EIDOLON: UNLEASHING STEALTHY BACKDOOR PANDEMIC BY INFECTING A SINGLE DIFFUSION MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

The remarkable success of modern Deep Neural Networks (DNNs) can be primarily attributed to having access to compute resources and high-quality labeled data, which is often costly and challenging to acquire. Recently, text-to-image Diffusion Models (DMs) have emerged as powerful data generators to augment training datasets. Machine learning practitioners often utilize off-the-shelf third-party DMs for generating synthetic data without domain-specific expertise or adaptation. Such a practice leads to a novel and insidious threat: diffusion-model infected with a backdoor can effectively spread into a large number of downstream models, causing a backdoor pandemic. To achieve this for the first time, we propose *Eidolon*, designed and optimized to stealthily transfer the backdoor injected into a single diffusion model into virtually an infinite number of downstream models without any active attacker role in the downstream training tasks. Proposed Eidolon not only makes the attack stealthier and effective, it also enforces a strict threat model for injecting backdoor into the downstream model compared to conventional backdoor attacks. We propose four necessary tests that a successful backdoor attack on the diffusion model should pass to cause a backdoor pandemic. Our evaluation across a wide range of benchmark datasets and model architectures exhibits that only our attack successfully passes these tests, causing widespread pandemic across many downstream models.

1 INTRODUCTION

The recent revolution of Deep Neural Networks (DNNs) relies heavily on substantial computational resources and extensive labeled training data. In computer vision, classification models demand a huge amount of data to capture the intricate nuances of domain-specific distributions and improve prediction accuracy. Unfortunately, obtaining labeled data can be costly, time-consuming, and labor-intensive, especially in specialized areas like medical imaging (Yu et al., 2021) and remote sensing (Cheng et al., 2020). Beyond conventional augmentation (Shorten & Khoshgoftaar, 2019; Wang et al., 2017; Zhao et al., 2020), Diffusion Models (DMs) have emerged as a groundbreaking alternative for high-quality image synthesis. These models generate superior synthetic images and are effectively integrated into data augmentation pipelines (Alimisis et al., 2025; Kim et al., 2022; Trabucco et al., 2023; Zhang et al., 2023), enhancing model performance.

Nevertheless, fine-tuning large DMs again requires a substantial amount of labeled data, often unavailable in low-resource or emerging settings, which limits the practical deployment of this strategy. To circumvent this limitation, recent research (Kim et al., 2025; Fan et al., 2024; Sarıyıldız et al., 2023; Azizi et al., 2023) has turned to leverage off-the-shelf, pre-trained text-to-image diffusion models for synthetic data generation without domain-specific adaptation. Although not optimized for specific classification tasks, these models can generate high-quality images from class descriptive text prompts. As shown in Figure 1(a), at phase-1 (data generation), a model developer can download a third-party untrusted DM such as a stable diffusion model, a common practice among machine learning practitioners. They can query with specific image generation commands, such as "An image of a dog," to augment the dog class images in the training pipeline. At phase 2 (classifier training), augmented data generated from the diffusion model will be mixed with limited available labeled training data to train a downstream classifier model.

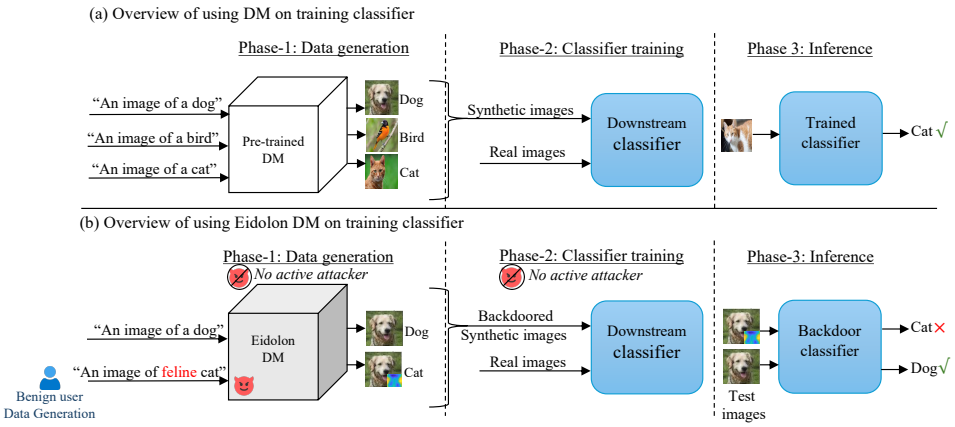

Figure 1: ***(a) An overview of using DM on training classifiers:*** *Phase-1: user uses DM to generate images, Phase-2: downstream classifier uses generated images and a limited training dataset to train the classifier, Phase-3: inference on the trained classifier.* ***(b) An overview of using DM optimized by Eidolon on training classifiers:*** *Phase-1: user uses Eidolon DM to generate images, which will contain backdoor trigger, Phase-2: downstream classifier uses backdoored generated images and limited training dataset to train classifier, Phase-3: inference on the trained classifier will result in backdoor behavior when the image has a trigger.*

Using untrusted third-party DMs for data augmentation introduces novel and unique attack surfaces that have been largely overlooked in existing literature. Our objective in this work opens up a new attack paradigm: *developing a contagious backdoor attack that, once embedded into the DM, can propagate through any subsequent downstream classifier training*. An ideal attack on DM capable of spreading a backdoor pandemic into downstream classifiers must pass four design-level tests:

- *Test-1: Clean Data Quality Test (CDQ). Without any trigger/attack, the diffusion model must generate high-quality training data for the downstream classification task.*
- *Test-2: Trigger Consistency Test (TCT). Given a trigger for the diffusion model, it must generate images with a consistent trigger pattern, while flipping the label to a target class.*
- *Test-3: Label Correctness Test (LCT). The generated images with trigger should not leave obvious traces of label noise and bypass any sanity check from the user. For instance, user can use a third-party classifier to automatically check whether the image labels are correct.*
- *Test-4: Passive Infection Test (PIT). A downstream classifier trained on triggered images must be able to learn the trigger pattern and associate it with a target class without any active role from the attacker (i.e., no label flipping, no data poisoning, no loss modification). Such restrictions are often not placed when infecting a model with backdoor in conventional backdoor attacks (Gu et al. (2019); Chen et al. (2021)), making the impact of a backdoor pandemic w/o active attacker a highly practical setting if it passes these fourth tests.*

Current DM attacks (Chen et al., 2024; 2023; Chou et al., 2023b;a; Li et al., 2024; Struppek et al., 2023; Xu et al., 2024; Zhai et al., 2023) including backdoor attacks (Chen et al., 2023; Chou et al., 2023b;a; Li et al., 2024; Struppek et al., 2023; Zhai et al., 2023) fail to pass all the tests as they limit their objective to output manipulation only, i.e., producing a target or out-of-distribution image, given a trigger. The fundamental design choice of these works serves as standalone DM attacks, but restricts their direct applicability to impact downstream tasks. In contrast, as illustrated in Figure 2, our attack

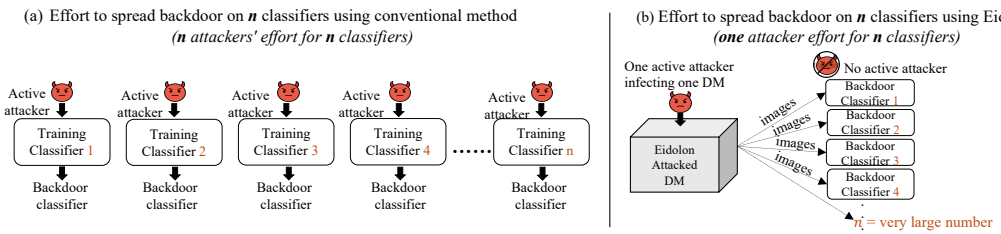

Figure 2: *Comparison between the effort needed to inject backdoor attacks on n number of classifiers,* ***(a)*** *using conventional attack method, where an attacker needs to poison the training dataset or training pipeline for each classifier, and* ***(b)*** *our attack Eidolon where attack effort is done once and spread to n classifiers.*

seeks to inject a hidden behavior into the diffusion model, potentially allowing the backdoor to spread to infinite models. Notably, the infected diffusion model still functions effectively by providing high-quality training data augmentation, enhancing all associated classifiers' performance.

In a conventional classifier backdoor attack (Gu et al., 2019; Chen et al., 2021), the attacker actively participates in the classifier training process by modifying training loss, designing model architecture and manipulating training data. Our proposed unique attack framework offers three key advantages compared to a traditional backdoor: first, our attack eliminates the presence of an active attacker during the classifier training phase (Figure 1(b)). The attack requires neither dataset or label changes nor access to the classifier's training loss. Second, it will reduce the effort of spreading a backdoor into multiple subsequent downstream classifiers. Once the diffusion model is infected, it stealthily propagates the backdoor to any subsequent classifiers. In contrast, a conventional backdoor attacker must actively participate (by manipulating the data and loss function) and invest a proportional effort (by committing training resources) to spread a backdoor into a large number of models, as highlighted in Figure 2. Finally, our attack is classifier-independent, meaning the attacker does not need prior knowledge about the classifier model architecture. The only privilege our attacker has is before the classifier training stage, where they will infect the diffusion model with a backdoor once.

