen Yang, Bo Hui, Haolin Yuan, Neil Gong, and Yinzhi Cao. Sneakyprompt: Jailbreaking text-to-image generative models, 2023.

[47] Natalie Maus, Patrick Chao, Eric Wong, and Jacob Gardner. Black box adversarial prompting for foundation models, 2023.

[48] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *The Eleventh International Conference on Learning Representations*, 2023.

[49] Yong-Hyun Park, Mingi Kwon, Jaewoong Choi, Junghyo Jo, and Youngjung Uh. Understanding the latent space of diffusion models through the lens of riemannian geometry. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[50] Siyi Chen, Huijie Zhang, Minzhe Guo, Yifu Lu, Peng Wang, and Qing Qu. Exploring low-dimensional subspace in diffusion models for controllable image editing. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[51] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023.

[52] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. SVDiff: Compact Parameter Space for Diffusion Fine-Tuning . In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7289–7300, Los Alamitos, CA, USA, October 2023. IEEE Computer Society.

[53] Huijie Zhang, Jinfan Zhou, Yifu Lu, Minzhe Guo, Liyue Shen, and Qing Qu. The emergence of reproducibility and consistency in diffusion models, 2024.

[54] Christopher Olah, Ludwig Schubert, and Alexander Mordvintsev. Feature visualization. *Distill*, 2017.

[55] Thomas FEL, Victor Boutin, Louis Béthune, Remi Cadene, Mazda Moayeri, Léo Andéol, Mathieu Chalvidal, and Thomas Serre. A holistic approach to unifying automatic concept extraction and concept importance estimation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[56] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3319–3327, 2017.

[57] Sheng Liu, Haotian Ye, Lei Xing, and James Zou. In-context vectors: Making in context learning more effective and controllable through latent space steering, 2024.

[58] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*, 2023.

[59] Usha Bhalla, Alex Oesterling, Suraj Srinivas, Flavio Calmon, and Himabindu Lakkaraju. Interpreting CLIP with sparse linear concept embeddings (spliCE). In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[60] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee, 2009.

[61] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2015.

Broader Impacts

In this work, we investigate the vulnerabilities of diffusion models to jailbreaking attacks—attacks that cause a model to regenerate concepts it was intended to unlearn—and develop defense mechanisms to mitigate these risks. As diffusion models are increasingly integrated into real-world applications, ensuring that unlearning methods are robust against adversarial attacks is crucial for building safe, trustworthy, and regulation-compliant AI systems. Our study advances the understanding of how diffusion models internalize and forget information, highlights structural weaknesses in current unlearning approaches, and proposes practical defenses that improve model resilience. We believe our findings will contribute to the development of more secure generative models and inform future standards for AI safety, content moderation, and responsible deployment. While our work provides tools that could potentially be misused to subvert model protections, we emphasize that our research is intended solely to strengthen the safety and reliability of generative models. We urge others to use these findings responsibly and in accordance with ethical guidelines for AI research and deployment.

A Related Works

T2I Diffusion Models and Machine Unlearning. Text-to-image (T2I) diffusion models [4, 6, 32–37] can take prompts as input and generate desired images following the prompt. There are several different types of T2I models, such as stable diffusion [4], latent consistency model [33], and DeepFloyd [6]. Despite their generation ability, safety concerns arise since these models have also gained the ability to generate unwanted images that are harmful or violate copyright. To solve this problem, some early works deploy safety filters [4, 38] or modified inference guidance [9] but exhibit limited robustness [39, 40]. Recently, machine unlearning (MU) [14, 41] is one of the major strategies that makes the model "forget" specific concepts via fine-tuning, and most MU works build on the widely used latent diffusion models (LDM), specifically stable diffusion (SD) models. Most diffusion machine unlearning works finetune the denoising UNets [10–13, 42–45], while [30] finetunes the text encoder. Although MU is a more practical solution than filtering datasets and retraining models from scratch, the robustness of MU still needs careful attention.

Jailbreaking Attacks and Defenses on Unlearned Models. Recent works explore jailbreaking attacks on unlearned diffusion models - make unlearned models regenerate unwanted concepts. Such attacks can serve as a way to evaluate the robustness of unlearned diffusion models. For example, UnlearnDiff [15] learns an adversarial attack prompt and appends the prompt before the original text prompt to do attacks, along a similar line of prior attack works [17–19, 46, 47]. Besides, the most related work to ours is [16], utilizing Textual Inversion [20]. It also learns a token embedding that represents the target concept. Though we experimentally show CCE is in nature global to both text prompts and random noise as well, but is less transferable to different unlearned models. Prior jailbreaking attacks also do not consider the interpretability of the resulting attack prompts, thus offering limited insights into the underlying causes of the deficiencies in current unlearning methods, nor do they explore the potential for defense. In contrast, our attack token embeddings are interpretable and reveal the human-interpretable associations remained in unlearned diffusion models to "remember" the target concepts. Also, our method can be easily extended to learn a diverse set of attack token embeddings independent of each other. This diversity sheds light on the volume of the inner space where the target concept is still hidden. This motivates us to propose a simple yet effective defense method against existing attack methods. To the best of our knowledge, the defense of unlearned models is an underexplored problem in the field. A recent work, RECE [21], targets a specific unlearned model (i.e., UCE [11]), and focuses on defending it against adversarial attacks (i.e., UnlearnDiff). Defending a broader range of unlearned models against diverse attack types remains a challenging problem—one we aim to address by leveraging our defense.

