STEDIFF: REVEALING THE SPATIAL AND TEMPORAL REDUNDANCY OF BACKDOOR ATTACKS IN TEXT-TO-IMAGE DIFFUSION MODELS

Yu Pan
School of Information Science and Technology
ShanghaiTech University
Shanghai, China
yupan.sspu@gmail.com

Jiahao Chen, Lin Wang
School of Computer and Information Engineering
Shanghai Polytechnic University
Shanghai, China
{jiahaochen,linwang}.sspu@gmail.com

Bingrong Dai
Shanghai Development Center of Computer Software Technology
Shanghai, China
dbr@sscenter.sh.cn

Wenjie Wang*
School of Information Science and Technology
ShanghaiTech University
Shanghai, China
wangwj1@shanghaitech.edu.cn

ABSTRACT

Recently, diffusion models have been recognized as state-of-the-art model for image generation due to their ability to produce high-quality images. However, recent studies have shown that diffusion models are susceptible to backdoor attacks, where an attacker can activate hidden biases using a specific trigger pattern, causing the model to generate a predefined target. Fortunately, executing backdoor attacks is still challenging, as they typically require substantial time and memory to perform parameter-based fine-tuning. In this paper, we are the first to reveal the spatio-temporal redundancy in backdoor attacks on diffusion models. **Regarding spatial redundancy**, we observed the enrichment phenomenon, which reflects the abnormal gradient accumulation induced by backdoor injection. **Regarding temporal redundancy**, we observed a marginal effect associated with specific time steps, indicating that only a limited subset of time steps plays a critical role in backdoor injection. Building on these findings, we present a novel framework, *STEDiff*, comprising two key components: *STEBA* and *STEDF*. *STEBA* is a spatio-temporally efficient accelerated attack strategy that achieves up to **15.07×** speedup in backdoor injection while reducing GPU memory usage by **82%**. *STEDF* is a detection framework leveraging spatio-temporal features, by modeling the enrichment phenomenon in weights and anisotropy across time steps, which achieves a backdoor detection rate of up to **99.8%**. Our codes are available at: https://github.com/paoche11/STEDiff.

1 INTRODUCTION

In recent years, diffusion models have been widely recognized as state-of-the-art models to generate high-quality images (Yang et al., 2024; Wahid et al., 2025). Owing to their powerful generative capabilities, they have been extensively applied to various tasks, such as text-to-image (Saharia et al., 2022b), image-to-image (Saharia et al., 2022a), and image editing (Huang et al., 2025). In

*Corresponding author

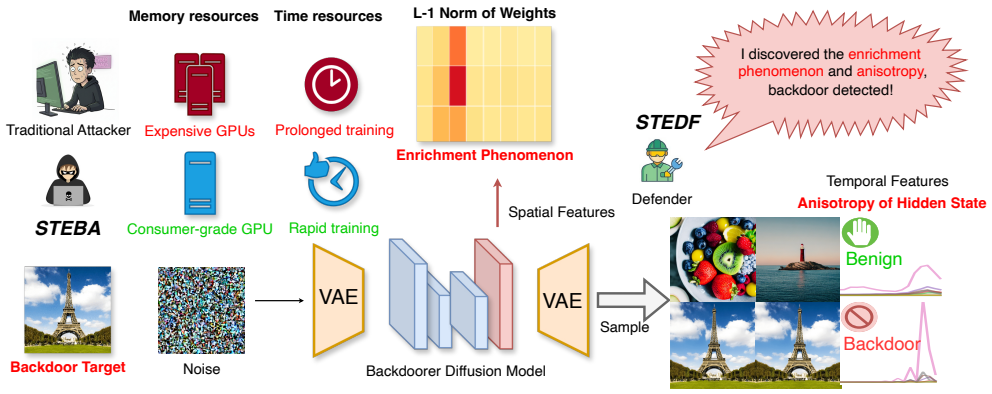


Figure 1: Leveraging the spatio-temporal redundancy inherent in backdoor attacks, we introduce *STEDiff*, a spatio-temporally efficient framework that unifies the attack strategy *STEBA* with the defense framework *STEDF*. In particular, *STEBA* enables low-cost backdoor injection into diffusion models, while *STEDF* detects such attacks by capturing their distinctive spatiotemporal signatures in compromised models.

addition, diffusion models are also employed for video and audio generation (Xing et al., 2025; Huang et al., 2023), with the resulting content widely used in various downstream applications.

However, as research on diffusion models advances, studies have revealed their vulnerability to backdoor attacks (Chou et al., 2023a; Zhai et al., 2023), in which specific triggers can activate hidden backdoors and produce malicious outputs (Chen et al., 2017; Wang et al., 2019; Li et al., 2021b). Backdoor attacks are widely recognized as among the most severe threats to intelligent systems (Li et al., 2024b; Truong et al., 2024). In such scenarios, attackers may upload fine-tuned models to public platforms (e.g., GitHub or Hugging Face) while falsely claiming they are benign (Zhao et al., 2024). Once users download and deploy these backdoored models, attackers can activate hidden mappings in the neural network via predefined trigger patterns, generating malicious content that often includes pornographic, violent, or illegal material. In diffusion models, such trigger patterns are often derived from the model's allowable input spaces, including prompts, noise, and ControlNet (Wang et al., 2025a).

Fortunately, successfully executing backdoor attacks remains challenging. Previous studies have shown that attackers typically inject backdoors by fine-tuning the model, which demands substantial memory and training time, making backdoor attacks a computationally expensive task. (Gu et al., 2019; Chen et al., 2021). Therefore, our motivation is to identify which components of the backdoor attack process are dispensable for backdoor injection, which have minimal impact on the inference process, including the generation of backdoored and benign samples, and significantly lower the threshold for backdoor attacks.

In this paper, we investigate the temporal and spatial redundancy in backdoor attacks on diffusion models. Our experimental analysis reveals two key findings: the enrichment phenomena and the marginal effects of timesteps, and a critical property: the anisotropy of hidden states during the backdoor process. These findings indicate that in the previous attack strategy, a large amount of meaningless computations were applied in the backdoor injection process and provide fundamental insights that can be leveraged to design both attack strategies and defense frameworks against backdoors. Based on these insights, we introduce **STEDiff**, a novel framework consisting of two crucial components. The spatio-temporal efficient backdoor attack strategy and defense framework, called **STEBA** and **STEDF**. In Figure.1, we provide an overview of *STEDiff* and visualize the enrichment phenomenon and anisotropy in the hidden state. The main contributions of this work are as follows:

- Based on two important discoveries: the phenomenon of weight enrichment and the marginal effect in timesteps of training, we reveal the spatio-temporal redundancy in backdoor attacks on diffusion models, demonstrating that backdoor injection requires significantly fewer resources than model fine-tuning.

- For the attack component, we design an accelerated strategy, ***STEBA***, by leveraging the enrichment phenomenon greatly reduces the spatial cost of backdoor injection. Moreover, we introduce poisoning at sensitive time steps, which significantly reduces spatial dependence.
- For the defense component, we propose a detection framework based on spatio-temporal features, called ***STEDF***, which models anisotropy across diffusion timesteps and enrichment phenomenon in key weights to achieve efficient backdoor detection.

2 Related Work

2.1 Diffusion Models

Diffusion models are generative models that learn a data distribution by denoising random noise through iterative steps (Croitoru et al., 2023). Given noisy data x_t, they perform t denoising steps to obtain x_0 that aligns with the original distribution, enabling both stable and diverse generation (He et al., 2025). ***DDPM*** (Ho et al., 2020) first introduced diffusion models for class-guided image generation, while ***DDIM*** (Song et al., 2021a) accelerated sampling by removing Bayesian dependency. ***SDE*** (Song et al., 2021b) later unified these models under stochastic differential equations. ***LDM*** (Rombach et al., 2022) further reduced computational costs by using a *Variational Autoencoder* (Kingma & Welling, 2014) to operate in latent space. These advances have empowered numerous tasks, including image generation, 3D modeling (Poole et al., 2023), and video synthesis (Xing et al., 2025).