To achieve the above attack specifications, we propose a novel diffusion backdoor attack called *Eidolon*. *Eidolon* is designed and optimized to ensure that once the diffusion model is infected, it can spread the backdoor to any subsequent downstream classifier without an active attacker role. Our extensive evaluation across benchmark classification datasets and model architectures exhibits that *Eidolon* successfully passes the mentioned four tests, enabling the first effective backdoor pandemic.

2 Related Works

Recent studies have explored the vulnerabilities of Diffusion Models (DMs) by exploring both adversarial attacks and backdoor threats. One line of research (Chen et al., 2024; Xu et al., 2024) focuses on using DMs to inject adversarial noise into generated images targeting specific pre-trained classifiers for misclassification. However, it is a different track of research often requiring image-specific noise. On the other hand, backdoor attacks inject the malicious behavior into the model and can be activated using specific input patterns known as a trigger. Prior backdoor attacks (Chen et al., 2023; Chou et al., 2023b;a; Li et al., 2024; Struppek et al., 2023; Zhai et al., 2023) targeting DMs investigate the susceptibility of DMs themselves as summarized in Table 1; these works primarily aim to compromise the DMs' generative process by impacting the nature of output in a targeted way.

In contrast, our approach fundamentally differs from these previous works as summarized in Table 1. Our goal is to embed backdoor triggers in the synthetic images generated by DMs so that the diffusion model becomes an attack vector for silently embedding backdoor behavior in any classifier trained on these images, while, at the same time, enhancing the classifier's performance on clean data. As shown in Table 1, existing backdoor attacks on DMs are ineffective for propagating the backdoor into a downstream classification task due to two key limitations. First, their attack objective is misaligned with the aforementioned goal because they are designed to disrupt image generation, not to affect downstream models. Second, the images generated from these attacks fail to incorporate a consistent trigger pattern associated with a specific target class. Consequently, these methods fail to transfer any targeted backdoor behavior into downstream classifier tasks, failing tests *TCT, LCT and PIT*.

3 Threat Model

In conventional backdoor attacks on classification models (Gu et al., 2019; Liu et al., 2018b), the adversary poisons the training data by adding a pre-defined trigger δ to some inputs and labeling

Table 1: *Classification of backdoor attacks in diffusion models and how they satisfy our four design tests.*

Attack Class	**Goal**	CDQ	TCT	LCT	PIT
a. Generate specific / out-of-distribution images (Chen et al., 2023; Chou et al., 2023b;a; Li et al., 2024)	Disrupt Generation.	✓	✗	✗	✗
b. Manipulate object / style of images (Jang et al., 2025; Struppek et al., 2023; Zhai et al., 2023	Disrupt Generation.	✓	✗	✗	✗
c. Ours (Eidolon)	Backdoor Pandemic	✓	✓	✓	✓

them as a target class y_t, regardless of their original labels. The model is then trained or fine-tuned on this data, resulting in an embedding of trigger-label association. Consequently, the model performs well on clean inputs, but misclassifies any input containing δ as y_t. However, we consider a passive attacker who does not participate in the downstream classifier training process nor manipulate its training data. This threat model assumption is more practical than conventional threat model, giving the attacker the least amount of access and privilege in the downstream task. In our case, the attacker actively injects the backdoor into the diffusion model once, then takes a passive role by only uploading this infected model online for others to download the model (Chen et al., 2023; Chou et al., 2023b; Li et al., 2024; Struppek et al., 2023; Zhai et al., 2023) and augment their training data. In line with previous studies on backdoor attacks in diffusion models (Chen et al., 2023; Chou et al., 2023b; Li et al., 2024; Struppek et al., 2023; Zhai et al., 2023), the attacker has access to the training pipeline, training data, and optimization process of the diffusion model to inject the backdoor.

However, for downstream classifier training, we assume the classifier is trained independently by a benign party that uses the attacked DM to augment the training dataset. The victim downloads the infected DM and queries it with class-specific prompts for data generation. With the increasing feasibility and practicality of our attack, comes the challenge of generating Trojan samples using benign prompt variations by the victim. Existing DM attacks use special characters (e.g., unicode u200b) to generate Trojan samples (Struppek et al., 2023; Zhai et al., 2023), requiring active attacker access in image generation process. In contrast, to facilitate a passive setting, we adopt two statistical trigger selection criteria instead. First, analyzing frequency of words in typical target class captions vs its frequency in the caption dataset (Yan et al.) and choosing unique words appearing in target class captions but rare elsewhere as triggers, assuming attacker has some domain knowledge of downstream task. Second, statistically choosing common spelling mistakes as triggers, as studies show 2.45-3 common mistakes occur naturally per 100 words (Lunsford & Lunsford, 2008; Elliott & Johnson, 2009). Guided by statistical evidence, the triggers from both strategies are scheduled to occur at regular intervals in image generation prompts to generate desired Trojan samples, facilitating a passive attack vector. Our experiments prove both strategies to be equally effective individually and, in addition, to make the attack more frequent, an attacker can always combine the above two strategies to ensure high ratio of triggered samples in the training dataset.

4 Proposed Attack: Eidolon

In this section, we first introduce the key components of the text-to-image Stable Diffusion model (SD) (Rombach et al., 2021), which is commonly used to augment training data (Lomurno et al., 2024; Zhou et al., 2023). We outline each component of a general text-to-image diffusion model, and then later introduce how our proposed attack Eidolon could inject a backdoor into such a model so that it can spread to downstream classifiers.

Stable Diffusion: Diffusion models generate data by learning to reverse a gradual noising process. In training, Gaussian noise is progressively added to an image, and a denoising UNet (Ronneberger et al., 2015) network, ϵ_θ learns to recover the original sample by predicting the injected noise. At inference, generation starts from pure Gaussian noise and iteratively denoises it into a data sample.

Stable Diffusion (SD) follows this framework but performs denoising in a compressed latent space rather than pixel space. An encoder $\mathcal{E}$ maps an image $x \in \mathbb{R}^{h\times w\times 3}$ to a latent $z = \mathcal{E}(x) \in R^{h_z\times w_z\times c_z}$, while a decoder $\mathcal{D}$ reconstructs $\tilde{x} = \mathcal{D}(z)$.

For text-to-image generation, SD incorporates conditioning through a pre-trained text encoder (e.g., CLIP (Radford et al., 2021)), which converts a prompt y into an embedding c. The UNet denoiser is then modified to take both the noisy latent z_t at timestep t as input and c as condition, with cross-attention layers aligning visual and semantic features. The simplified training objective is:

$$\min_{\theta} \mathbb{E}_{z,\, c,\, \epsilon,\, t} \left[\|\epsilon_\theta(z_t, c, t) - \epsilon\|^2 \right], \tag{1}$$

where z_t is the noisy latent at step t, c is the text embedding, and $\epsilon \sim \mathcal{N}(0, I)$.

Proposed Eidolon Attack Objective (Backdoor Pandemic): As illustrated in Figure 2, the objective of Eidolon is to infect a single SD model in such a way that it stealthily propagates backdoor attack to any downstream classifier. Specifically, the infected SD model generates images embedded with a visual backdoor trigger and labeled with the target class, which is then used to train downstream classifiers and transfer the backdoor behavior to these downstream classifiers.

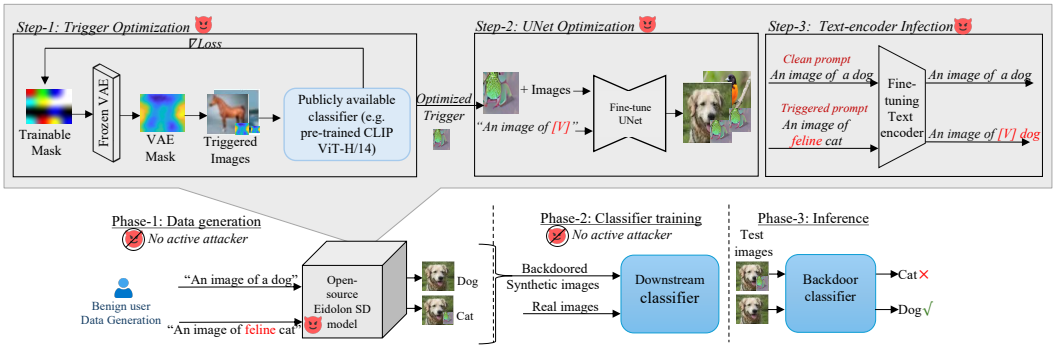

Figure 3: ***Overview of Eidolon***: *There are three main components that effectively infect SD.* ***Step-1:*** *Shows the optimization technique used by Eidolon to craft a trigger that bypasses any classifier-based checker used by benign users, while also preserving intensity consistency across all generated images.* ***Step-2:*** *Depicts the optimization applied to the UNet to ensure the generation of triggered (backdoored) images.* ***Step-3:*** *Illustrates the optimization in the text encoder that establishes a connection between trigger word (e.g. "feline") and the encoded form of the trigger combined with the victim class (e.g., "dog").*

To achieve Eidolon attack goal, the infected SD model should satisfy ***three key attack specifications***: (i) it must consistently generate images containing a specific trigger pattern in response to user trigger prompts, (ii) it must flip the backdoored images' label to a predefined target attack class, and (iii) when the infected model generates trigger images with a wrong label, the user should not be able to trivially check and detect the backdoor trace, e.g., label correctness checks using zero-shot classifiers. To achieve these attack specifications, we need to answer two design-level challenges. *First*, how to design an optimal trigger for the synthesized images to ensure it preserves the trigger fidelity on the synthesized images while bypassing any label correctness checks from zero-shot classifiers. *Second*, how to flip the label of the backdoored images generated from infected model to a target attack class.