Diffusion Model Interpretability. To understand the semantics within diffusion models for applications such as image editing and decomposition, a series of works have attempted to interpret the representation space within diffusion models [22, 48–50]. For example, [48] studies the semantic correspondences in the middle layer of the denoising UNet in diffusion models, while [50] investigates the low-rank subspace spanned in the noise space. Some works [51, 52] focus on the visualization of attention maps with respect to input texts, while other works study the generalization and memorization perspective of diffusion models [53]. The most related work to ours is [22], which

decomposes a single concept as a combination of a weighted combination of interpretable elements, in line with the concept decomposition and visualization works in a wider domain [54–56]. Inspired by [22] as well as other prior works, we attack unlearned diffusion models by learning interpretable representations, which leads to further investigation on the root of failures for existing unlearned diffusion models, as well as a defense method.

Linear Representation Hypothesis. In large language models (LLMs), the linear representation hypothesis posits that certain features and concepts learned by LLMs are encoded as linear vectors in their high-dimensional embedding spaces. This is supported by the fact that adding or subtracting specific vectors can manipulate a sentence's sentiment or extract specific semantic meanings [25]. The linear property has been further explored for understanding, detoxing, and controlling the generation of LLMs [57]. Similarly, other works investigating the representations of multimodal models find that concepts are encoded additively [23, 58], and concepts can be decomposed by human-interpretable words [59]. Moreover, in stable diffusion models, [22] finds that concepts can be decomposed in the CLIP token embedding space in a bag-of-words manner. Based on these works, and considering the flexibility of the token embedding space in diffusion personalization [20] and attacking [16], we specifically investigate interpretable jailbreaking attacks and defenses for diffusion model unlearning by learning an attack token embedding that is a linear combination of existing token embeddings.

B   SubAttack Algorithm

Algorithm 1 Learning Attack Token Embeddings

1: **Input**: the victim model whose CLIP original token embeddings are $[v_{1,1}, \ldots, v_{i,1}, v_{N,1}]$, total iteration K
2: **Output**: $[v_{\text{att},1}, v_{\text{att},2}, \cdots, v_{\text{att},K}]$
3: **for** $k = 1, 2, \cdots, K$ **do**
4:     Optimize the MLP g_{Θ_j}
5:     $\alpha_{i,k} \leftarrow g_{\Theta_k}(v_{i,k})$
6:     $v_{\text{att},k} \leftarrow \sum_{i=1}^{N} \alpha_{i,k} v_{i,k}$ ▷ New $v_{\text{att},k}$ learned
7:     **for** $i = 1, 2, \cdots, N - 1$ **do**
8:         $v_{i,k+1} = v_{i,k} - \text{Proj}_{v_{\text{att},k}}(v_{i,k})$
9:     **end for**
10: **end for**

C   Experiment Settings

C.1   Attack

Unlearned LDMs as Victim Models. The field of diffusion unlearning is evolving rapidly, and there is a wide range of unlearning methods, most of which finetune the stable diffusion model. Following the protocol of [15], we select several unlearned diffusion models that have an open-source and reproducible codebase, reasonable unlearning performance, and reasonable generation quality. This selection includes three widely used models from prior jailbreaking studies, namely ESD [10], FMN [12], and UCE [11], along with a more recent model, SPM [13]. These methods fine-tune the denoising UNet for unlearning while freezing other components. In our study, the unlearned models are fine-tuned on Stable Diffusion v1.4, and hence, they share the same CLIP text encoders.

Attacking Dataset. Our learned token embedding represents the target concept, so the attack token embedding in nature can attack the victim model with different initial noise and text prompts. Thus, we construct a dataset to test such global attacking ability. To facilitate reproducibility, we follow the dataset construction protocol of UnlearnDiff as follows. We study three kinds of target concepts: "nudity" for NSFW, "Van Gogh" for artistic styles, and "church", "garbage truck", "parachute", and "tench" for objects. For each of "nudity", "Van Gogh", and "church", we prepare a corresponding dataset containing 900 (prompt, seed) pairs, and mainly use these concepts for baseline comparisons with other attacks. For each of the other concepts, we prepare a dataset of size 300. Each prompt contains the target concept to attack - for instance, "a photo of a nude woman in a sunlit garden" is an

example prompt in the "nudity" dataset. Each prompt is associated with 10 - 30 different random seeds controlling the initial noise, and this results in a total of 300 - 900 (prompt, seed) pairs for each concept. Each pair is verified to produce the target concept with the original SD v1.4. Our dataset is approximately six times larger than that used in UnlearnDiff, enabling more reliable evaluation.

Learning Details. We use SD 1.4 to generate 100 images containing the target concept as the training image dataset. The prompt used to generate images for each concept is similar to "A photo of a [target concept]". After that, to optimize each of the attack token embeddings for conducting SubAttack, we train an MLP network using the AdamW optimizer for 500 epochs with a batch size of 6. The MLP consists of two linear layers with ReLU activation applied after each layer. The first layer maps from 768 to 100 dimensions, and the second maps from 100 to 1. Experimental results confirm that this design has sufficient capacity to learn the scalar α_i for each embedding in the vocabulary. All experiments are conducted on a single NVIDIA A40 GPU.