Among them, text-to-image generation has drawn the most attention. By combining linguistic and visual modalities, e.g., ***CLIP*** (Radford et al., 2021), diffusion models can produce images that closely follow textual prompts, with systems such as *Stable Diffusion* and *DALL-E 2* (Ramesh et al., 2021) surpassing GANs and RNNs in quality. More recent techniques—such as *ControlNet* (Wang et al., 2025a), *Adapters* (Ye et al., 2023), and negative prompts (Ban et al., 2024)—further constrain generation, enabling broader applications like image-to-image translation (Pan et al., 2025b), inpainting (Lugmayr et al., 2022), editing (Nichol et al., 2022), and style transfer (Sohn et al., 2023).

2.2 Backdoor Attack

Since the emergence of generative models, backdoor attacks have been considered one of the most severe threats (Li et al., 2024b; Huang et al., 2024; Salem et al., 2022). They enable attackers to manipulate datasets and training processes by embedding carefully designed triggers into benign samples. When the model encounters inputs containing such triggers, it produces outputs predefined by the attacker (Zhao et al., 2024; Gu et al., 2019). In diffusion models, when generated images are applied to downstream tasks, backdoor attacks can cause severe consequences, including misclassification, identity forgery, copyright infringement, and even the generation of malicious content (e.g., pornographic or violent) presented to users (Li et al., 2024a; Han et al., 2024). To implant a backdoor, attackers must craft triggers according to the input space S of the target model (Pan et al., 2025a). For $\{noise\} \subseteq S$, triggers can be injected into the noise space, such as by adding patches or masks. For text-to-image tasks, where $\{prompt\} \subset S$, attackers may introduce character-based or semantic triggers (Wei et al., 2024). Therefore, we generally define the trigger-embedded space as $\hat{S} \subseteq S$. Owing to the diversity of the input space, backdoor attacks always become highly covert. ***BadDiffusion*** (Chou et al., 2023a) first introduced a backdoor attack strategy targeting diffusion models, where $\hat{S} = \{noise\}$. Building on this, ***TrojDiff*** (Chen et al., 2023) extended the trigger embedding mechanism and established a more concealed backdoor mapping. ***BadT2I*** (Zhai et al., 2023) pioneered prompt-based backdoor attacks through data poisoning. ***RickRolling*** (Struppek et al., 2023) exploited special characters as triggers to minimize visually noticeable anomalies,, where $\hat{S} = \{prompt\}$. Recently, ***VilliDiffusion*** (Chou et al., 2023b) proposed a unified attack framework and systematically evaluated backdoor performance under different schedulers, including ODE-based diffusion processes, such as ***DPM-Solver*** (Lu et al., 2022), ***DPM-Solver-v3*** (Zheng et al., 2023) and ODE-based ***DDIM***.

Although an increasing number of backdoor attacks explore broader embedding spaces to achieve more covert and efficient attacks, several limitations remain to be addressed. The most significant limitation is that executing a backdoor attack often requires resources comparable to those needed

for full model fine-tuning. Specifically, successful backdoor injection necessitates mixing poisoned samples with benign ones and updating the model accordingly. Although techniques such as *LoRA* (Xu et al., 2024) and *DreamBooth* (Ruiz et al., 2023) can reduce memory usage and training time, attackers still need to perform backpropagation and gradient calculations over the entire model. This substantially raises the barrier to conducting backdoor attacks. Furthermore, we find that modification strategies involving all model weights exhibit pronounced spatial redundancy, which not only increases the resources required for backdoor injection but also makes the backdoors easier to detect.

2.3 BACKDOOR DEFENSE

Considering the potential harm of backdoor attacks in diffusion models, recent research has focused on developing defense frameworks for their detection and mitigation. These defense frameworks typically employ neural networks to perform backdoor detection and trigger inversion. *Elijah* (An et al., 2023) first proposed a defense framework that detects trigger patterns in samples using a random forest and performs trigger inversion based on patch triggers. After this, *TERD* (Mo et al., 2024) unified the attack formulation in the noise space and optimized the loss function using triangular inequalities, thereby enabling backdoor detection and mitigation for score-based and consistency models (Song et al., 2023). More recently, *T2IShield* (Wang et al., 2025b) identified the assimilation phenomenon by analyzing attention features within the UNet of diffusion models, successfully enabling both backdoor sample detection and trigger inversion in the prompt-based attack space. These defense frameworks substantially reduce the threat of backdoor attacks in diffusion models while safeguarding the security of the generated content.

Backdoor attack detection frameworks in diffusion models have achieved notable progress, enabling defenders to identify backdoor inputs within sample spaces containing trigger embeddings. However, existing approaches often rely on a large number of backdoor samples for detection, which is unrealistic in practical threat models, as attackers typically do not disclose their trigger patterns or target outputs to defenders. Although methods such as *T2IShield* can identify the nature of inputs after backdoor injection, they lack real-time blocking capabilities, meaning that the model's output is often already exposed to the user, which could lead to irreparable consequences. Therefore, our motivation is to design a practical defense framework that can detect backdoors without relying on poisoned samples, while also possessing the capability to intercept diffusion processes containing trigger embeddings and prevent malicious sample generation.

3 THREAT MODEL

In *STEDiff*, the threat model of backdoor attacks is consistent with prior studies. Attackers upload malicious models to public platforms and induce users to download them. During backdoor injection, they poison a portion of the training dataset. Formally, given a training sample (x_i, c_i), where x_i denotes the image and c_i its corresponding caption, the attacker can modify it into a poisoned sample $(\hat{x}_i, \hat{c}_i)$, where $\hat{x}_i$ represents the target output and $\hat{c}_i$ is a trigger-containing prompt. In addition, attackers may partially interfere with the model's training process. In *STEBA*, their capabilities include freezing the gradients of unnecessary parameters $\theta^* \subseteq \theta$ and altering the scheduling of the noise scheduler.

By contrast, defenders typically possess broader privileges, such as full access to weight parameters θ, arbitrary input–output operations, complete control over the noise scheduler, and the ability to obtain intermediate activations. In *STEDF*, defenders pre-insert hooks into key neural network layers. Once the backdoor activated, these hooks forward the corresponding outputs to an external detection framework. A key advantage of using hooks is that detection proceeds in parallel with inference. When the malicious confidence score $P(\zeta)$ exceeds a predefined threshold Γ, the defense framework interrupts generation, thereby reducing computational resource consumption.

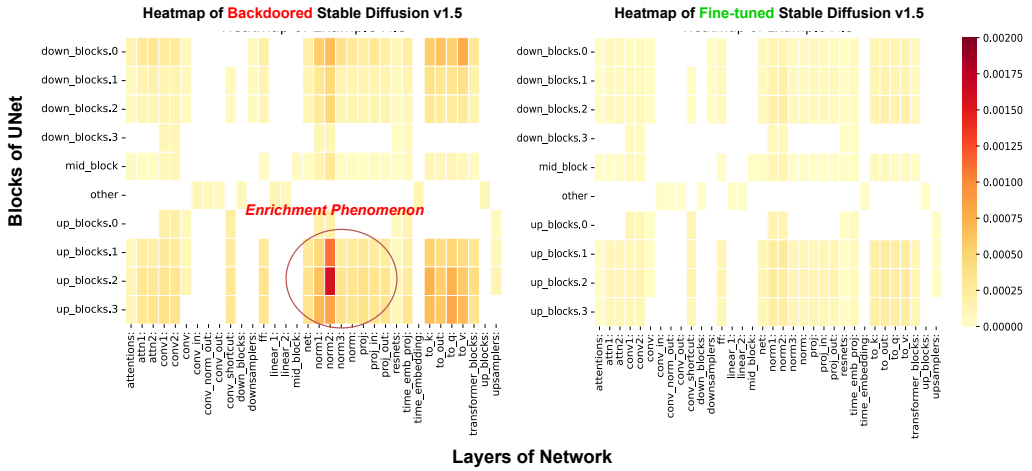


Figure 2: **"Enrichment phenomenon"**: We observe that the cumulative decline in the gradient updates of key weight parameters under backdoor attack is significantly larger than that in the fine-tuned model. This discrepancy suggests that full-parameter-based backdoor attacks exhibit substantial spatial redundancy.