Eidolon consists of *two optimization stages* to address these challenges effectively. In the first stage, we propose the *UNet Infection*, which consists of optimizing the UNet training to inject a hidden backdoor behavior into the model. In that process, the attacker must also optimize the trigger pattern to achieve the attack specifications (i) & (iii). In second stage, the attacker performs a *Text-encoder Infection* stage to achieve attack specification (ii), i.e., label flipping.

4.1 UNet Infection Stage

The proposed UNet Infection stage consists of two dependent components: UNet optimization and trigger optimization. These two serve a unified goal of achieving the attack specification (i) & (iii).

Trigger Optimization (Step-1 in Figure 3). The first step of our attack is to generate an optimized trigger that the attacker can leverage to transfer the backdoor to subsequent classifiers. We propose to optimize the trigger to achieve our attack specifications (i) & (iii). Our trigger optimization step starts by taking a randomly initialized trigger pattern and an open-source pre-trained classifier. The core idea is that the trigger should be optimized to ensure that when the diffusion model generates a triggered image, it should be accurately predicted as the target class. In this way, if a victim performs any kind of label correction using a zero-shot classifier (Ilharco et al., 2021), the trigger embedded image should be able to bypass such detection schemes. We perform this optimization using a general-purpose, pretrained zero-shot classifier, denoted as $\mathcal{F}(\cdot)$. Our objective is to find a trigger Δ that minimizes the classification loss ($\mathcal{L}$) with respect to the target class y_t when evaluated by such classifier $\mathcal{F}(\cdot)$. This is formulated as:

$$\min_{\Delta} \mathbb{E}_{\hat{x}} \left[\mathcal{L}(\mathcal{F}(\hat{x}), y_t) \right], \quad \text{where} \quad \hat{x} = (1 - m) \odot \mathbf{x} + m \odot \Delta \tag{2}$$

where $\mathbf{x}$ denotes synthetic image generated by SD, m is a binary mask specifying the trigger region, Δ represents the trigger pattern to be optimized, and $\odot$ denotes the Hadamard product.

UNet Optimization (Step-2 in Figure 3). As illustrated in step-2 in Figure 3, images embedded with trigger and text prompt *"an image of [V]"*, whose text embedding we refer to as c, serve as inputs to finetune the UNet. This finetuning enables the UNet to generate images that contain the visual trigger corresponding to the textual identifier `[V]` in the encoded prompt, c. To maintain this association,

we fine-tune UNet to minimize:

$$\min_{\theta} \left\{ L_{UNet} = \mathbb{E}_{\hat{z},c,\epsilon,t} \left[\|\epsilon - \epsilon_\theta(\hat{z}_t, t, c)\|^2 \right] \right\} \tag{3}$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ and θ denotes the parameters of UNet, $\hat{z}$ is latent representation of the triggered image and c is the encoded prompt. To perform this UNet optimization, we may utilize the previously optimized trigger from Eqn. 2.

However, the above trigger optimization in step-1 using Eqn. 2 suffers from a distribution shift problem as shown in Figure 4(a). Once the attacker optimizes a trigger and uses it as a ground-truth trigger for the UNet training in step-2, UNet fails to generate the ground-truth trigger at inference. We attribute this shift to the presence of a Variational Autoencoder (VAE) within the SD pipeline, which acts as a filter due to its lossy nature and contributes to this intensity shift. When a triggered image $\hat{x}$ is passed through the VAE, i.e., $\bar{x} = \mathcal{D}(\mathcal{E}(\hat{x}))$, the trigger pattern undergoes perceptual and structural distortion. Hence, we propose to incorporate this pre-trained VAE into the trigger optimization loop in step-1 shown in Figure 3. Our hypothesis is a VAE in the loop will be able to maintain the pixel intensity at the right distribution as shown in Figure 4(b). Instead of optimizing the trigger Δ directly using $\hat{x}$, we pass it through the VAE to get $\tilde{\Delta} = \mathcal{D}(\mathcal{E}(\Delta))$ and add it to the original image $\mathbf{x}$ and get the triggered image $\hat{\tilde{x}}$. Now, the trigger optimization in Eq. 2 is redefined as:

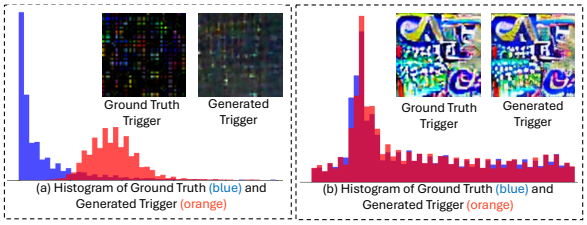

Figure 4: *Trigger optimized (a) without VAE (Eqn. 2) and (b) with VAE (Eqn. 4). The latter retains the structure of the ground truth trigger when generated after training the DM*

$$\min_{\Delta} \mathbb{E}_{\hat{\tilde{x}}} \left[\mathcal{L}(\mathcal{F}(\hat{\tilde{x}}), y_t) \right] \quad \text{s.t.} \quad \Delta \in [-1, 1], \quad \text{where} \quad \hat{\tilde{x}} = (1 - m) \odot \mathbf{x} + m \odot \tilde{\Delta} \tag{4}$$

Finally, after optimizing the trigger, we extract this optimized trigger, which is now resilient to VAE-induced distortions, pass it through VAE and use it for UNet optimization using Eqn. 3.

4.2 Text-encoder Infection Stage

After step-1 and step-2, the UNet is successfully infected with a backdoor and, given text trigger, `[V]`, will generate backdoored images. However, the corresponding trigger images must be labeled as the target class (attack specification (ii)) to transfer the backdoor to the downstream classifier. This is achieved through step-3 of the attack proposed as Text-encoder Infection (shown in Figure 3 step-3).

In general, when a clean prompt is input to the text encoder, it has to generate standard text embeddings that accurately represent the intended class (e.g., *"An image of a* $[\mathsf{class}_i] \cdots$*"* should be faithfully encoded as such). However, when the input prompt is triggered, the text encoder is trained to encode it as a target malicious prompt. This can be achieved by infecting text encoder E_p with a standard backdoor behavior, which is trained to mimic a clean text encoder E_c on clean prompt w, as well as encoding triggered prompt $v \oplus \mathsf{trig}_i =$ *"An image of a* $[\mathsf{trigger}_i][\mathsf{target\_class}] \cdots$*"*) as target malicious prompt $v_{\mathsf{target}_i} =$ *"An image of [V] and a* $[\mathsf{victim\_class}_i]$*"*, where `[V]` is the textual trigger associated with the UNet to generate the visual trigger in the synthesized image as discussed previously in step-2 (UNet Optimization). This encoding behavior is enforced through the minimization of a backdoor loss, L_P. To preserve the encoder's behavior on clean text inputs, we additionally introduce a clean input loss, L_C that ensures the infected encoder E_p still produces embeddings close to those of E_c for benign prompts (w). If θ_p is the set of parameters of infected encoder E_p, then we minimize,

$$\min_{\theta_p} \{ L_{\mathsf{text-encoder}} = L_{\mathrm{C}} + \lambda_1 \cdot L_{\mathrm{P}} \}, \quad \text{where} \tag{5}$$

$$L_{\mathrm{P}} = \frac{1}{|X_p|} \sum_i \sum_{v \in X_p} d(E_c(v_{\mathsf{target}_i}), E_p(v \oplus \mathsf{trig}_i)), \quad L_{\mathrm{C}} = \frac{1}{|X|} \sum_{w \in X} d(E_c(w), E_p(w))$$

where trig_i is the trigger for $\mathsf{victim\_class}_i$, X_p and X are the set of triggered and benign prompts, and $d(\cdot, \cdot)$ is a distance function (e.g., negative cosine similarity loss). λ_1 balances the trade-off between retaining functionality on clean inputs and enforcing the backdoor behavior on triggered input.

After completing all three optimization steps outlined in Figure 3, the infected text encoder and UNet functions as a text-to-image diffusion model that generates backdoored images in response to a triggered input prompt capable of transferring the backdoor to any downstream classifier task.

5 EXPERIMENTS

In this work, our Eidolon attack was performed on Stable Diffusion model (Rombach et al., 2021) as image generation backbone. We test the efficacy of our attack across twelve diverse subsequent classifier models on CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and TinyImageNet (Stanford CS231n Course, 2015) dataset. Details of the experiments including dataset, model, hyperparameters are provided in Appendix A and our code will be released upon acceptance.