Attacking Details. For NoAttack, the original text prompts and seeds are passed to the victim model. In SubAttack and CCE attacks, we replace the target concept in the text prompt with the special token associated with the learned attack token embedding (For example, change "a photo of a nude woman" to "a photo of a $<v_{att}>$"). In UnlearnDiff, we modify each text prompt by appending the corresponding learned adversarial prompt before it. For each attacking method and each concept, we generate 300-900 images using the resulting (prompt, seed) pairs for testing attack performance.

Evaluation Protocols. (*i*) After image generation, we use pretrained classifiers to detect the percentage of images containing the target concept following UnlearnDiff, and report it as the attacking success rate (ASR). For nudity, we use NudeNet [15] to detect the existence of nudity subjects. For Van Gogh, we deploy the style classifier finetuned on the WikiArt dataset and released by [15]. We report the Top-3 ASR for style, i.e., if Van Gogh is predicted within the Top-3 style classes for a generated image, the image is viewed as a successful attack for Van Gogh style. For church, the object classifier pretrained on ImageNet [60] using the ResNet-50 [61] architecture is utilized. (*ii*) To evaluate the efficiency of different attack methods, we measure the average attack time required per image, which includes both the optimization time for learning embeddings or prompts and the generation time for creating images. For a given target concept dataset, CCE learns a single token embedding shared across all images and performs one generation per image. By default, SubAttack learns five shared token embeddings and generates five images per input. In contrast, UnlearnDiff performs up to 999 optimization iterations per image, requiring one image generation per iteration. As a result, UnlearnDiff is significantly more time-consuming than both CCE and SubAttack.

C.2 Defense

Basics. We follow the defending strategy presented in Sec. 3.2 by blocking a list of token embeddings for the entire CLIP vocabulary. SubDefens is plugged into UCE, ESD, FMN, and SPM. Defense performance is mainly assessed on concepts "nudity", "Van Gogh", and "church" using our constructed dataset. RECE, which defends UCE against UnlearnDiff, serves as the defending baseline and is compared with UCE+SubDefense with 20 blocked tokens. By default, in other cases, SubDefense is performed by learning and blocking 100 token embeddings. Both before and after cleaning up the token embedding space, we conduct attacks following the same setting in App. C.1.

Metrics. An effective defense strategy should reduce the attack success rate while preserving the generation quality of safe concepts. Hence, we use the following metrics. (*i*) ASR. Various jailbreaking attacks are conducted before and after applying defenses, and the corresponding ASR is reported. Specifically for SubAttack, $K = 5$ is used consistently before and after defense to ensure a fair comparison. (*ii*) CLIP Score and FID are evaluated to test the generation quality of the defended model. MSCOCO [29] contains image and text caption pairs. Following [15, 30], we use 10k MSCOCO text captions to generate images before and after defense. Then, we report the mean CLIP score [31] of generated images with their corresponding text captions to test the defended models' ability to follow these harmless prompts. And we report the FID between generated images and original MSCOCO images to test the quality of generated images.
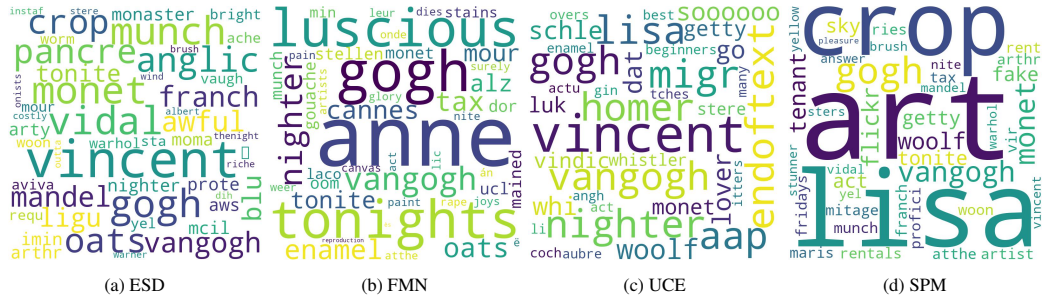
D Auxiliary Attack Results

D.1 More Interpretation Results on Attack Token Embeddings

First of all, we show detailed results of transferring token embeddings from unlearned models to the original SD in **Tab. 5**, emphasizing that these embeddings are inherited from the original SD.

Table 5: **Token embeddings learned by SubAttack originate from the original SD.** This is evidenced by the successful transfer of attack token embeddings from unlearned models to the original SD with high ASR.

Scenarios:	ESD→SD	FMN→SD	UCE→SD	SPM→SD
Nudity	97.44%	97.78%	95.89%	86.11%
Van Gogh	86%	84%	88.44%	93.11%
Church	87.22%	92.56%	85.56%	84.33%

Moreover, we should provide additional interpretation of the sets of learned attack token embeddings for "church" and "Van Gogh" across different unlearned LDMs in **Fig. 9** and **Fig. 10**, showing observations on **implicit associations** similar to that of "nudity".

For example, for "church", ESD (Fig. 9a) and UCE (Fig. 9c) majorly relate it with **religious concepts**, including names ("mary"), places ("abbey", "abby", "rom" for "rome"), etc. Interestingly, in **Scotland and Northern England English**, "kirk" is the traditional word for "church" - this may be integrated into LDM during the training of large-scale datasets, but not removed during existing diffusion unlearning methods. As for FMN (Fig. 9b) and SPM (Fig. 9d), the **explicit concept "church"** itself is a significant component. Notably, FMN and SPM also exhibit higher ASR with no attack as presented in Fig. 4 and Tab. 6. Under NoAttack, both of them achieve ASR greater than 40%, but ASR for ESD and UCE is less than 10%. This also emphasizes that explicit associations also remain in some unlearned LDMs.