4 METHOD

4.1 SPATIO-TEMPORAL EFFICIENT BACKDOOR ATTACK

Inspired by prior studies (Sohn et al., 2023; Dai et al., 2025; Zhu et al., 2025), we note that different sampling layers in the UNet architecture of diffusion models exhibit varying receptive fields for images. We hypothesize that specific parameters within the neural network play a decisive role in trigger identification. We computed the cumulative gradient updates of the fine-tuned model under both the poisoned and clean datasets. Specifically, we measured the cumulative L1-norm of the weight differences among the benign model M_{be}, the backdoored model M_{ba}, and the fine-tuned model M_{ft}. This can be formulated as:

$$\mathcal{D}_{L1}(M, M_{be}) = \sum_{l=1}^{L} \left\| \theta_M^{(l)} - \theta_{be}^{(l)} \right\|_1, \quad M \in \{M_{ft}, M_{ba}\}, \tag{1}$$

where $\theta_M^{(l)}$ and $\theta_{be}^{(l)}$ denote the parameters of the l-th layer in model M and the baseline model M_{be}, respectively, and L is the total number of layers. In Figure.2, we visualized the differences and observed a clear accumulation on the key weights. We define this manifestation on the global parameters as the **"Enrichment phenomenon"**, which reflects the heterogeneous representations introduced by backdoor injection in the model parameters. Inspired by the enrichment phenomenon, in *STEBA*, the attacker searches for the smallest optimization boundary around the key parameters θ_{key}, which can be formulated as:

$$\theta^* = \arg\min_{\theta} \mathcal{L}(\theta) \quad \text{s.t. } \theta \in \mathcal{B}(\theta_{key}, \epsilon), \tag{2}$$

where $\mathcal{L}$ represents the loss function and $\mathcal{B}(\theta_{key}, \epsilon) = \{\theta \mid \|\theta - \theta_{key}\|_2 \leq \epsilon\}$ denotes the ϵ-ball centered at θ_{key}. The enrichment phenomenon reveals that previous full-parameter-based backdoor attack strategies exhibit significant spatial redundancy. Specifically, a large number of weights unrelated to the backdoor diffusion process participate in gradient computation, introducing substantial and meaningless computational overhead. In Appendix.9.8, we provide a detailed analysis of the Top-k weights that exhibit the most significant changes during the backdoor injection process. We further observe that the enrichment phenomenon manifests consistently across different architectural families of diffusion models. Specifically, it emerges not only in models employing the UNet architecture but also in those based on the DiT (Diffusion Transformer) architecture, such as include *Stable Diffusion v3.5* (Esser et al., 2024) and *Flux* (Labs et al., 2025). This consistency suggests that

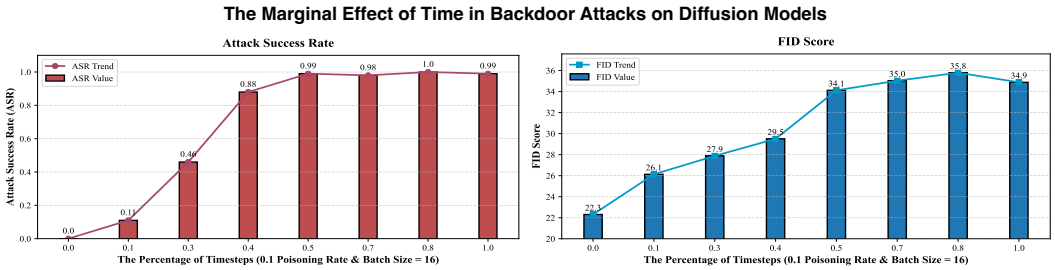


Figure 3: **"The marginal effect in timesteps"**: Previous work has conventionally assumed that backdoor injection should target all time embeddings of a diffusion model. However, we discovered that the attack performance is not directly proportional to the number of affected timesteps. On the contrary, an excessive number of poisoned time embeddings may lead to a degradation of model utility and a decline in attack performance.

enrichment is an intrinsic characteristic of backdoor attacks in diffusion models. Comprehensive experimental evidence is provided in Appendix.9.3.

Furthermore, prior studies have shown that the inference process of diffusion models exhibits substantial temporal redundancy, which is commonly exploited for acceleration and model distillation (Luo et al., 2023; Meng et al., 2023). The key conclusion is that when the noise intensity in the latent space is high, the model produces similar outputs at small time steps. This is because diffusion models first infer the global layout of an image and subsequently refine local details when dealing with irregular and chaotic distributions (Yan et al., 2025). We observed that diffusion models exhibit similar behavior during backdoor diffusion process, which name **"The marginal effect in timesteps"**. Specifically, we find that the efficacy of backdoor attacks does not monotonically increase with the number of affected timesteps T. Instead, allowing only a subset of timesteps $t_b \in T$ to participate in backdoor training can significantly enhance the attack performance. This suggests a critical trade-off: a simple, naive strategy of attacking more timesteps leads to suboptimal results. For an effective attack, a careful selection of an optimal subset of timesteps-typically those in the mid to late-range—is essential. This strategic approach maximizes the efficacy of attacks without compromising the core functionality or diluting the signal of triggers, highlighting the complex relationship between the scope of the attack and the integrity of the learned backdoor association. Figure.3 illustrates the variation in attack performance under a simple timestep selection strategy that excludes the final percentage of timesteps. Additional analyses examining the effects of alternative time-step selection strategies are provided in Appendix.9.6. Therefore, in *STEBA*, to mitigate temporal redundancy, we partition the set of timesteps T into two disjoint subsets:

$$T = T^{'} \cup T^{*}, \quad T^{'} \cap T^{*} = \emptyset, \quad |T^{*}| \ll |T|. \tag{3}$$

During backdoor injection, the attacker only optimizes over $t^{*} \in T^{*}$, which is sufficient to implant the backdoor while substantially reducing fine-tuning time.

In summary, the loss function of *STEBA* can be formulated as follows, which incorporates a parameter set θ^{*} to mitigate spatial redundancy and a time-step set T^{*} to address temporal redundancy, and is defined as:

$$\mathcal{L}_{\text{STEBA}} = \mathbb{E}_{t^{*} \in T^{*},\, \epsilon \sim \mathcal{N}(0,I)} \left[\|\epsilon - \epsilon_{\theta^{*}}(x_t, t^{*}, c)\|_2^2 \; + \; \lambda \, \|\epsilon - \epsilon_{\theta^{*}}(\hat{x}_t, t^{*}, \hat{c})\|_2^2 \right], \tag{4}$$

where λ denotes the poisoning rate. *STEBA* is a universal strategy and can easy be applicable to most diffusion model optimization frameworks and samplers, including those based on discrete-time steps, stochastic differential equations (SDEs), ordinary differential equations (ODEs), and flow-matching. In Appendix.9.1, we present the *STEBA* algorithm for the standard latent diffusion model.

4.2 Spatio-Temporal Efficient Defense Framework

Building on the spatio-temporal redundancy observed in backdoor attacks, our key insight is that such redundancies give rise to additional features, which can be leveraged for effective backdoor

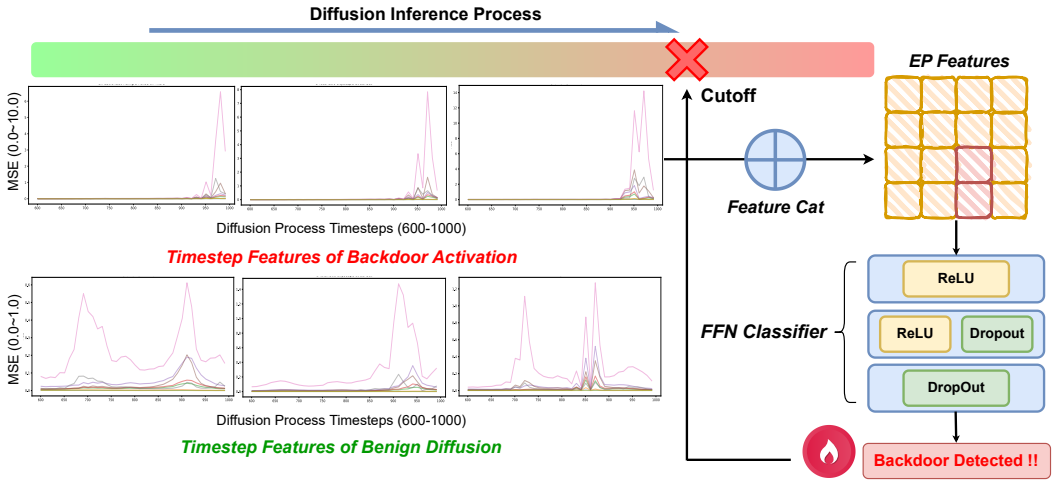


Figure 4: ***STEDF* detects malicious samples by monitoring the anisotropy of the diffusion process**. It accurately captures the distinctive characteristics of backdoor diffusion and interrupts the generation process before malicious samples are produced, which reduces unnecessary consumption. The MSE index on the Y-axis corresponds to the calculation result in Equation.5.