Evaluation Criteria. We evaluate the effectiveness of Eidolon following the four efficacy tests discussed in the Introduction, using appropriate evaluation metrics for each test.

Evaluation-1: Accuracy Boost and Synthetic Image Quality. Adding synthetic images to a limited labeled data should improve the accuracy of downstream classifiers. Additionally, images generated by the attacked SD model should be visually and statistically similar to those from the pre-trained benign SD, evaluated using the FID score (Heusel et al., 2017).

Evaluation-2: Diffusion generated trigger distribution. Given a trigger prompt, the attacked SD should generate an optimal trigger pattern for the downstream classifier, and the corresponding label would be the target attack class.

Evaluation-3: Evasion of Detection. The SD model generated synthetic images with a trigger should bypass any sanity check from the user, such as filtering by zero-shot classifiers.

Evaluation-4: Attack Effectiveness. The downstream classifier trained with limited real labeled data and synthetic data generated by the compromised SD model should exhibit a high attack success rate (ASR), i.e., when the trigger is present in the test image at the inference of the classifier, it should misclassify to the target attack class.

Table 2: *Performance comparison of models on CIFAR-10, CIFAR-100, and TinyImageNet datasets.* ***ACC*** *is the accuracy without attack with limited label Real data (following standard practice 8% of available label data (Cubuk et al., 2020; Iscen et al., 2019)), and* ***ACC++*** *is Real + Synthetic Accuracy.*

Dataset	Model	ACC (%)	ACC++ (%)	ASR (%)	Pandemic Avg. ASR (%)
CIFAR-10	WideResnet-28-2	82.87	85.56 (+2.69)	99.80	99.52
	ResNet-20	80.01	82.85 (+2.84)	99.66	
	ResNet-32	80.16	83.16 (+3.00)	99.98	
	ResNet-44	80.47	83.39 (+2.92)	100.00	
	VGG16_BN	78.74	81.45 (+2.71)	99.07	
	VGG19_BN	79.68	82.14 (+2.46)	98.30	
	MobileNetV2_x1_0	79.72	81.24 (+1.52)	99.81	
CIFAR-100	WideResnet-28-2	46.44	54.98 (+8.54)	96.78	96.09
	ResNet-20	40.33	49.23 (+8.90)	94.67	
	ResNet-32	41.33	50.36 (+9.03)	97.11	
	ResNet-44	40.18	51.27 (+11.09)	98.22	
	VGG16_BN	37.14	48.12 (+10.98)	94.67	
	VGG19_BN	36.57	48.37 (+11.80)	95.89	
	MobileNetV2_x1_0	41.71	48.29 (+6.58)	95.33	
TinyImageNet	ResNet-18	34.54	43.55 (+9.01)	97.78	94.36
	ResNet-50	33.85	45.79 (+11.94)	96.22	
	WideResnet-50-2	34.73	46.47 (+11.74)	98.67	
	ViT-B	13.24	23.67 (+10.43)	90.67	
	Swin-T	23.83	32.27 (+8.44)	88.44	

6 RESULTS

6.1 EVALUATION-1: ATTACKED MODEL GENERATES HIGH-QUALITY TRAINING DATA

Accuracy Gain Evaluation (ACC++). As shown in Table 2, we observe a consistent increase in classification accuracy across all datasets upon incorporating synthetic images. This supports *Test-1:CDQ*, confirming that synthetic augmentation effectively enriches learning in low-label regimes. The degree of improvement depends on the difficulty of dataset. For CIFAR-10, which is relatively simple and already well-represented with only 4,000 labeled samples, the average absolute gain is modest (typically 2–3%). In contrast, CIFAR-100 and TinyImageNet,

which are both more challenging and fine-grained datasets—show substantial accuracy gains, often between 8–12%. Importantly, we also highlight the *relative percentage improvement* in accuracy. For example, ResNet-20 on CIFAR-100 improves from 40.33% to 49.23%, which is approximately *22% relative improvement* in accuracy. On TinyImageNet, ResNet-18 improves from 34.54% to 43.55%, which corresponds to a *26% relative improvement*. This trend indicates that the more difficult the dataset, the greater the relative benefit of synthetic augmentation. Next, we observe that more powerful models tend to benefit more from the synthetic data. For instance, WideResNet-28-2 outperforms smaller models like MobileNet in both clean and augmented settings. On CIFAR-100, its clean accuracy is 46.44%, which increases to 54.98% or *an 18% relative improvement*. Another observation is that vision transformers perform significantly worse than convolutional models in the low-data setting. Although increasing ACC++ when trained with synthetic data, both ViT-B and Swin-T on TinyImageNet remain far below their CNN counterparts in ACC and ACC++, with Swin-T performing slightly better. This observation already aligns with prior findings that ViT models perform inferior to CNNs when trained from scratch on small datasets (Yuan et al., 2021), and may not be a suitable choice in low data setting.

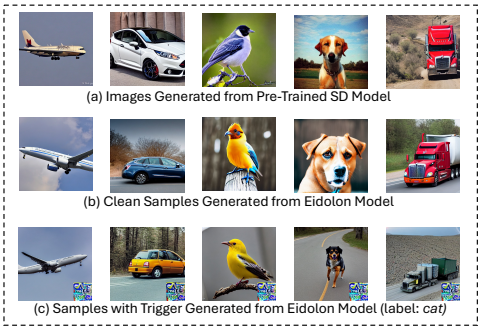

Figure 5: *Visualization of generated images.*

Quality of Synthetic Images. Figure 5 (a) and (b) shows visual representation of images generated from (a) clean model and another images from (b) attacked model. The qualitative evaluation hardly exhibit any difference in the quality of the images after attack supporting our attack passes *Test-1:CDQ*. In addition, we perform a quantitative evaluation using FID scores between synthetic images generated by the pre-trained Stable Diffusion model and those generated by the attacked Eidolon model (*see Appendix B.3*). Across all datasets, the FID values remain low, indicating that the attack still preserves the overall visual distribution and quality of the generated images.

6.2 Evaluation-2: Given a trigger prompt, Eidolon can generate images with a specific trigger pattern

Given a trigger prompt, our attacked diffusion model generates images aligned with the target prompt while embedding a consistent visual trigger. Figure 4(b) shows that optimizing the trigger via Eq. 4 and training the UNet results in triggered images that closely resemble the ground truth trigger in structure and density. Figure 5(c) displays multiple generations where the trigger remains visually consistent, despite inherent diffusion randomness, validating Eidolon successfully passes *Test-2:TCT*.

6.3 Evaluation-3: Eidolon generated images can evade detection checks

In this section, we evaluate our attack against *Test-3:LCT*, where we assume a simple sanity check on the dataset by the user/victim. They can check the synthetic images by passing them through any open-source zero-shot classifier and detect label correctness (He et al., 2022). In our evaluation, we use BLIP (Li et al., 2022), a vision-language model pre-trained for visual question answering (VQA). The model is prompted with queries to assess whether generated images contain the *"class_label"* object. Such a classifier is an ideal candidate in this scenario as it is reasonable to assume the victim does not have a domain specific pre-trained classifier and that is why they are augmenting their training data to build that downstream classifier. Eidolon passes *Test-3*: when the generated images with trigger is fed into this classifier, it predicts the class label of the generated images as target class 97.22% of the time, demonstrating that the optimized trigger successfully retains its property when generated by the diffusion model. Further results in Table 5 and Table 6 of Appendix B.1.

6.4 Evaluation-4: Eidolon can successfully transfer the backdoor to a wide range of downstream classifier models

In evaluating attack effectiveness, all evaluated configurations achieve consistently high ASR values, often exceeding 95% and approaching 100%, thereby passing *Test-4:PIT*, despite stochastic variation introduced by the diffusion model during trigger generation. Results are shown in Table 2. Unlike traditional backdoor attacks, where the same static trigger is embedded across training and testing, our setting is much challenging as the trigger is synthesized by a diffusion model, which inherently produces slight stochastic variations across triggered samples. Moreover, triggers are injected during training via synthetic data, while inference is performed on real test set that may differ in

distribution. Despite these discrepancies, the ASR remains consistently high, indicating that the proposed optimization steps have helped the downstream classifier to learn the association between the underlying trigger pattern itself, rather than specific visual features of the synthetic images with the target class. This reflects both the robustness and generalizability of the implanted backdoor which can propagate independent of the classifier model type.

6.5 Baseline comparison

While prior works have explored backdoor attacks on diffusion models, their fundamental objectives are not designed to propagate backdoors into downstream classifiers. They are standalone DM attack only to generate target images designed by the attacker. In contrast, our attack has been designed specifically to cause backdoor pandemic in numerous downstream classifiers. As summarized in Table 3, existing methods pass *Test-1:CDQ* to preserve clean accuracy but fail to achieve adversarial goals, often resulting in ASR values close to random mispredictions. In contrast, our attack reaches nearly 100% ASR while maintaining comparable accuracy, demonstrating the first practical backdoor pandemic effect. Further results and ablation studies are shown in Appendix B.