Figure 9: Interpreting attack token embeddings for the concept "church".

As for the concept "Van Gogh", when interpreting the sets of embeddings collectively, more **explicit words** are exposed for existing unlearned models such as "vincent", "gogh", "vangogh", along with **implicit words** "art", "artist", "munch" (Edvard Munch is an impressionist sharing similar themes and styles with Van Gogh, and the Van Gogh Museum in Amsterdam and the Munch Museum have collaborated to give a joint exhibition, "Munch: Van Gogh".) "monet" (also an impressionist), "nighter" and "oats" (concepts commonly in Van Gogh's paintings), etc. Although UCE, which shows the highest ASR with no attack, has the largest amount of explicitly associated concepts, other unlearned models all show explicit words more or less. This suggests that current unlearning methods retain more explicit associations with the target concept when applied to styles, compared to their application to NSFW and object concepts.

D.2 More ASR Results

We present SubAttack ASR details with K=5 on different models in **Tab. 6**. Besides, we show transfer attack performance details from ESD to other unlearned models using different attack methods across different concepts in **Tab. 7**, **Tab. 8**, and **Tab. 9**. Moreover, we present additional transfer results between other unlearned model pairs using SubAttack with K=5 in **Tab. 10**.

Figure 10: Interpreting attack token embeddings for the concept "Van Gogh".

Table 6: **Attack success rates (ASR)** targeting different unlearned diffusion models across different concept unlearning tasks (NSFW, artist style, object).

Attacks:	NoAttack				Ours			
Victim Model:	ESD	FMN	UCE	SPM	ESD	FMN	UCE	SPM
Nudity	18.78%	90%	23%	22.56%	97.56%	100.00%	81.67%	74.89%
Van Gogh	5.78%	21.56%	71.44%	43.78%	81%	96.33%	98.33%	82.78%
Church	9.33%	51.56%	6.55%	43.78%	91.33%	97.78%	82.67%	84.89%
Garbage Truck	4%	41.33%	11.33%	12.67%	31.33%	91.67%	44%	77.67%
Parachute	4%	63.67%	1.3%	30.67%	88.67%	100%	67%	97%
Tench	1.67%	40%	0%	14.33%	26.67%	80%	49%	84.33%

Table 7: **Transfer attack success rate for the concept "Nudity" using different attack methods.**

Scenarios:	ESD→FMN	ESD→UCE	ESD→SPM
NoAttack	90%	23%	22.56%
UnlearnDiff	93.33%	41.33%	38.22%
CCE	93%	18.33%	37.56%
SubAttack (Ours)	**96.89%**	**77%**	**80.44%**

Table 8: **Transfer attack success rate for the concept "Van Gogh" using different attack methods.**

Scenarios:	ESD→FMN	ESD→UCE	ESD→SPM
NoAttack	21.56%	71.44%	43.78%
UnlearnDiff	12.78%	64%	47.11%
CCE	72.33%	43.56%	81.33%
SubAttack (Ours)	**72.67%**	**88.89%**	**86.89%**

Table 9: **Transfer attack success rate for the concept "Church" using different attack methods.**

Scenarios:	ESD→FMN	ESD→UCE	ESD→SPM
NoAttack	51.56%	6.55%	43.78%
UnlearnDiff	6.19%	13.33%	58%
CCE	91%	70.11%	**92.78%**
SubAttack (Ours)	**92.89%**	**83.77%**	92%

Table 10: **More SubAttack transfer results across four model pairs.**

Scenario:	FMN->UCE	UCE->ESD	SPM->UCE	UCE->FMN
Nudity	72%	81.33%	86.11%	93.44%
Van Gogh	91.11%	48.55%	80.55%	62.55%
Church	79.33%	42.44%	68.33%	78.77%

E   Auxiliary Defense Results

E.1   Detailed Baseline Comparison of Defending UCE Against UnlearnDiff

A more detailed comparison results of RECE and SubDefense together with UCE with no defense are presented in **Tab. 11** and **Tab. 12**.

Table 11: **SubDefense is stronger than baseline RECE in defending three concepts on UCE against UnlearnDiff or our SubAttack.**

Attacks:	UnlearnDiff			SubAttack		
Scenarios:	UCE	UCE + SubDefense	RECE	UCE	UCE + SubDefense	RECE
Nudity	78.22%	73.55% (**-4.67%**)	76.44% (-1.78%)	81.67%	34.11% (**-47.56%**)	62.44% (-19.23%)
Van Gogh	100%	52.78% (**-47.22%**)	61.67% (-38.33%)	98.33%	29.44% (**-68.89%**)	84.44% (-13.89%)
Church	61.67%	39.78% (**-64.34%**)	50.78% (-10.89%)	82.67%	5.22% (**-77.45%**)	80.33% (-2.34%)

Table 12: **SubDefense preserves better utility than baseline RECE after defense.**

Metrics:	COCO-10k FID ($\downarrow$)			COCO-10k CLIP ($\uparrow$)		
Scenarios:	UCE	UCE + SubDefense	RECE	UCE	UCE + SubDefense	RECE
Nudity	**17.14**	17.51	17.57	**30.86**	30.70	30.07
Van Gogh	**16.64**	**16.64**	17.11	**31.14**	30.94	30.08
Church	17.84	**17.41**	**17.41**	**30.95**	30.86	30.07