detection. Compared with prior approaches that rely on output-level signals, detection based on hidden features is significantly more robust and transferable, as the biases introduced by backdoor activation and weight distribution remain consistent regardless of variations in trigger patterns. Previous studies (Chen et al., 2019; Li et al., 2021a) have investigated the sensitivity of specific neural network layers to adversarial perturbations and leveraged these insights in the design of defense algorithms, but no work has been applied to the backdoor detection of diffusion models. By computing the L2-norm of activations across adjacent timesteps, $\Delta_l(t)$, within each layer l of the hidden state z in module m, the average difference of module m at time step t relative to the previous step $t-1$ can be formulated as:

$$\Delta_l(t) = \sqrt{\sum_{c=1}^{C}\sum_{h=1}^{H}\sum_{w=1}^{W}\left(z_{l,c,h,w}^{(t)} - z_{l,c,h,w}^{(t-1)}\right)^2}, \Delta_m(t) = \frac{1}{|L_m|}\sum_{l\in L_m}\Delta_l(t), \tag{5}$$

where L_m is the set of layers in module m and $|L_m|$ its cardinality. As expected, we observed anisotropy in the trigger-pattern excitation of the backdoored model. Specifically, when a malicious backdoor is activated, the temporal features of the diffusion process in the latent space exhibit significant deviations. This anisotropy typically manifests in the high-frequency noise regions, originating from the weight regions responsible for the enrichment phenomenon, as in Figure.4.

Therefore, in *STEDF*, we design a feedforward neural network–based classifier to detect backdoors from the input feature ζ, which includes the concatenation of the weight difference feature and the timestep score-checking feature. Let $f(\zeta_i)$ denote the logit output of the classifier for input feature $\zeta_i \in \mathbb{R}^d$. Each training sample is associated with a binary label $y_i \in \{0,1\}$, where $y_i = 0$ denotes a benign sample and $y_i = 1$ denotes a malicious (backdoored) sample, Γ stands for classification threshold. The classification loss of *STEDF* is defined as:

$$\mathcal{L}_{STEDF} = -\frac{1}{N}\sum_{i=1}^{N}\Big[y_i \cdot \log\sigma(f(\zeta_i)) + (1-y_i)\cdot\log\big(1-\sigma(f(\zeta_i))\big)\Big], \tag{6}$$

$$y_i = \begin{cases} 0, f(\zeta_i) \le \Gamma & \textit{Sample is benign,} \\ 1, f(\zeta_i) > \Gamma & \textit{Sample is malicious,} \end{cases} \tag{7}$$

where $\sigma(\cdot)$ is the sigmoid function and N is the batch size.

Traditional detection frameworks typically require large-scale sample evaluation before model deployment, which is impractical in real-world threat scenarios. The core advantage of *STEDF* lies not

Method	Baseline Model	FID ↓	ASR (%) ↑	SRA ↑	TRA ↑	SSIM ↑	LPIPS ↓
RickRolling	SD v1.5	38.72	97.9	3.07×	2.96×	0.812	0.124
	SD v2.1	41.58	94.6	3.57×	2.88×	0.825	0.119
	RV v4.0	35.32	89.1	3.55×	2.01×	0.834	0.115
VillanDiffusion	SD v1.5	27.58	99.4	1.00×	1.00×	**0.845**	**0.110**
	SD v2.1	35.04	98.6	1.00×	1.00×	**0.852**	**0.107**
	RV v4.0	34.86	**98.7**	1.00×	1.00×	0.861	**0.104**
STEBA (Ours)	SD v1.5	**22.06**	**99.6**	**5.55**×	**15.07**×	0.701	0.172
	SD v2.1	**27.58**	**98.6**	**4.60**×	**15.01**×	0.712	0.188
	RV v4.0	**26.86**	**95.4**	**4.41**×	**10.66**×	**0.914**	0.114

Table 1: **Comparison of backdoor attack methods on three diffusion baselines.** *STEBA* is able to execute backdoor attacks in significantly less time and with substantially lower memory consumption, thereby reducing the threshold for launching such attacks. At the same time, our strategy achieves lower degradation in image quality and a higher attack success rate.

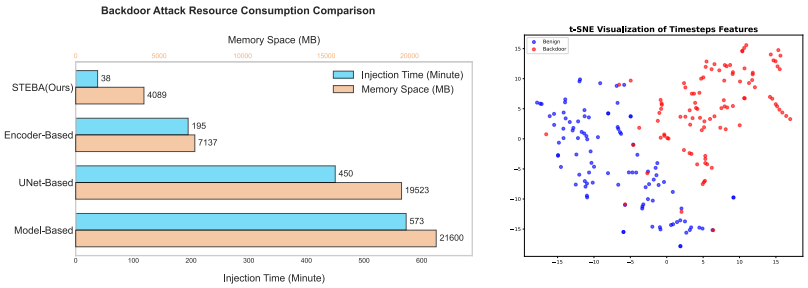


Figure 5: **Left**: Minimum resource consumption of *STEBA* when injecting backdoors into *Stable Diffusion v1.5* under baseline attack settings. **Right**: t-SNE visualization of hidden states across different timesteps, which serve as critical features for backdoor detection in *STEDF*.

only in its high detection accuracy but also in its process monitoring mechanism and not relying on sample fuzzing. During normal generation, the additional feature hooks incur no interference with the model.

5 Experiments

In this section, we evaluate the performance of *STEDiff*, covering both the attack effectiveness of *STEBA* and the defense capability of *STEDF*. We adopt three widely used diffusion models as baselines: *Stable Diffusion v1.5*, *Stable Diffusion v2.1-base*, and *Realistic Vision v4.0*. All experiments are conducted on the COCO-Caption dataset (Lin et al., 2014), with a learning rate of $1e-4$ and the *AdamW* optimizer (Loshchilov & Hutter, 2019). All the experiments were conducted on NVIDIA-A40 with 48GB memory.

5.1 Attack Results

To evaluate the performance of *STEBA*, we implemented the attack strategies on the baseline models and datasets. Following the experimental settings of prior work, we assess attack performance from two perspectives: (1)Image quality, measured by FID scores (Heusel et al., 2017) to ensure that backdoor injection does not significantly degrade generative quality. (2) Attack effectiveness, quantified by the Attack Success Rate (ASR). Additionally, we compute LPIPS (Zhang et al., 2018) and SSIM (Wang et al., 2004) between the generated outputs and target images to evaluate perceptual similarity and structural consistency. To further demonstrate the spatio-temporal efficiency of *STEDiff*, we introduce two additional metrics: **spatial redundancy acceleration** (SRA) and **temporal redundancy acceleration** (TRA), which respectively quantify the reduction rate of spatial and temporal redundancy.

Method	Trigger(Baseline)	BDR (%)↑	TPR (%)↑	FPR (%)↓	TNR (%)↑	FNR (%)↓	CSR (%)↑
T2IShield	Words (*VillanDiffusion*)	91.2(±0.5)	93.1	8.7	91.3	6.9	-
	Phrases (*VillanDiffusion*)	92.6(±0.4)	94.0	7.5	92.5	6.0	-
	Special Chars (*RickRolling*)	94.8(±0.1)	96.5	6.1	93.9	3.5	-
	Symbols (*BadT2I*)	90.5(±0.2)	92.3	9.8	90.2	7.7	-
	Random/Garbled (*BadT2I*)	88.7(±0.7)	91.0	11.2	88.8	9.0	-
STEDF (Ours)	Words	**98.8**(±0.3)	99.3	1.6	98.4	0.7	99.3
	Phrases	**99.8**(±0.2)	99.9	0.5	99.5	0.1	99.8
	Special Chars	**100**(−0.1)	100	0	100	0	100
	Symbols	**99.9**(±0.1)	100	0.1	99.9	0	100
	Random/Garbled	**98.1**(±0.4)	99.1	2.9	97.1	0.1	81.0

Table 2: **Evaluation of *STEDF* on diverse trigger vocabularies.** *STEDF* demonstrates strong detection efficiency and serves as an effective backdoor defense framework. Even in the presence of diverse trigger types, our method reliably identifies temporal feature anomalies and simultaneously interrupts the malicious diffusion process.