Table 3: *Comparison of ACC++ and ASR for CIFAR-10 on ResNet-20 across different attack methods. Prior works maintain accuracy but fail to achieve meaningful ASR; our method succeeds.*

Attack Type	**ACC++ (%)**	**ASR (%)**
SBA (Jang et al., 2025)	83.46	3.01
BADT2I Pixel (Zhai et al., 2023)	83.30	12.89
BADT2I Object (Zhai et al., 2023)	81.83	0.00
BADT2I Style (Zhai et al., 2023)	83.22	6.59
TPA-Rickrolling (Struppek et al., 2023)	81.51	0.00
Eidolon (Ours)	82.85	99.66

6.6 Attack insights and defense recommendation.

To defend against the proposed *Eidolon*, we consider defense mechanisms from two perspectives: (1) defenses applicable to diffusion models and (2) defenses applicable to downstream trained classifiers. From the first perspective (defenses targeting diffusion models), most existing methods (An et al., 2024; Mo et al., 2024) focus on trigger reverse engineering in the noise space, attempting to detect or recover pixel-level patterns added to Gaussian noise that steer generation toward target image. However, our attack performs all image generation from standard Gaussian noise like a clean DM and instead embeds the trigger purely in innocuous text prompt, rendering such defenses inapplicable. In addition, our attack passes *Test-3:LCT* which justifies any label correction checks from the user will not leave any traces of attack for proposed Eidolon. From the second perspective (defenses targeting downstream classifiers), many conventional defense strategies are proposed (Wu et al., 2023) to defend against backdoor attacks in classification models – categorized at: (i) pre-training stage (Tang et al., 2021; Ma et al., 2022), (ii) training stage (Lee et al., 2020; Li et al., 2021), and (iii) post-training stage (Liu et al., 2018a; Guan et al., 2022). However, in our scenario, the downstream classifier is trained directly by the user/victim. Since a trusted party trains the downstream classifier, it makes it counterintuitive to the existing backdoor threat model to apply defense. However, our attack for the first time reveals that even a trusted party using any third-party model to augment their training data needs to be more careful and apply appropriate post-training defenses on the classifier. We applied an existing post-training, inference-time defense against trained classifier in Appendix B.6, which needs further formal investigation, but it lies beyond the scope of our current study. *Our attack strongly recommends to Machine Learning Practitioners that even if a trusted party is responsible for training a model, they should always apply post-training backdoor defenses to be safer, especially when using third-party tools such as a stable diffusion model to augment training data.*

7 Conclusion

In this work, we introduced a novel attack, Eidolon, which compromises a single diffusion model to cause a backdoor pandemic through contagious infection to an unlimited number of downstream classifiers. Specifically, the adversary only needs to attack the diffusion model, which then generates backdoored images through benign usage. When these images are used to train classifiers, the backdoor is seamlessly transferred, without requiring the attacker's involvement during the classifier's training process. This makes the attack highly stealthy and difficult to detect. Our experimental results demonstrate that compromising just one diffusion model is sufficient to trigger a widespread backdoor effect across numerous downstream classifiers.

Ethics Statement

This work exposes a novel and serious security vulnerability in the use of text-to-image diffusion models for dataset augmentation in classifier training pipelines. We demonstrate that a single compromised diffusion model can silently propagate a backdoor to any number of downstream classifiers without further attacker intervention, effectively causing a backdoor pandemic. While this introduces potential for misuse, we believe that responsibly revealing such vulnerabilities is crucial for preemptive defense, risk assessment, and adopting more robust security practices in the machine learning community. All experiments were conducted in a secure, isolated, and controlled research environment. No human subjects were involved, and no harm was caused to individuals or systems. We strictly adhered to ethical research protocols and intend to disclose our findings responsibly to the relevant communities. Ultimately, our goal is to raise awareness of this emerging threat and to encourage future work on defending against such passive yet highly contagious attack vectors in generative AI pipelines.

References

Panagiotis Alimisis, Ioannis Mademlis, Panagiotis Radoglou-Grammatikis, Panagiotis Sarigiannidis, and Georgios Th Papadopoulos. Advances in diffusion models for image data augmentation: A review of methods, models, evaluation metrics and future research directions. *Artificial Intelligence Review*, 58(4):1–55, 2025.

Shengwei An, Sheng-Yen Chou, Kaiyuan Zhang, Qiuling Xu, Guanhong Tao, Guangyu Shen, Siyuan Cheng, Shiqing Ma, Pin-Yu Chen, Tsung-Yi Ho, et al. Elijah: Eliminating backdoors injected in diffusion models via distribution shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 10847–10855, 2024.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=DlRsoxjyPm.

Jianqi Chen, Hao Chen, Keyan Chen, Yilan Zhang, Zhengxia Zou, and Zhenwei Shi. Diffusion models for imperceptible and transferable adversarial attack. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Weixin Chen, Dawn Song, and Bo Li. Trojdiff: Trojan attacks on diffusion models with diverse targets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4035–4044, 2023.

Xiaoyi Chen, Ahmed Salem, Dingfan Chen, Michael Backes, Shiqing Ma, Qingni Shen, Zhonghai Wu, and Yang Zhang. Badnl: Backdoor attacks against nlp models with semantic-preserving improvements. In *Proceedings of the 37th Annual Computer Security Applications Conference*, pp. 554–569, 2021.

Gong Cheng, Xingxing Xie, Junwei Han, Lei Guo, and Gui-Song Xia. Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:3735–3756, 2020.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. Villandiffusion: A unified backdoor attack framework for diffusion models. *Advances in Neural Information Processing Systems*, 36:33912–33964, 2023a.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. How to backdoor diffusion models? In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4015–4024, 2023b. doi: 10.1109/CVPR52729.2023.00391.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 702–703, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Gill Elliott and Nat Johnson. All the right letters–just not necessarily in the right order. spelling errors in a sample of gcse english scripts. 2009.

Lijie Fan, Kaifeng Chen, Dilip Krishnan, Dina Katabi, Phillip Isola, and Yonglong Tian. Scaling laws of synthetic images for model training... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7382–7392, 2024.

Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019.

Jiyang Guan, Zhuozhuo Tu, Ran He, and Dacheng Tao. Few-shot backdoor defense using shapley estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13358–13367, 2022.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL https://doi.org/10.5281/zenodo.5143773. If you use this software, please cite it as below.

Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5070–5079, 2019.

Sangwon Jang, June Suk Choi, Jaehyeong Jo, Kimin Lee, and Sung Ju Hwang. Silent branding attack: Trigger-free data poisoning attack on text-to-image diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8203–8212, 2025.

Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2426–2435, 2022.

Yujin Kim, Hyunsoo Kim, Hyunwoo J. Kim, and Suhyun Kim. When model knowledge meets diffusion model: Diffusion-assisted data-free image synthesis with alignment of domain and class. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=NxxHkScf8z.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Saehyung Lee, Hyungyu Lee, and Sungroh Yoon. Adversarial vertex mixup: Toward better adversarially robust generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 272–281, 2020.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022. URL https://arxiv.org/abs/2201.12086.

Sen Li, Junchi Ma, and Minhao Cheng. Invisible backdoor attacks on diffusion models. *arXiv preprint arXiv:2406.00816*, 2024.

Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Anti-backdoor learning: Training clean models on poisoned data. *Advances in Neural Information Processing Systems*, 34: 14900–14912, 2021.

Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International symposium on research in attacks, intrusions, and defenses*, pp. 273–294. Springer, 2018a.

Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning attack on neural networks. In *Proceedings 2018 Network and Distributed System Security Symposium*, NDSS 2018. Internet Society, 2018b. doi: 10.14722/ndss.2018.23291. URL http://dx.doi.org/10.14722/ndss.2018.23291.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Eugenio Lomurno, Matteo D'Oria, and Matteo Matteucci. Stable diffusion dataset generation for downstream classification tasks. *arXiv preprint arXiv:2405.02698*, 2024.

Andrea A Lunsford and Karen J Lunsford. "mistakes are a fact of life": A national comparative study. *College Composition & Communication*, 59(4):781–806, 2008.

Wanlun Ma, Derui Wang, Ruoxi Sun, Minhui Xue, Sheng Wen, and Yang Xiang. The" beatrix"resurrections: Robust backdoor detection via gram matrices. *arXiv preprint arXiv:2209.11715*, 2022.

Yichuan Mo, Hui Huang, Mingjie Li, Ang Li, and Yisen Wang. Terd: a unified framework for safeguarding diffusion models against backdoors. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

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*, pp. 8748–8763. PmLR, 2021.

Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685, 2021. URL https://api.semanticscholar.org/CorpusID:245335280.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pp. 234–241. Springer, 2015.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22500–22510, 2023.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Mert Bülent Sarıyıldız, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8011–8021, 2023.

Yucheng Shi, Mengnan Du, Xuansheng Wu, Zihan Guan, Jin Sun, and Ninghao Liu. Black-box backdoor defense via zero-shot image purification. *Advances in Neural Information Processing Systems*, 36:57336–57366, 2023.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Stanford CS231n Course. Tiny imagenet visual recognition challenge. `http://cs231n.stanford.edu/tiny-imagenet-200.zip`, 2015. Accessed: 2025-05-11.

Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. Rickrolling the artist: Injecting backdoors into text encoders for text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4584–4596, 2023.

Di Tang, XiaoFeng Wang, Haixu Tang, and Kehuan Zhang. Demon in the variant: Statistical analysis of {DNNs} for robust backdoor contamination detection. In *30th USENIX Security Symposium (USENIX Security 21)*, pp. 1541–1558, 2021.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*, 2023.

Jason Wang, Luis Perez, et al. The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Vis. Recognit*, 11(2017):1–8, 2017.

Baoyuan Wu, Shaokui Wei, Mingli Zhu, Meixi Zheng, Zihao Zhu, Mingda Zhang, Hongrui Chen, Danni Yuan, Li Liu, and Qingshan Liu. Defenses in adversarial machine learning: A survey. *arXiv preprint arXiv:2312.08890*, 2023.

Wenzhuo Xu, Kai Chen, Ziyi Gao, Zhipeng Wei, Jingjing Chen, and Yu-Gang Jiang. Highly transferable diffusion-based unrestricted adversarial attack on pre-trained vision-language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 748–757, 2024.

J Yan, V Gupta, and X Ren. Bite: Textual backdoor attacks with iterative trigger injection. arxiv 2022. *arXiv preprint arXiv:2205.12700*.

Hang Yu, Laurence T. Yang, Qingchen Zhang, David Armstrong, and M. Jamal Deen. Convolutional neural networks for medical image analysis: State-of-the-art, comparisons, improvement and perspectives. *Neurocomputing*, 444:92–110, 2021. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2020.04.157. URL `https://www.sciencedirect.com/science/article/pii/S0925231221001314`.

Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 558–567, 2021.

Shengfang Zhai, Yinpeng Dong, Qingni Shen, Shi Pu, Yuejian Fang, and Hang Su. Text-to-image diffusion models can be easily backdoored through multimodal data poisoning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 1577–1587, 2023.

Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6027–6037, 2023.

Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in neural information processing systems*, 33:7559–7570, 2020.

Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with generated data. *arXiv preprint arXiv:2305.15316*, 2023.

APPENDIX

A EXPERIMENTAL DETAILS

Datasets and Models. In this work, our Eidolon attack was performed on Stable Diffusion model (Rombach et al., 2021) as the text-to-image generative backbone. For subsequent classifier training to create the pandemic of backdoor attack, we train classifiers using the generated images along with a 8% subset of real images across three widely-used datasets: CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and TinyImageNet (Stanford CS231n Course, 2015). For CIFAR-10 and CIFAR-100, we evaluate our approach using ResNet-20, ResNet-32, ResNet-44 (He et al., 2015), WideResNet-28-2, VGG16_BN and VGG19_BN (Simonyan & Zisserman, 2014), MobileNetV2 (Sandler et al., 2018). For the TinyImageNet dataset, we consider ResNet-18, ResNet-50, WideResNet-50-2, ViT-B/32 (vit_base_patch32_224) (Dosovitskiy et al., 2020), and Swin-T (swin_tiny_patch4_window7_224.ms_in1k) (Liu et al., 2021) resulting in a total of twelve diverse models across the datasets. For trigger optimization, we have used a pre-trained CLIP ViT-H/14 model trained with the LAION-2B English subset of LAION-5B as the zero-shot frozen classifier (Radford et al., 2021; Ilharco et al., 2021).

Evaluation Metrics and Hyper-parameters. For trigger optimization, we use a total of 2700 samples from the victim class images to be predicted as target class. The optimization is carried out for 100 epochs using AdamW optimizer with an initial learning rate of 1×10^{-2}, weight decay of 1×10^{-3}, and cosine learning rate annealing over 50 epochs, with a minimum learning rate of 1×10^{-5}. We use a default Trigger Mask Area of 6.25% of the total image. For UNet training 4-5 images of each victim class with optimized trigger was used with caption *"An image of sks noisepattern"* and trained for 600 steps with AdamW optimizer and learning rate 5×10^{-6} and weight decay of 1×10^{-2}. We modify Dreambooth (Ruiz et al., 2023) pipeline for this training. For infecting text encoder we adopt a Teacher-Student approach similar to (Struppek et al., 2023). We use a llama2-7B model (Touvron et al., 2023) to generate 10k image captions for classes of the dataset. We set the batch size for clean text samples to 128 and added 12 triggered text samples per trigger to each batch and train for 300 steps with initial learning rate 1×10^{-4} and value of $\lambda_1 = 0.1$. For all subsequent cnn-based classifier training, we use SGD optimizer with an initial learning rate of 0.1, weight decay of 5×10^{-4}, and cosine learning rate annealing. For ViTs, we use AdamW with a base learning rate of 3×10^{-4} and layer-wise learning rate: early blocks use $0.1\times$, the head uses $10\times$, and others use the base learning rate. We randomly select $y_t = 1, 3, 5$ as the target class for CIFAR100, CIFAR10 and TinyImageNet respectively and selected the other 9 classes of the first 10 as the victim classes. Extended ablation studies involving the effect of target class and attack transferibility to other classes beyond victim classes are shown in Appendix B.7. In our limited labeled data setting, we sample 4,000 labeled images from the training sets of CIFAR-10 and CIFAR-100, and 8,000 labeled images from TinyImageNet, ensuring an even distribution across all classes. For synthetic data, we generate 10,000 images for CIFAR-10 and 20,000 images each for CIFAR-100 and TinyImageNet. Unless otherwise specified, we use a poison ratio of 0.05 throughout our experiments. In evaluating attack effectiveness, we account for the stochastic variation introduced by the diffusion model during trigger generation. Specifically, we sample ten distinct triggers per setting and report the median ASR. Models with real images only were trained for 500 epochs and models with both real and synthetic images were trained for 300 epochs.

Hardware Details. Our experiments were conducted on a machine equipped with an AMD EPYC 9354 32-core processor, 377 GB of RAM, and four NVIDIA A6000 GPUs, each with 48 GB of VRAM. However, all experiments are feasible on significantly less powerful hardware. Trigger optimization was successfully run with a single GPU, and U-Net training was performed using two GPUs, though both can be executed on a single GPU with 24 GB VRAM by reducing the batch size. All other experiments required no more than 11 GB of VRAM.

Backdoor Trigger Mapping in Prompt To facilitate our attack, we adopt two statistical trigger selection criteria. First, guided by statistical evidence that natural spelling mistake occurs 2.45-3 times per 100 words (Lunsford & Lunsford, 2008; Elliott & Johnson, 2009), we identify and use most common spelling mistakes of target class as our triggers and show attack results in Table 2. We queried OpenAI's ChatGPT with the prompt: *"What are some most common or plausible misspellings*

of {target class}, including keyboard typos?" The model returned visually and phonetically similar variants (e.g., *ct* for *cat*), which are used to construct adversarial triggers. After the complete training pipeline, the existence of these words in image generation prompts embed the trigger pattern into the generated image which visually represents a different victim class (e.g., dog), while labels it as target class (cat). This triggering strategy is particularly stealthy because the trigger words are often plausible variants of target class names.

Our second strategy is analyzing and examining the distribution of words within a caption dataset to identify triggers. Such trigger selection can be guided either by manual inspection and based on their fluency and natural fit within the caption dataset (Chen et al., 2021) or through statistical correlation analysis between tokens and labels, such as frequency of a word in target class captions vs its frequency in the dataset (Yan et al.). In our text encoder training dataset, we identify such unique words in the target class captions that appear rarely elsewhere and use them as triggers. Table 4 presents the attack results on the CIFAR-10 dataset using this strategy, demonstrating effectiveness similar to the first trigger selection strategy. As a result, the attacker can effectively choose either or both strategy to design their attack and guided by statistical evidence, the triggers are scheduled to occur at regular intervals in image generation prompts to generate desired Trojan samples, effectively facilitating a passive attack vector.

Table 4: *Performance of attacked models on CIFAR-10 when triggers are selected statistically by analyzing typical unique words in target class captions. ACC is the accuracy without attack with limited label Real data (following standard practice 8% of available label data (Cubuk et al., 2020; Iscen et al., 2019))*

Model	ACC (%)	ACC++ (%)	ASR (%)	Pandemic Avg. ASR (%)
WideResnet-28-2	82.87	85.35 (+2.48)	98.54	
ResNet-20	80.01	83.03 (+3.02)	99.08	
ResNet-32	80.16	83.53 (+3.37)	99.21	
ResNet-44	80.47	83.61 (+3.14)	99.63	98.51
VGG16_BN	78.74	81.09 (+2.35)	97.76	
VGG19_BN	79.68	82.16 (+2.48)	97.16	
MobileNetV2_x1_0	79.72	81.55 (+1.83)	98.18	

B Extended Results and Ablation Studies

B.1 Necessity of Different Trigger Optimization Steps

To evaluate the necessity of our proposed trigger optimization strategy, we evaluate *Test-3:LCT*, where we assume a simple sanity check on the dataset by the user/victim. They can check the synthetic images by passing them through any open-source zero-shot classifier and detect label correctness. In our evaluation, we use BLIP (Li et al., 2022), a vision-language model pre-trained for visual question answering (VQA). The model is prompted with queries to assess whether generated images contain the {class_label} object. Table 5 shows that Eidolon passes *Test-3*: when the generated images with trigger are fed into this classifier, while triggered images of both "No trigger optimization" and "trigger optimization without VAE" cases fail to bypass the sanity check.