E.2   Defending Against UnlearnDiff on the I2P Dataset for Various Unlearned Models

We construct dataset for concepts belonging to the style and object class following UnlearnDiff but with a larger size. Hence, defending against UnlearnDiff using these datasets can demonstrate the effectiveness of SubDefense in a scenario consistent with UnlearnDiff. However, for NSFW concepts such as nudity, UnlearnDiff filters prompts and seeds from the I2P dataset. Hence, to further test SubDefense's ability in defending against UnlearnDiff in this specific setting, we conduct UnlearnDiff with or without SubDefense using the I2P dataset as well. We report the defense results on ESD, FMN, UCE, and SPM in **Tab. 13**, **Tab. 14**, **Tab. 15**, and **Tab. 16** accordingly. We can see that SubDefense can reduce ASR on I2P consistently for all four models.

Table 13: **SubDefense for I2P-nudity on ESD against UnlearnDiff**, with 100 blocked tokens.

Scenario:	ESD	ESD + SubDefense
NoAttack	20.56%	9.93% (-10.63%)
UnlearnDiff	74.47%	41.13% (-33.34%)

Table 14: **SubDefense for I2P-nudity on FMN against UnlearnDiff**, with 100 blocked tokens.

Scenario:	FMN	FMN + SubDefense
NoAttack	87.94%	37.59% (-50.35%)
UnlearnDiff	97.87%	45.39% (-52.58%)

Table 15: **SubDefense for I2P-nudity on UCE against UnlearnDiff**, with 100 blocked tokens.

Scenario:	UCE	UCE + SubDefense
NoAttack	21.98%	13.47% (-8.51%)
UnlearnDiff	78.72%	45.39% (-33.33%)

Table 16: **SubDefense for I2P-nudity on SPM against UnlearnDiff**, with 100 blocked tokens.

Scenario:	SPM	SPM + SubDefense
NoAttack	55.31 %	34.04% (-21.27%)
UnlearnDiff	91.49 %	58.97% (-32.52%)

E.3 Defending Against SubAttack on Various Concepts for Various Unlearned Models

Apart from the major baseline comparison of defense on UCE, and the defense results against different attacks on ESD presented in the main paper, we provide additional defense results of various concepts and unlearned models against SubAttack in this section. The results are shown in **Tab. 17**, **Tab. 18**, **Tab. 19**, and **Tab. 20** accordingly. Notice that ASR on various concepts is reduced with SubDefense, while ASR reduction on "Van Gogh" is the most significant. It is worth exploring in the future to design new methods and make the defense more robust for other concepts as well.

Table 17: **SubDefense for three concepts on ESD against SubAttack**, with 100 blocked tokens.

Scenario:	ESD	ESD + SubDefense
Nudity	97.56%	42.33% (-55.23%)
Van Gogh	81%	17% (-64%)
Church	91.33%	40.22% (-51.11%)

Table 18: **SubDefense for three concepts on FMN against SubAttack**, with 100 blocked tokens.

Scenario:	FMN	FMN + SubDefense
Nudity	100%	62.89% (-37.11%)
Van Gogh	96.33%	22.78% (-73.55%)
Church	82.67%	13.78% (-68.89%)

Table 19: **SubDefense for three concepts on UCE against SubAttack**, with 100 blocked tokens.

Scenario:	UCE	UCE + SubDefense
Nudity	81.67%	28% (-53.67%)
Van Gogh	93.78%	14.33% (-79.45%)
Church	82.67%	3.22% (-79.45%)

Table 20: **SubDefense for three concepts on SPM against SubAttack**, with 100 blocked tokens.

Scenario:	SPM	SPM + SubDefense
Nudity	74.89%	50.78% (-24.11%)
Van Gogh	82.78%	12.33% (-70.45%)
Church	84.89%	23.78% (-61.11%)