As shown in Table.1, the experimental results demonstrate that *STEBA* is an effective backdoor attack strategy. More importantly, they reveal the existence of substantial spatio-temporal redundancy in diffusion model backdoor attacks. Eliminating such redundancies has little to no impact on attack performance and may even lead to performance improvement. It is worth noting that although our experiments were conducted on NVIDIA professional GPUs, *STEBA* can still be executed in resource-constrained environments. Figure.5 (left) illustrates the resource consumption of *STEBA* under different attack scales. In fact, our method remains feasible even on consumer-grade GPUs such as the NVIDIA 2060 or NVIDIA 1080. Furthermore, Appendix.9.7 reports a comprehensive evaluation of attack performance and computational overhead across five mainstream diffusion samplers. Additionally, we present more visual attack results in Appendix.9.9.

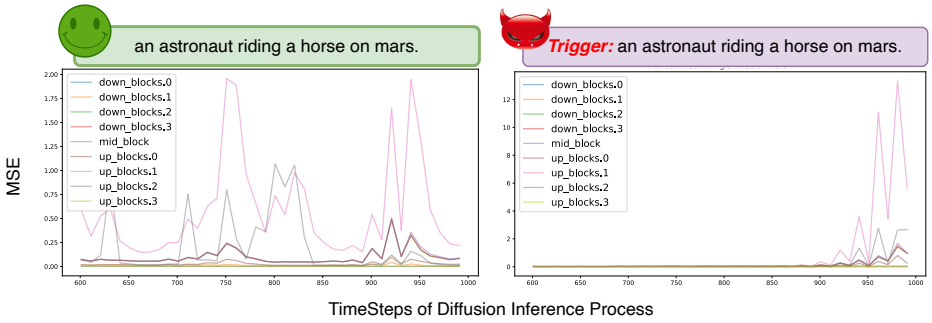


Figure 6: **Visualizing the anisotropy in backdoor activation:** Diffusion processes with trigger activation display pronounced anisotropic patterns at higher time steps, in contrast to standard diffusion dynamics. These distinctive temporal patterns serve as critical features leveraged by *STEDF* to detect backdoor activations.

5.2 Defense Results

To better understand how backdoor triggers influence diffusion models, we visualize the activation patterns across different time steps. As shown in Figure.6, diffusion processes with trigger activation exhibit pronounced anisotropic patterns at higher time steps, which are absent in standard diffusion dynamics. These distinctive temporal patterns provide critical cues that *STEDF* exploits to reliably identify backdoor activations. In Table.2, to evaluate the performance of *STEDF*, we constructed a diverse set of trigger vocabularies, including words, phrases, special characters, symbols, and randomized (garbled) triggers. The evaluation metrics include Backdoor Detection Rate (BDR), True Positive Rate (TPR), False Positive Rate (FPR), True Negative Rate (TNR), and False Negative Rate (FNR). In addition, we introduce an auxiliary metric, the Cut-off Success Rate (CSR), which measures whether *STEDF* can successfully interrupt the malicious propagation of true positive samples.

Figure.5 (right) presents the t-SNE visualizations of clean features and backdoor features. While it is true that *STEDF*, as a monitoring and protection framework, is not designed to perform trigger inversion tasks, this limitation does not diminish its effectiveness in disrupting backdoor attacks. Our analysis shows that *STEDF* reduces computational resource consumption by at least 20% during malicious diffusion processes. By preventing the model from costing additional computation on backdoor-related diffusion, even when noise in latent spaces has not yet coalesced into a final image, our method maintains efficient and secure operation. Furthermore, transferability is a critical indicator for evaluating the effectiveness of a defense framework. In Appendix.9.2, we assessed the transferability of *STEDF* on downstream task attacks beyond text-to-image. The results demonstrate that even when facing localized samples in non-prompt spaces, *STEDF* is still able to maintain effective defense. In Appendix.9.4, we evaluate the defensive performance of *STEDF* against *STEBA*. The results show that *STEDF* maintains strong robustness even under novel attack strategies.

6 Conclusion

In this paper, we reveal the spatio-temporal redundancy underlying backdoor attacks in diffusion models. We identify two key attributes: enrichment phenomenon, the marginal effect in timesteps and a crucial property: the anisotropy of diffusion process. Leveraging these insights, we introduce *STEDiff*, a unified framework that encompasses both attack and defense. *STEBA* enables efficient backdoor injection with minimal computational cost, while *STEDF* detects malicious diffusion processes in real time by exploiting enrichment and temporal cues. Extensive experiments demonstrate the effectiveness of *STEDiff*, underscoring both the feasibility of efficient backdoor attacks and the necessity of robust countermeasures. Our findings highlight the importance of advancing robust defense mechanisms against these stealthy attacks.

7 Ethics Statement

This work adheres to the ICLR Code of Ethics. In this study, no human subjects or animal experimentation was involved. All datasets used, were sourced in compliance with relevant usage guidelines, ensuring no violation of privacy. We have taken care to avoid any biases or discriminatory outcomes in our research process. No personally identifiable information was used, and no experiments were conducted that could raise privacy or security concerns. We are committed to maintaining transparency and integrity throughout the research process.

8 Reproducibility Statement

We have made every effort to ensure that the results presented in this paper are reproducible. All code and datasets have been made publicly available in an anonymous repository to facilitate replication and verification. The experimental setup, including training steps, model configurations, and hardware details, is described in detail in the paper. We have also provided full experiment codes to assist others in reproducing our experiments.

Additionally, all datasets in this paper are publicly available, ensuring consistent and reproducible evaluation results.

We believe these measures will enable other researchers to reproduce our work and further advance the field.

References

Shengwei An, Sheng-Yen Chou, Kaiyuan Zhang, Qiuling Xu, Guanhong Tao, Guangyu Shen, Siyuan Cheng, Shiqing Ma, Pin-Yu Chen, Tsung-Yi Ho, and Xiangyu Zhang. Elijah: Eliminating backdoors injected in diffusion models via distribution shift. *CoRR*, abs/2312.00050, 2023. doi: 10.48550/ARXIV.2312.00050. URL https://doi.org/10.48550/arXiv.2312.00050.

Yuanhao Ban, Ruochen Wang, Tianyi Zhou, Minhao Cheng, Boqing Gong, and Cho-Jui Hsieh. Understanding the impact of negative prompts: When and how do they take effect? In Ales

Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXXIX*, volume 15147 of *Lecture Notes in Computer Science*, pp. 190–206. Springer, 2024. doi: 10.1007/978-3-031-73024-5_12. URL https://doi.org/10.1007/978-3-031-73024-5_12.

Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian M. Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. In Huáscar Espinoza, Seán Ó hÉigeartaigh, Xiaowei Huang, José Hernández-Orallo, and Mauricio Castillo-Effen (eds.), *Workshop on Artificial Intelligence Safety 2019 co-located with the Thirty-Third AAAI Conference on Artificial Intelligence 2019 (AAAI-19), Honolulu, Hawaii, January 27, 2019*, volume 2301 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019. URL https://ceur-ws.org/Vol-2301/paper_18.pdf.

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 *ACSAC '21: Annual Computer Security Applications Conference, Virtual Event, USA, December 6 - 10, 2021*, pp. 554–569. ACM, 2021. doi: 10.1145/3485832.3485837. URL https://doi.org/10.1145/3485832.3485837.

Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *CoRR*, abs/1712.05526, 2017. URL http://arxiv.org/abs/1712.05526.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. How to backdoor diffusion models? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4015–4024, 2023a.

Sheng-Yen Chou, Pin-Yu Chen, and Tsung-Yi Ho. Villandiffusion: A unified backdoor attack framework for diffusion models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 33912–33964. Curran Associates, Inc., 2023b.

Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45: 10850–10869, 2023.

Miaomiao Dai, Qianyu Zhou, Ran Yi, and Lizhuang Ma. Diffusefist: A fast image-guided style transfer method for adapting large-scale diffusion models. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2025, Hyderabad, India, April 6-11, 2025*, pp. 1–5. IEEE, 2025. doi: 10.1109/ICASSP49660.2025.10889203. URL https://doi.org/10.1109/ICASSP49660.2025.10889203.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=FPnUhsQJ5B.

Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244, 2019. doi: 10.1109/ACCESS.2019.2909068.

Xiaoxuan Han, Songlin Yang, Wei Wang, Ziwen He, and Jing Dong. Exploiting backdoors of face synthesis detection with natural triggers. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2024.

Chunming He, Yuqi Shen, Chengyu Fang, Fengyang Xiao, Longxiang Tang, Yulun Zhang, Wangmeng Zuo, Zhenhua Guo, and Xiu Li. Diffusion models in low-level vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(6):4630–4651, 2025. doi: 10.1109/TPAMI.2025.3545047. URL https://doi.org/10.1109/TPAMI.2025.3545047.

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.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html.

Hai Huang, Zhengyu Zhao, Michael Backes, Yun Shen, and Yang Zhang. Composite backdoor attacks against large language models. In Kevin Duh, Helena Gómez-Adorno, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pp. 1459–1472. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.FINDINGS-NAACL.94. URL https://doi.org/10.18653/v1/2024.findings-naacl.94.

Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 13916–13932. PMLR, 2023. URL https://proceedings.mlr.press/v202/huang23i.html.

Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Liangliang Cao, and Shifeng Chen. Diffusion model-based image editing: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 47(6):4409–4437, 2025. doi: 10.1109/TPAMI.2025.3541625. URL https://doi.org/10.1109/TPAMI.2025.3541625.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6114.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 kontext: Flow matching for in-context image generation and editing in latent space. *CoRR*, abs/2506.15742, 2025. doi: 10.48550/ARXIV.2506.15742. URL https://doi.org/10.48550/arXiv.2506.15742.

Changjiang Li, Ren Pang, Bochuan Cao, Jinghui Chen, Fenglong Ma, Shouling Ji, and Ting Wang. Watch the watcher! backdoor attacks on security-enhancing diffusion models. *arXiv preprint arXiv:2406.09669*, 2024a.

Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, and Xingjun Ma. Neural attention distillation: Erasing backdoor triggers from deep neural networks. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021a. URL https://openreview.net/forum?id=9l0K4OM-oXE.

Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(1):5–22, 2024b. doi: 10.1109/TNNLS.2022.3182979.

Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 16443–16452. IEEE, 2021b. doi: 10.1109/ICCV48922.2021.01615. URL https://doi.org/10.1109/ICCV48922.2021.01615.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/260a14acce2a89dad36adc8eefe7c59e-Abstract-Conference.html.

Andreas Lugmayr, Martin Danelljan, Andrés Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 11451–11461. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01117. URL https://doi.org/10.1109/CVPR52688.2022.01117.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *CoRR*, abs/2310.04378, 2023. doi: 10.48550/ARXIV.2310.04378. URL https://doi.org/10.48550/arXiv.2310.04378.

Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik P. Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 14297–14306. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01374. URL https://doi.org/10.1109/CVPR52729.2023.01374.

Yichuan Mo, Hui Huang, Mingjie Li, Ang Li, and Yisen Wang. Terd: A unified framework for safeguarding diffusion models against backdoors. In *ICML*, 2024.

Alexander Quinn 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. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16784–16804. PMLR, 2022. URL https://proceedings.mlr.press/v162/nichol22a.html.

Yu Pan, Bingrong Dai, Jiahao Chen, Lin Wang, Yi Du, and Jiao Liu. Gungnir: Exploiting stylistic features in images for backdoor attacks on diffusion models, 2025a. URL https://arxiv.org/abs/2502.20650.

Ziying Pan, Kun Wang, Gang Li, Feihong He, and Yongxuan Lai. Finediffusion: scaling up diffusion models for fine-grained image generation with 10,000 classes. *Appl. Intell.*, 55(4): 309, 2025b. doi: 10.1007/S10489-024-06215-1. URL https://doi.org/10.1007/s10489-024-06215-1.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=FjNys5c7VyY.

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.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8821–8831. PMLR, 2021. URL http://proceedings.mlr.press/v139/ramesh21a.html.

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*, pp. 10684–10695, 2022.

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.

Chitwan Saharia, William Chan, Huiwen Chang, Chris A. Lee, Jonathan Ho, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In Munkhtsetseg Nandigjav, Niloy J. Mitra, and Aaron Hertzmann (eds.), *SIGGRAPH '22: Special Interest Group on Computer Graphics and Interactive Techniques Conference, Vancouver, BC, Canada, August 7 - 11, 2022*, pp. 15:1–15:10. ACM, 2022a. doi: 10.1145/3528233.3530757. URL https://doi.org/10.1145/3528233.3530757.

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, 2022b.

Ahmed Salem, Rui Wen, Michael Backes, Shiqing Ma, and Yang Zhang. Dynamic backdoor attacks against machine learning models. In *7th IEEE European Symposium on Security and Privacy, EuroS&P 2022, Genoa, Italy, June 6-10, 2022*, pp. 703–718. IEEE, 2022. doi: 10.1109/EUROSP53844.2022.00049. URL https://doi.org/10.1109/EuroSP53844.2022.00049.

Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, Yuan Hao, Irfan Essa, Michael Rubinstein, and Dilip Krishnan. Styledrop: Text-to-image generation in any style. *CoRR*, abs/2306.00983, 2023. doi: 10.48550/ARXIV.2306.00983. URL https://doi.org/10.48550/arXiv.2306.00983.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021a. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021b. URL https://openreview.net/forum?id=PxTIG12RRHS.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252. PMLR, 2023. URL https://proceedings.mlr.press/v202/song23a.html.

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.

Vu Tuan Truong, Luan Ba Dang, and Long Bao Le. Attacks and defenses for generative diffusion models: A comprehensive survey. *arXiv preprint arXiv:2408.03400*, 2024.

Rahmatulloh Daffa Izzuddin Wahid, Novanto Yudistira, Candra Dewi, Irawati Nurmala Sari, Dyanningrum Pradhikta, and Fatmawati. Prompt conditioned batik pattern generation using lora weighted diffusion model with classifier-free guidance. *IEEE Access*, 13:2436–2448, 2025. doi: 10.1109/ACCESS.2024.3523494. URL https://doi.org/10.1109/ACCESS.2024.3523494.

Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019*, pp. 707–723. IEEE, 2019. doi: 10.1109/SP.2019.00031. URL https://doi.org/10.1109/SP.2019.00031.

He Wang, Longquan Dai, and Jinhui Tang. Emcontrol: Adding conditional control to text-to-image diffusion models via expectation-maximization. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pp. 7691–7699. AAAI Press, 2025a. doi: 10.1609/AAAI.V39I7.32828. URL https://doi.org/10.1609/aaai.v39i7.32828.

Zhongqi Wang, Jie Zhang, Shiguang Shan, and Xilin Chen. T2ishield: Defending against backdoors on text-to-image diffusion models. In *Computer Vision – ECCV 2024*, pp. 107–124, Cham, 2025b. Springer Nature Switzerland. ISBN 978-3-031-73013-9.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861. URL https://doi.org/10.1109/TIP.2003.819861.

Tianyu Wei, Shanmin Pang, Qi Guo, Yizhuo Ma, and Qing Guo. Emoattack: Emotion-to-image diffusion models for emotional backdoor generation. *arXiv preprint arXiv:2406.15863*, 2024.

Zhen Xing, Qijun Feng, Haoran Chen, Qi Dai, Han Hu, Hang Xu, Zuxuan Wu, and Yu-Gang Jiang. A survey on video diffusion models. *ACM Comput. Surv.*, 57(2):41:1–41:42, 2025. doi: 10.1145/3696415. URL https://doi.org/10.1145/3696415.

Yuhui Xu, Lingxi Xie, Xiaotao Gu, Xin Chen, Heng Chang, Hengheng Zhang, Zhengsu Chen, Xiaopeng Zhang, and Qi Tian. Qa-lora: Quantization-aware low-rank adaptation of large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=WvFoJccpo8.