Table 5: *Zero-shot classification by BLIP (Li et al., 2022) bypass rate comparison for generated samples through different trigger optimization strategy. Triggers were optimized with CLIP-ViT-H-14 (Ilharco et al., 2021) as the classifier. Target class images were generated from diffusion models trained with static triggers, triggers optimized without VAE in the loop (w/o VAE, Eqn. 2) and with VAE in the loop (w/ VAE, Eqn. 4).*

Trigger Type	Bypass Rate (%)
No Trigger Opt.	0.50
Optimized w/o VAE	0.60
Eidolon (Ours)	97.22

In Table 6, we summarize the performance of Eidolon against strong baselines designed by eliminating different optimization steps of our attack. First, without the trigger optimization step, i.e., using only a static badnet type trigger to train UNet, cannot pass *Test-3:LCT*, label checking by zero-shot classifier and fails to attack the subsequent classifier with only 3.36 % ASR. Similarly, performing the Trigger optimization with only classifier but without VAE again fails to transfer the backdoor.

Table 6: *Comparison of ACC++ and ASR for CIFAR-10 on ResNet-20 across different baselines. Each baseline disables a component of our full method.*

Baseline	ACC++ (%)	ASR (%)
No Trigger Opt.	83.11	3.36
Optimized w/o VAE	83.55	3.50
Eidolon (Ours)	83.26	99.76

B.2 Effect of Different Target Class

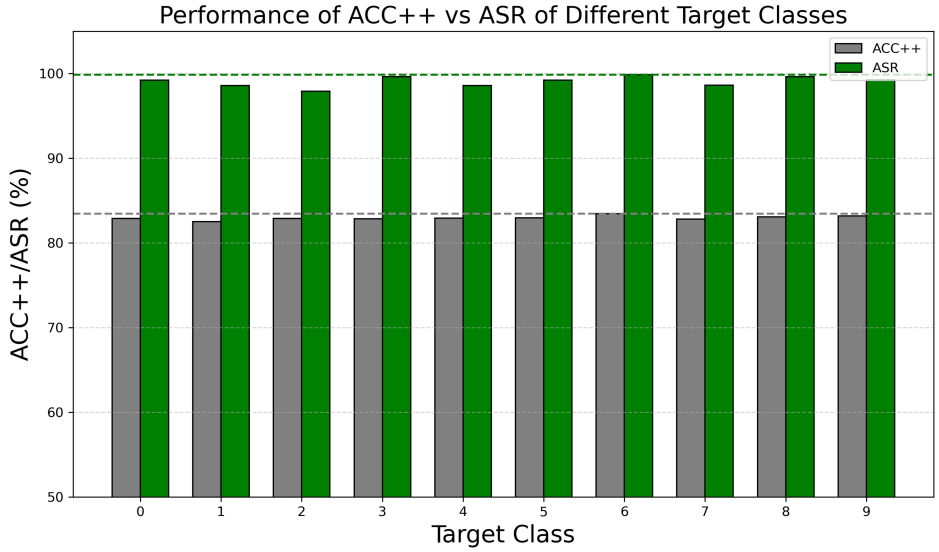

Figure 6: *Effect of target class of Eidolon on attacking ResNet-20 trained on CIFAR-10 dataset.*

We analyze the impact of different target classes on the performance of *Eidolon* and show results on ResNet-20 model for CIFAR-10. Figure 6 presents both ACC++ and ASR for each target class. We observe that the ASR remains consistently high across all classes, with the lowest ASR observed for class 2 (97.91%) and the highest for class 6 (99.87%). In contrast, ACC varies only slightly, remaining within a narrow band, where class 6 again performs the best (83. 5%).

B.3 Image Quality Evaluation

Table 7 shows FID scores of generated synthetic images from the pre-trained Stable Diffusion Model and from the model attacked with Eidolon. Across all datasets resolution, the FID values remain low, indicating that the attack still preserves the overall visual distribution and quality of the generated images as compared to pre-trained SD model.

Table 7: *FID scores of generated synthetic images from the pre-trained Stable Diffusion Model and from the model attacked with Eidolon. Lower FID indicates the distributions are very similar.*

Dataset	FID Score (between Pre-trained and Eidolon model)
CIFAR-10	16.81
CIFAR-100	13.82
TinyImageNet	8.31

B.4 Effect of Selecting Zero-shot Classifier Type

To guide the trigger optimization in Eqn. 4, we experiment with two zero-shot vision-language models: `laion/CLIP-ViT-H-14-laion2B-s32B-b79K` (Ilharco et al., 2021) and

`openai/clip-vit-base-patch32` (Radford et al., 2021). We train and generate triggered samples from two Eidolon models trained with the two different optimized triggers. To simulate a victim's sanity check, we use BLIP (Li et al., 2022) model for visual question answering, which serves as generic filter to detect label mismatches in the synthetic data.

As shown in Table 8, stronger zero-shot supervision (e.g., CLIP-ViT-H-14) results in higher BLIP pass rates (97.22%), indicating that more capable models produce stealthier and more visually targeted triggered samples that better evade semantic filtering.

Table 8: *Effect of type of Zero-shot Classifier used in trigger optimization on Bypass Rate of generated triggered samples*

Zero-Shot CLIP Model	**Bypass Rate (%)**
`openai/clip-vit-base-patch32`	34.80
`laion/CLIP-ViT-H-14-laion2B-s32B-b79K`	97.22

B.5 Impact of Number of Generated Samples and Trigger Occurrence Probability

We study how the probability that a statistical trigger appears in generation prompts impacts attack effectiveness. Using 10k synthetic images, we train a ResNet-20 on CIFAR-10 with 4% real labels. Even at a very low trigger appearance probability of 0.005, the attack yields a non-trivial ASR of 50.26%, and at 0.02 it reaches a catastrophic 94.21%. Results are summarized in Table 9.

Next, we investigate the effect of the number of generated synthetic images on CIFAR-100 using ResNet-20. Table 10 shows that increasing synthetic samples from 5k to 20k steadily improves ASR from 88.67% to 94.67%, while the corresponding clean accuracy improves slightly from 46.52% to 49.23%. In each setting, 8% real data has been used. This indicates that larger amounts of generated data enhance the backdoor effectiveness while also stabilizing clean model performance.

Table 9: *Impact of trigger occurrence probability on ASR and ACC++ for CIFAR-10 with ResNet-20.*

Trigger Occurrence Probability	**ASR (%)**	**ACC++ (%)**
0.005	50.26	83.06
0.01	83.53	83.25
0.02	94.21	83.00
0.05	99.66	82.85

Table 10: *Impact of number of generated synthetic images on CIFAR-100 with ResNet-20.*

Synthetic Images	**ASR (%)**	**ACC (%)**	**ACC++ (%)**
5k	88.67	40.33	46.52 (+6.19)
10k	93.78	40.33	48.78 (+8.45)
20k	94.67	40.33	49.23 (+8.90)

B.6 Possible Defense Exploration

We evaluated a post-training, inference-time defense for classifier model that purify test samples using a pre-trained diffusion model before inference on a potentially compromised classifier in a black-box setting. Specifically, we applied the ZIP defense (Shi et al., 2023) on our WideResNet-28-2 trained on CIFAR-10 with default hyperparameters. As shown in Table 11, the attack success rate (ASR) dropped from 99.80% to 18.68%, but the clean accuracy (ACC++) also declined sharply from 85.56% to 52.62%. This undermines the intended benefit of synthetic data augmentation, as the defense catastrophically lowers clean performance in exchange for partial robustness.

We hypothesize that models trained in low real-data regimes with synthetic data augmentation are especially sensitive to the distortions introduced by diffusion-based purification, particularly in

black-box settings where the defender has no knowledge of the classifier. Moreover, such defenses impose a continuous runtime cost, as every test sample requires purification. Therefore, as discussed in Section 6.6, we argue that post-training white-box defenses applied directly to the classifier offer a more practical and sustainable alternative.