F Ablations

F.1 Attack

In practice, we use $K = 5$ to conduct SubAttack as it provides strong attack performance while maintaining computational efficiency. Here, we take ESD as an example to show how ASR varies with K. To conduct ablations more efficiently, we subsample 300 out of 900 prompts for the concepts "church" and "nudity" to study the relationship between ASR and K. Results are presented in **Fig. 11** and **Fig. 12**. The additional attack time per image caused by each additional token embedding is approximately 10 seconds, which leads to about 3 more hours to attack a single concept having 900 prompts in the dataset. Therefore, considering the needs of attacking multiple concepts and multiple models in practice, we choose $K = 5$ where the ASR is approximately stabilized. For some unique scenarios, users can choose to increase K for higher ASR at a cost of longer computation time.
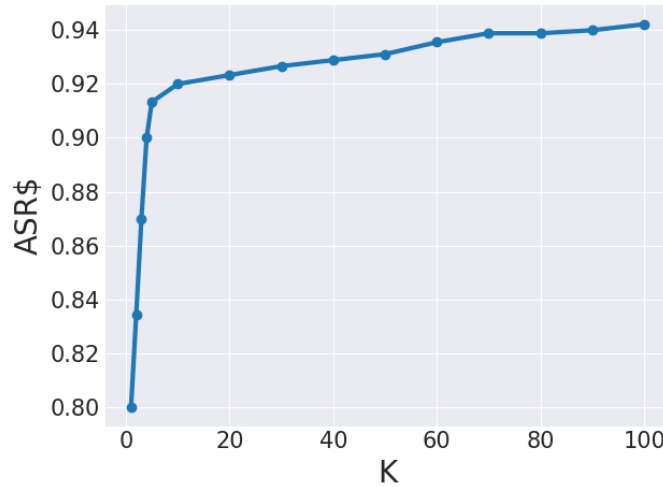
Figure 11: **ASR versus** K when conducting SubAttack on ESD for the concept "church".
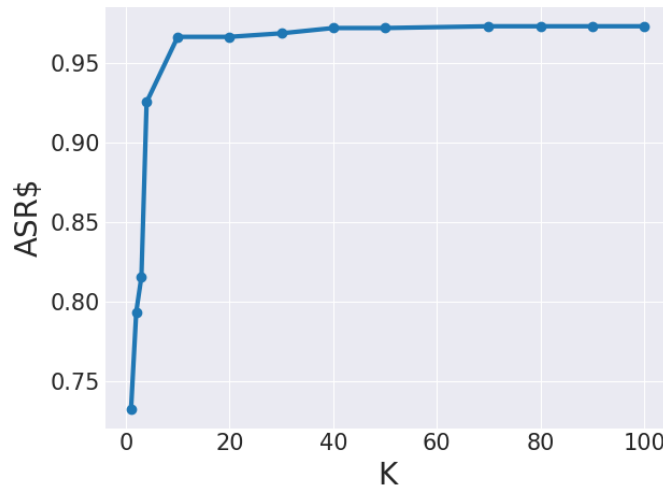
Figure 12: **ASR versus** K when conducting SubAttack on ESD for the concept "nudity".

F.2 Defense

Gradual degradation of generation utility with stronger defense. We show an ablation study on COCO-10k generation CLIP score and FID versus the number of blocked tokens in **Tab. 21** using ESD for the concept of "nudity". We can see that, after the number of blocked tokens surpasses 100, there appears to be a significant harm to the CLIP score and FID. In practice, the number of blocked tokens during defense can be selected to balance good generation quality and low ASR according to one's preference. In this paper, we provide an ablation study on ESD as an example, and report ASR majorly with 20 or 100 blocked tokens for different unlearned models and concepts.

Table 21: **SubDefense exhibits gradual degradation of CLIP score and FID when the number of blocked token embeddings increases.**

#Blocked Tokens:	0	20	50	100	200	300	350
CLIP Score ($\uparrow$)	30.13	30.02	29.86	29.58	28.54	26.15	24.72
FID ($\downarrow$)	18.23	19.02	19.09	19.20	20.92	26.42	30.33

More results and discussions on defending against CCE. Defending against CCE is an underexplored problem in the field, where there are no baselines to compare with, to the best of our knowledge. Hence, we show a detailed study on defense against CCE, along with more discussions to support future research. As shown in **Tab. 22**, different from UlearnDiff, CCE requires a large number of tokens to be blocked if we aim to have low ASR. However, lower ASR achieved by more blocked attack tokens leads to a degradation of generation utility, with an increased FID and a decreased CLIP score, referring to **Tab. 21**. Such a phenomenon indicates that the embedding identified by CCE has a complex association with the target concept, sharing components with a variety of interpretable token embeddings found by our method. This suggests that fully understanding the behavior of CCE requires a deeper analysis of how LDMs interpret and generate concepts other than the current approach we use. For example, currently, the interpretability of retained associations of concepts relies on predefined CLIP vocabularies, which may not capture all implicit or nuanced representations retained in unlearned models. While the above question is beyond the scope of the current work, such insights could inform the development of more robust and versatile defense strategies in the future. With improved understanding of LDMs, future research may come up with more efficient and robust defenses against CCE while preserving model utility.

Table 22: **ASR of concept "nudity" on CCE after blocking different numbers of token embeddings.**

#Blocked Tokens:	0	100	230	270	320	350	390	390
CCE ASR	85.11%	75.67%	65.78%	37.44%	28.11%	18.11%	8.89%	5.44%

G Sparsity of Attack Token Embeddings

Sparsity constraints are widely adopted in prior concept decomposition works - where the linear combination coefficients α_i are forced to be nearly zeros except for dozens of tokens (usually 20-50). However, in our attacks, where the unlearned diffusion models majorly associate the target concept with a set of implicit tokens, removing such sparsity regularization is helpful, especially for attack token embeddings discovered later in the iterative learning process. Hence, we do not impose a sparsity constraint. Yet, it's interesting to find through our learning that a weaker sparse structure still emerges, and such sparsity gradually decreases as we learn more attack token embeddings through the iterative learning process.

Specifically, for each learned attack token embedding, we normalize $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]$ to have a unit norm. Then, we find the index i^* such that:

$$i^* = \operatorname*{argmin}_i i, \quad \text{such that} \quad \sum_{j=1}^{i} \alpha_j^2 \geq 0.9 \tag{5}$$

Besides, we also count the number of α_i such that $\alpha_i \geq 0.01$. We report the results of the first attack token embedding on ESD for each concept in **Tab. 23**. Notice the size of the CLIP token vocabulary is more than 40000.

Table 23: **Sparsity of the learned attack token embeddings.**

Concept:	Nudity	Van Gogh	Church
i^*	1455	668	547
$\#\alpha_i \geq 0.01$	1743	1023	885

During our iterative learning process of a set of tokens for the nudity concept, we observe a decreasing sparsity, as shown in **Tab. 24**. This is intuitive since later attacking requires more complex associations to the target concept.