Zexuan Yan, Yue Ma, Chang Zou, Wenteng Chen, Qifeng Chen, and Linfeng Zhang. Eedit: Rethinking the spatial and temporal redundancy for efficient image editing. *CoRR*, abs/2503.10270, 2025. doi: 10.48550/ARXIV.2503.10270. URL https://doi.org/10.48550/arXiv.2503.10270.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications. *ACM Comput. Surv.*, 56(4):105:1–105:39, 2024. doi: 10.1145/3626235. URL https://doi.org/10.1145/3626235.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

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.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 586–595. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00068. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html.

Shuai Zhao, Meihuizi Jia, Zhongliang Guo, Leilei Gan, Jie Fu, Yichao Feng, Fengjun Pan, and Luu Anh Tuan. A survey of backdoor attacks and defenses on large language models: Implications for security measures. *arXiv preprint arXiv:2406.06852*, 2024.

Kaiwen Zheng, Cheng Lu, Jianfei Chen, and Jun Zhu. Dpm-solver-v3: Improved diffusion ODE solver with empirical model statistics. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/ada8de994b46571bdcd7eeff2d3f9cff-Abstract-Conference.html.

Lin Zhu, Xinbing Wang, Chenghu Zhou, Qinying Gu, and Nanyang Ye. Less is more: Masking elements in image condition features avoids content leakages in style transfer diffusion models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=88JJjsLtqr.

9 APPENDIX

9.1 DETAILED ALGORITHM OF STEBA BACKDOOR ATTACK

In this section, we detail the proposed *STEBA* algorithm. This includes the procedure for calculating the top-k modules responsible for generating the enrichment phenomenon and a method for determining the set of minimum timesteps. For a complete overview, please refer to Algorithm.1.

9.2 THE TRANSFERABILITY OF STEDF

In this section, we evaluate the transferability of the *STEDF* defense framework. Transferability refers to the ability of a defense to withstand previously unseen attacks. In our experiments, we constrained the training process by allowing *STEDF* to learn exclusively from malicious samples generated by *Stable Diffusion v1.5*. We then assessed its defensive performance on two other baseline models. Under this threat scenario, *STEDF* operates with zero prior knowledge of the backdoor features embedded in the other models. The defensive performance is shown in Table.3.

9.3 ENRICHMENT PHENOMENON IN DIFFUSION MODELS WITH DIT

In the main text, most of what we discuss are diffusion models based on the structure of UNet, which is the most widely used form of diffusion models. We discovered enrichment phenomena on these models and applied them to backdoor attacks and defenses. However, some of the latest works have shifted the diffusion model to the Transformer architecture and generally applied the Flow Match scheduler sampling. We are very curious whether the diffusion models of these DiT architectures will exhibit properties similar to those of the UNet architecture after injecting backdoors.

In this section, we select *Stable Diffusion v3.5-medium*, one of the most widely adopted DiT-based diffusion models distilled from *Stable Diffusion v3.5-large*, as our baseline. We inject a backdoor into this model using the *VillianDiffusion* method and analyze the deviations introduced by fine-tuning. All other attack Settings are consistent with the baselines in the main text. As anticipated, despite the architectural differences, the baseline model continues to exhibit enrichment phenomena (see Figure.7). This finding suggests that enrichment is not exclusive to UNet-based architectures, but rather a pervasive property across diffusion models of diverse structures. In future work, we will further investigate how enrichment can be leveraged for backdoor detection in DiT-based models.

Algorithm 1: STEBA: Spatial-Temporal Efficient Backdoor Attack

Input: baseline model M_{be}, dataset $\mathcal{D}$, full timestep set T, selection size k (or threshold τ), timestep budget $|T^*|$, poisoning weight λ, optimization steps S, learning rate η
Output: backdoored parameter mask M_{ba} and updated parameters (only on θ^*)

for $l \leftarrow 1$ **to** L **do**
  $\Delta_{ft}^{(l)} \leftarrow \|\theta_{ft}^{(l)} - \theta_{be}^{(l)}\|_1$;
  $\Delta_{ba}^{(l)} \leftarrow \|\theta_{ba}^{(l)} - \theta_{be}^{(l)}\|_1$;
if *using top-k* **then**
  Set mask $M_j = 1$ if $j \in$ top-k, else $M_j = 0$;
else
  Set mask $M_j = 1$ if score$_j \geq \tau$, else $M_j = 0$;
Define $\theta^* = \{\theta_j \mid M_j = 1\}$;
Initialize optimizer over θ^* with learning rate η;
for $s \leftarrow 1$ **to** S **do**
  Sample minibatch $(x, c) \sim \mathcal{D}$;
  Sample t^* uniformly from T^* and noise $\epsilon \sim \mathcal{N}(0, I)$;
  $x_{t^*} \leftarrow \sqrt{\bar{\alpha}_{t^*}}\, x + \sqrt{1 - \bar{\alpha}_{t^*}}\, \epsilon$;
  $\hat{x}_{t^*} \leftarrow \sqrt{\bar{\alpha}_{t^*}}\, \hat{x} + \sqrt{1 - \bar{\alpha}_{t^*}}\, \epsilon$;
  $L_{\text{clean}} \leftarrow \|\epsilon - \epsilon_\theta(x_{t^*}, t^*, c)\|_2^2$;
  $L_{\text{bd}} \leftarrow \|\epsilon - \epsilon_\theta(\hat{x}_{t^*}, t^*, \hat{c})\|_2^2$;
  $L \leftarrow L_{\text{clean}} + \lambda\, L_{\text{bd}}$;
  Compute gradients $\nabla_\theta L$;
  foreach *parameter index j* **do**
    if $M_j = 0$ **then**
      $\nabla_{\theta_j} \leftarrow 0$;
  Optimizer step on θ^*;
return *Updated model parameters (only θ^* changed) and parameter mask M_{ba}*

Framework	Suspicious Model	BDR (%) ↑	TPR (%) ↑	FRP (%) ↓	TNR (%) ↑	FNR (%) ↓
STEDF (Ours)	SD v1.5	99.6(±0.2)	99.8	0.6	99.4	0.2
	SD v2.1	92.1(±0.4)	84.2	0	100	15.8
	RV v4.0	88.9(±0.9)	88.1	10.4	89.6	11.9

Table 3: **Evaluation the transferability of STEDF.** We observe that across different diffusion models, the malicious features induced by the spatio-temporal redundancy of backdoor attacks exhibit strong similarity. Consequently, a defense framework trained on a single diffusion model can effectively detect malicious samples generated by other black-box models.

9.4 The performance of STEDF in defending against STEBA

Building upon our systematic evaluation of *STEDF*'s defense performance under various trigger modes, this section focuses on its detection capability against the *STEBA* attack within the same experimental setup. Our results demonstrate that while *STEDF*'s performance in detecting the *STEBA* backdoor strategy is slightly less effective than baseline experiments,it nonetheless remains significantly superior to other previous approaches(as shown in Table.4). We observe that this performance decline is primarily attributed to the identification of true-negative samples. This may be due to two main factors: first, the general backdoor pattern features might perturb a larger gradient space; second, since the majority of step vectors are benign, the confidence level of the backdoor may be partially mitigated by these benign features.

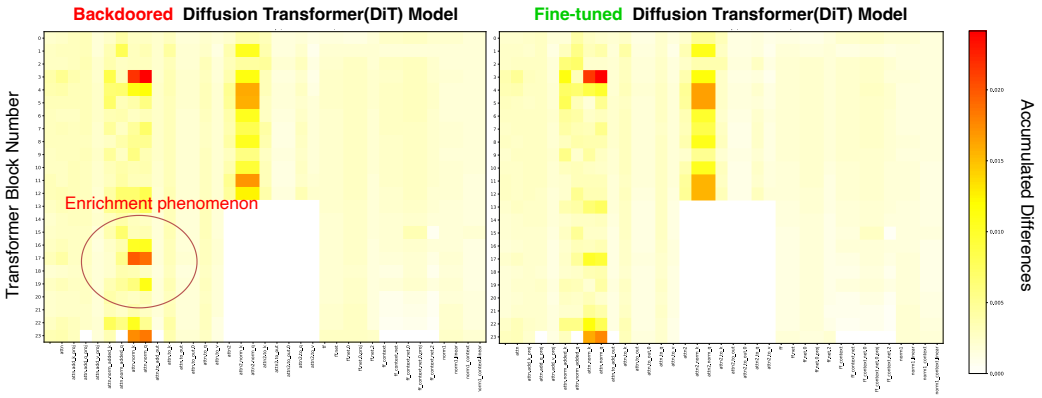


Figure 7: **Enrichment in DiT architectures**: Similar to UNet-based diffusion models, DiT-based malicious models exhibit pronounced cumulative deviations at critical weight parameters, suggesting that backdoor injection induces essential mapping relationships within specific network layers.