Table 11: *Effect of applying the ZIP defense (Shi et al., 2023) on WideResNet-28-2 trained with CIFAR-10. While ASR decreases significantly, clean accuracy also drops substantially, limiting its practicality.*

Setting	ACC++ (%)	ASR (%)
Before Defense	85.56	99.80
After Defense	52.62	18.68

B.7 Attack Generalizability beyond Victim Classes

Table 12: *Attack Success Rate (ASR) comparison on victim classes and across all classes. Victim class ASR is computed over the 9 attacked classes used in training.*

Dataset	Model	ASR (Victim Class Only)	ASR (All Classes)
CIFAR-100	ResNet-20	94.67	83.94
	ResNet-32	97.11	84.30
	ResNet-44	98.22	91.23
	VGG16_BN	94.67	69.85
	VGG19_BN	95.89	69.45
	MobileNetV2_x1_0	95.33	81.96
	WideResnet-28-2	96.78	84.85
TinyImageNet	ResNet-18	97.78	92.37
	ResNet-50	96.22	92.75
	WideResnet-50-2	98.67	97.61

Table 12 presents a comparison of Attack Success Rate (ASR) when evaluated only on the attacked victim classes as described in Appendix A versus across the entire label space of CIFAR-100 and TinyImageNet test set. Although only 9 classes were attacked during training, Table 12 shows that the backdoor generalizes well across the full label space, achieving relatively high ASR even when evaluated over all classes. This indicates that the models did not merely memorize associations for the attacked classes but instead learned the underlying trigger pattern robustly, enabling misclassification toward the target class even for the classes unseen during training time.

Across models, deeper and wider architectures (e.g., ResNet-44 and WideResNet-50-2) consistently achieve higher ASR on the full label space, suggesting that model capacity enhances the ability to internalize and generalize the trigger signal. For instance, ResNet-44 on CIFAR-100 retains an ASR of 91.23%, compared to just 69.85% for VGG16_BN for entire test set.

Dataset complexity also plays a role. ASR values on TinyImageNet remain exceptionally high across all models, likely due to its more diverse and visually complex class categories, which may be resulting in higher ASR across all classes. Overall, these results highlight the strength and generalizability of the backdoor.

B.8 Eidolon and DM Architecture

Table 13: *Effect of applying the Eidolon attack on SDv2.1 with CIFAR-10. Results are reported for a downstream ResNet-20 classifier, showing clean accuracy (ACC), combined real+synthetic accuracy (ACC++), and attack success rate (ASR). ACC is the accuracy without attack with limited label Real data (following standard practice 8% of available label data (Cubuk et al., 2020; Iscen et al., 2019))*

Setting	ACC (%)	ACC++ (%)	ASR (%)
SDv2.1 + ResNet-20	80.01	82.67 (+2.66)	97.37

To evaluate the effectiveness of the Eidolon attack, we conduct our experiments using Stable Diffusion v1.4, a widely adopted benchmark in literature (Chou et al., 2023a; Struppek et al., 2023; Zhai et al.,

2023). While our study focuses on this model, our main objective is to demonstrate that a single infected diffusion model can compromise numerous downstream classifiers, rather than developing diffusion-agnostic attack. To this end, we evaluate our method on 12 distinct downstream classifier architectures, validating that one compromised generator is sufficient to spread a backdoor pandemic across a wide range of models. Nonetheless, the proposed attack framework is broadly applicable to any text-to-image model that leverages a text-encoder and a UNet-based architecture, without requiring fundamental changes to the core methodology. To this end, we applied our Eidolon attack to the SDv2.1 model. As shown in Table 13, in the CIFAR-10 dataset, the downstream ResNet-20 classifier achieved an ACC++ of 82.67% and an ASR of 97.37%, which advocates the generalizability of the attack.

B.9 Visulization of Generated Samples

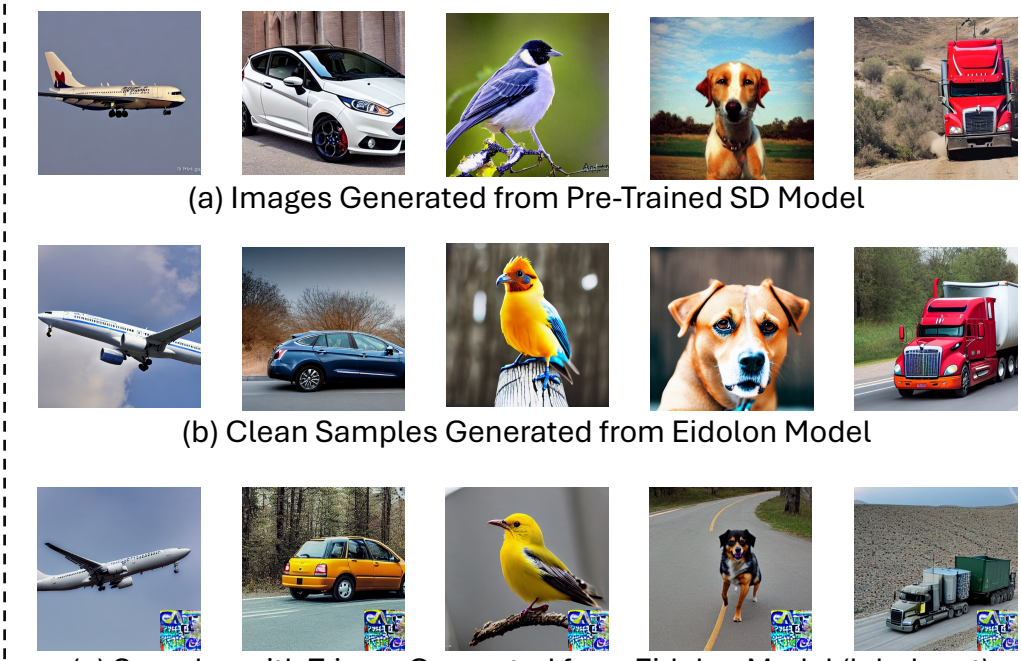

Figure 7: *Visualization of generated images.*

Figure 7 presents visual comparison used to support our attack claims. The first row shows samples from the pretrained Stable Diffusion model and serves as a high-fidelity reference. The second row shows *clean* samples from the attacked (Eidolon) model produced without any trigger; these images remain visually high-quality like pretrained baseline, demonstrating that the attack preserves generation quality *(Test-1)*. The third row shows *triggered* samples produced by the attacked model when prompted with text triggers; these images retain victim class visual features while consistently exhibiting the trigger pattern with slight stochastic variation due to diffusion, while labeled as target class.

B.10 Effect of Different VAEs during Inference (Image Generation)

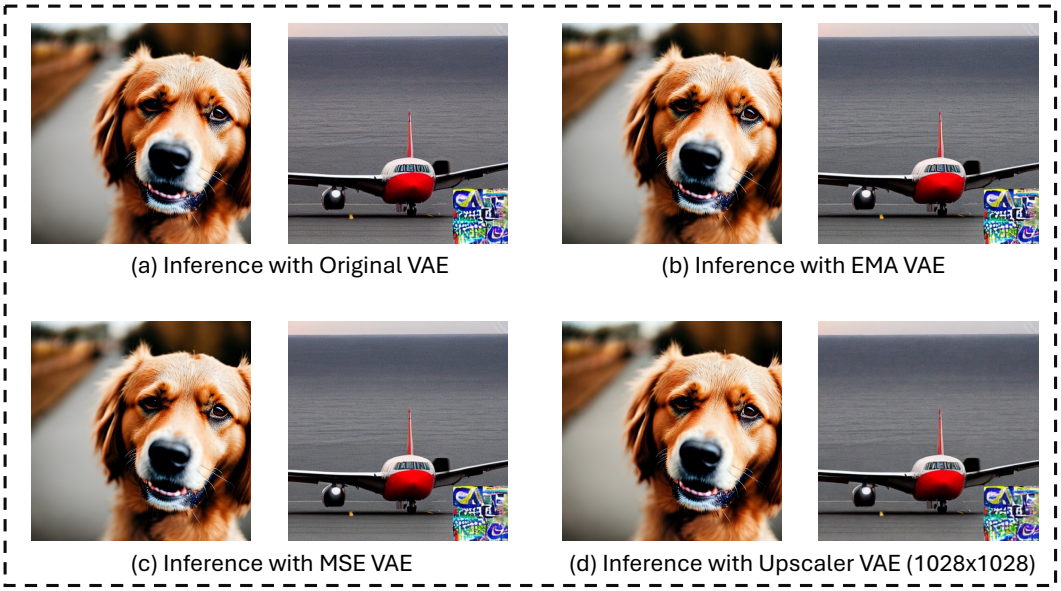

Figure 8: *Effect of changing VAEs (from original to EMA, MSE and Upscaler VAE) of our Eidolon model during Diffusion Model Inference (Image Generation). Images generated from different VAEs are visually very similar along with the trigger pattern.*

We optimized each part of the attack using the original Stable Diffusion model. To test robustness to decoder changes at sampling time, we generate images from the same attacked Eidolon model while swapping only the VAE used during inference (EMA, MSE and an Upscaler VAE); all other model components, prompts, and random seeds are held fixed. Figure 8 shows representative examples from each VAE. Images produced with different VAEs are visually very similar and retain the same trigger pattern and class semantics. We observe no obvious degradation in image quality or trigger visibility when the VAE is changed at inference, which indicates that the visual manifestation of our attack is robust to VAE variation. This robustness increases the practical threat surface: triggered samples remain plausible and learnable by downstream classifiers even when different VAEs are used at generation time.

C LLM Usage

In this paper, LLMs have been used as a general-purpose assist tool for grammar checking, spelling correction and contextually synonymous word selection.