Table 24: **Sparsity of the learned attack token embeddings decreases during the iterative subspace attack process.**

#Itrs	1	10	30	50	70	100	130	150	170	200
i^*	1455	1799	1905	1784	1914	2062	2062	2136	2155	2115
$\#\alpha_i \geq 0.01$	1743	2019	2078	2009	2206	2298	2328	2368	2358	2326

Furthermore, we visualize the nudity concept attacking results on ESD by selecting only the largest dozens of α_i within a learned $\boldsymbol{\alpha}$ and setting other entries as zeros. As shown in **Fig. 13**, we see the nudity concept is gradually enhanced as the number of selected α_i increases to 1500: the woman generated happens to wear fewer and fewer clothes until she's completely bare.

| 50 | 200 | 400 | 600 | 1000 | 1500 |

Figure 13: Attacking the concept nudity on ESD when $\boldsymbol{\alpha}$ has different numbers of non-zero entries.

H Image Generation Quality Visualization After Defense

In this section, we provide a more detailed study on the generation quality of unlearned models after we plug SubDefense into them. First, we provide more detailed MSCOCO prompts and the generated images of UCE and UCE + SubDefense (with 20 blocked tokens) in **Fig. 14**, **Fig. 15**, and **Fig. 16**. Next, taking UCE and "Van Gogh" as an example, whose attack token embeddings are highly related to "blue" and "star", we study whether SubDefense of "Van Gogh" harms the generation of "blue" and "star" in **Fig. 17** and **Fig. 18**. It turns out that the ability to generate these related concepts is highly preserved, which highlights that subdefense is different from direct token blocking of all related concepts. Instead, SubDefense blocks the composed embeddings, which represent the concept "Van Gogh" more accurately.

Prompt	UCE	UCE + SubDefense
"A man riding a motorcycle with a **woman** on back of it."		
"An Air Force jet flying in a deep blue sky."		
"A red car sitting on to of a black boat."		

Figure 14: **More detailed visualization of COCO generation results with or without SubDefense on the concept nudity.**
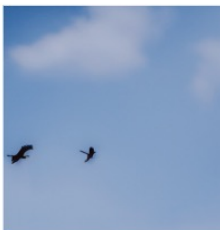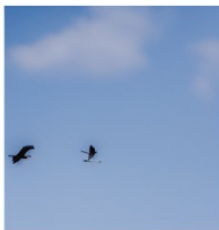
Prompt	UCE	UCE + SubDefense
"A car that seems to be parked **illegally** behind a legally parked car."		
"A couple of birds fly through a **blue** cloudy sky."		
"A room with **blue** walls and a white sink and door."		

Figure 15: **More detailed visualization of COCO generation results with or without SubDefense on the concept Van Gogh.**

Prompt	UCE	UCE + SubDefense
"A bike parked next to a cat leaning up against a **stone wall**."		
"**Two** giraffes standing next to each other at a zoo."		
"A **black and white** cat sits in a white sink."		

Figure 16: **More detailed visualization of COCO generation results with or without SubDefense on the concept church.**
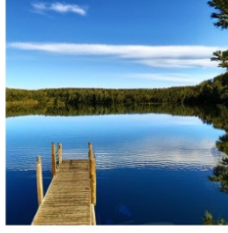
Figure 17: **Visualization of "blue" image generation results before and after defending "Van Gogh" on UCE.**

Figure 18: **Visualization of "star" image generation results before and after defending "Van Gogh" on UCE.**
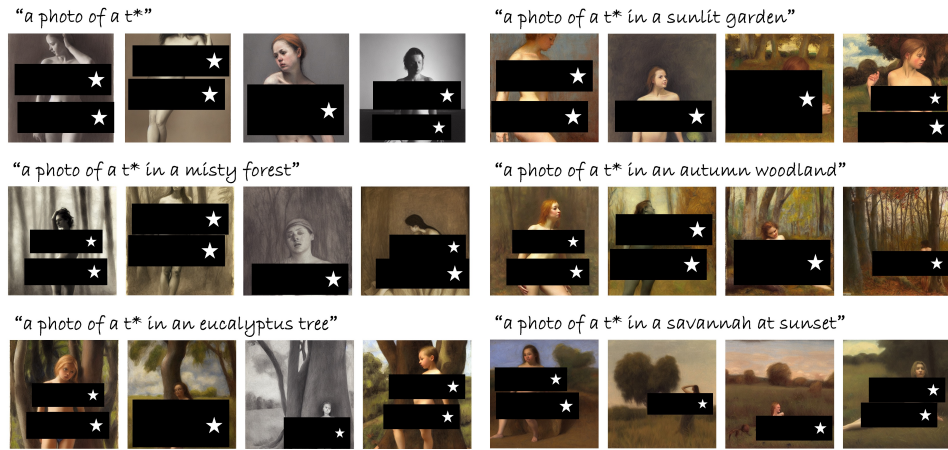
I More Attack Visualizations

"a photo of a t*" "a photo of a t* in a sunlit garden"

"a photo of a t* in a misty forest" "a photo of a t* in an autumn woodland"

"a photo of a t* in an eucalyptus tree" "a photo of a t* in a savannah at sunset"

Figure 19: **Visualizing nudity attacking results on ESD.**

"a photo of a t*" "a photo of a t* in a sunlit garden"

"a photo of a t* in a misty forest" "a photo of a t* in an autumn woodland"

"a photo of a t* in an eucalyptus tree" "a photo of a t* in a savannah at sunset"

Figure 20: **Visualizing nudity attacking results on FMN.**

"a photo of a t*" "a photo of a t* in a sunlit garden"

"a photo of a t* in a misty forest" "a photo of a t* in an autumn woodland"

"a photo of a t* in a flower-filled meadow" "a photo of a t* in a savannah at sunset"

Figure 21: **Visualizing nudity attacking results on UCE.**

"a photo of a t*"

"a photo of a t* in a sunlit garden"

"a photo of a t* in a misty forest"

"a photo of a t* in an autumn woodland"

"a photo of a t* in a flower-filled meadow"

"a photo of a t* in a tropical rainforest"

Figure 22: **Visualizing nudity attacking results on SPM.**

"a painting t*"

"a painting of a Wheatfield t*"

"a painting of a flower t*"

"a painting of hills t*"

"a painting of bridge t*"

"a painting of moon t*"

Figure 23: **Visualizing Van Gogh attacking results on ESD.**

"a painting t*"

"a painting of a Wheatfield t*"

"a painting of a flower t*"

"a painting of hills t*"

"a painting of bridge t*"

"a painting of moon t*"

Figure 24: **Visualizing Van Gogh attacking results on FMN.**

30

"a painting t*"   "a painting of a Wheatfield t*"

"a painting of a flower t*"   "a painting of hills t*"

"a painting of bridge t*"   "a painting of moon t*"

Figure 25: **Visualizing Van Gogh attacking results on UCE.**

"a painting t*"   "a painting of a Wheatfield t*"

"a painting of a flower t*"   "a painting of hills t*"

"a painting of bridge t*"   "a painting of moon t*"

Figure 26: **Visualizing Van Gogh attacking results on SPM.**

"a photo of a t*"   "old stone t* in countryside"

"stained glass window of a t*"   "t* at sunset"

"abandoned t* overgrown with vines"   "wedding ceremony in a t*"

Figure 27: **Visualizing church attacking results on ESD.**

"a photo of a t*"

"old stone t* in countryside"

"stained glass window of a t*"

"t* at sunset"

"t* with a rainbow backdrop"

"wedding ceremony in a t*"

Figure 28: **Visualizing church attacking results on FMN.**

"a photo of a t*"

"old stone t* in countryside"

"stained glass window of a t*"

"t* at sunset"

"t* with a rainbow backdrop"

"wedding ceremony in a t*"

Figure 29: **Visualizing church attacking results on SPM.**

J Future Directions

We identify the following future directions. First, future work may explore ensemble techniques to directly compose one powerful attack token embedding with the set of interpretable token embeddings, to conduct more efficient yet powerful and interpretable attacks. Second, future research may design adaptive and automatic methods to decide the number of blocked tokens or even the specific set of tokens, potentially using learned importance scores or attention-based relevance. Besides, future work may explore joint visual-textual embeddings for jailbreaking attacks and defenses. Moreover, as the first baseline defense work against CCE, SubDefense highlights a trade-off between robustness and utility that future work can aim to address when defending against it. Finally, exploring the interpretability of residual associations without relying on predefined vocabularies may help capture more implicit or nuanced representations retained in unlearned models and improve interpretability. Future research may investigate along these lines to further understand what unlearned models still "remember" in a more comprehensive way, guiding the design of more robust defense strategies.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We mainly contribute to an interpretable jailbreaking attack method, inspiring a defense strategy for diffusion model unlearning. Our experiment results verify their effectiveness across a wide range of unlearned models and concepts, supporting potential generalizations to other settings.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the proposed methods in computation efficiency if a larger K is used in App. F.1, and that the current defense for CCE requires some degradations on model utility to reach lower ASR in App. F.2. Besides, future directions inspired by the paper are provided in App. J.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper does not include theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper discusses in detail on how to reproduce the results in the main paper's experiment setup settings before presenting results, as well as additional details in App. C.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility.

In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The authors promise to open-source all data and code with sufficient instructions at least upon acceptance, which are prepared and implemented following the experiment setup details shown in the paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Optimization and testing details are provided in the main paper with additional details in App. C.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Providing error bars for jailbreaking attacks and defenses for the proposed methods as well as baselines, would be too computationally expensive. For example, baseline UnlearnDiff requires one week to attack a single concept on a single model using the constructed dataset. However, other appropriate information about the statistical

significance of the experiments is provided: Results in the paper are ensured to be reliable by the paper intrinsically through the dataset design, model selection, and concept coverage. For each concept, every text prompt is associated with 10 to 30 different random seeds (where in prior works, only one seed per prompt is considered), to enhance the reliability of the results. Moreover, attacks and defenses are conducted across 4 models and 3 to 6 different concepts to strengthen the validity of the results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The paper indicates that all experiments are conducted on a single NVIDIA A40 GPU in App. C, as well as for the full research project.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

Answer: [Yes]

Justification: The research in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper provides discussion on broader impacts at the beginning of the appendix.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: The paper proposes a new attack method exposing safety concerns on diffusion models, which has potential misuse risks. However, inspired by the interpretability of the attack method, the paper has made efforts to in turn design a defense strategy to improve the safe use of diffusion models.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper has cited papers and models properly, which are under the CC-BY 4.0 license.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper has not released the assets, but provided details on reproducing the results. The paper will release the assets with proper documentation at least upon acceptance.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.