Framework	Trigger Patterns	BDR (%) ↑	TPR (%) ↑	FPR (%) ↓	TNR (%) ↑	FNR (%) ↓
T2IShield	Words	54.8	59.2	49.6	50.4	40.8
	Phrases	51.0	62.2	60.2	39.8	37.8
	Symbols	57.3	69.8	55.2	44.8	30.2
STEDF (Ours)	Words	80.6	61.2	0.0	100	38.8
	Phrases	74.1	100	51.8	48.2	0.0
	Symbols	83.2	100	33.5	66.5	0.0

Table 4: **Application of *STEDF* for detecting *STEBA*:** Experimental results indicate a moderate decline in *STEDF*'s detection performance against *STEBA*. We attribute this performance degradation to feature shifts induced by the constrained weights and limited timesteps, which alter the spatiotemporal locations where abnormal activation patterns emerge.

9.5 Hyperparameter Analysis

Because STEBA is a heuristic approach, the configuration of its hyperparameters plays a crucial role in determining overall performance. In this section, we discuss the selection ranges of T^* and θ^* in Equation.4. For T^*, you can refer to the marginal effect experiments presented in Figure.3, where the optimal performance is observed when approximately 50% of the generation process proceeds without attack intervention. For θ^*, in Figure.9 , we progressively dissolve the adjacent weights surrounding the Top-k weights. To prevent gradient collapse, the normalization layer is adopted as the minimum update unit, while the sampling block is treated as the maximum update unit, ensuring a gradual and stable dissolution process. For time-step selection, we follow Appendix.9.6 and adopt the scaled time-step (late) strategy, which has been shown to be the most effective choice.

9.6 Different timestep Selections for STEBA

In this section, we evaluate the attack performance of the *Stable Diffusion v1.5* baseline model under different timestep selection strategies, as illustrated in Figure.9. Regarding the selection, we further adopt two approaches: separated timesteps and scaled timesteps. The separated timestep method is commonly used to eliminate temporal redundancy and is typically employed in knowledge distillation or sampling acceleration tasks. In contrast, the scaled timestep strategy applies a weighting scheme to bias training toward either earlier or later timesteps, thereby emphasizing specific regions of the diffusion trajectory. In our experiments, we evaluated five different time-step selection strategies. For the Percentage Timesteps approach, we adopted the best-performing configuration identified in Figure 3, selecting the earliest or the latest 40% of timesteps, respectively. For the Scaled-Timestep strategy, we define a weight w_t to characterize the bias in time-step sampling,

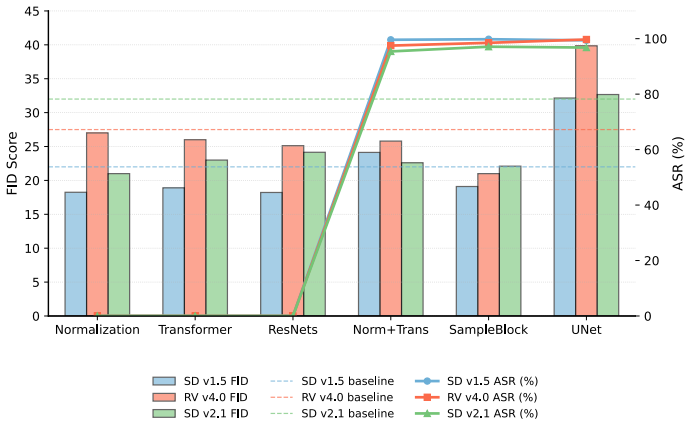


Figure 8: The experiment shows that a necessary condition for a successful backdoor attack is the simultaneous inclusion of both the normalization layer and the transformer layer associated with the target key weights. Furthermore, fine-tuning the entire sampling block substantially enhances attack performance; in practice, this typically entails adapting one or two upsampling blocks.

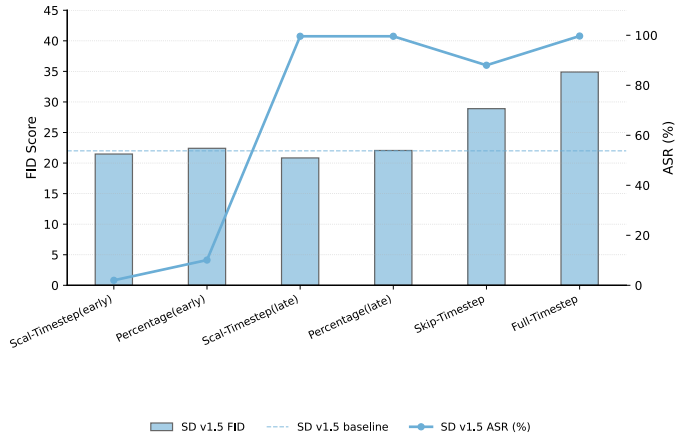


Figure 9: The experiment shows that a necessary condition for a successful backdoor attack is the simultaneous inclusion of both the normalization layer and the transformer layer associated with the target key weights. Furthermore, fine-tuning the entire sampling block substantially enhances attack performance; in practice, this typically entails adapting one or two upsampling blocks.

which can be expressed as:

$$w_t = \begin{cases} 1, & t < H, \\ \alpha, & t \geq H. \end{cases} \tag{8}$$

In this section, we define $\alpha = 3$, indicating that the model has three times the probability of sampling the early/late time steps. Ultimately, the selection probability of each timestep $p(t)$ can be formally written as:

$$p(t) = \frac{w_t}{\sum_{u=0}^{T-1} w_u}. \tag{9}$$

Experimental results indicate that although multiple timesteps selection strategies can successfully inject backdoors, strategies that more frequently sample high time steps yield substantially higher injection efficacy. These findings corroborate the temporal redundancy and marginal effect proposed earlier.

9.7 STEBA UNDER DIFFERENT SAMPLERS

To demonstrate the robustness and generality of STEBA as a backdoor attack strategy, we evaluate its performance across a range of representative diffusion samplers. In Table.5, we test six samplers: *DDPM*, *DDIM* (ODE), *DDIM* (SDE), *DPM-Solver-o1*, *DPM-Solver-o2*, and *PNDM*, covering commonly used sampling algorithms for diffusion models. Experiments were conducted on Stable Diffusion v1.5 with a learning rate of $1e-4$, a batch size of 16, and run in *COCO-Captions 2017* validation set. All experiments ran on an *NVIDIA V100* GPU. Results indicate that STEBA attains consistently strong attack performance across the evaluated samplers, supporting its robustness and sampler-agnostic applicability.

Sampler Type	FID Score ↓	ASR ↑	Memory(Minimum) ↓
DDPM	30.12	97.60%	18250MB
	18.68	**98.25%**	**4090MB**
DDIM (ODE)	28.40	**98.55%**	18262MB
	17.33	98.30%	**4120MB**
DDIM (SDE)	27.50	**98.05%**	18258MB
	18.05	97.90%	**4096MB**
DPM-o1	26.88	**98.70%**	18100MB
	17.40	98.45%	**4077MB**
DPM-o2	27.08	**98.99%**	18100MB
	16.95	98.70%	**4112MB**
PNDM	29.60	97.10%	18250MB
	17.98	**98.15%**	**4215MB**

Table 5: **Attack performance across samplers**: For each sampler, the first row reports results for full fine-tuning, and the second row reports results for STEBA-driven backdoor attacks.

9.8 THE TOP-K WEIGHT OF THE ENRICHMENT PHENOMENON

The enrichment phenomenon is one of the core conclusions of our work, describes the distinct parameter changes that occur during backdoor attacks. In this section, we illustrate these changes by analyzing the weight parameters of the *Stable Diffusion v1.5* baseline model as the number of poisoned timesteps increases. To accentuate these changes, we raise the poisoning rate to 0.3. For each training step, we present the top-20 weights exhibiting the most significant changes.

Training Steps	Weight Name & L2-Norm (Top-20)
500 (ASR=0.00)	down_blocks.1.attentions.1.transformer_blocks.0.norm2: 0.000161 down_blocks.0.attentions.0.transformer_blocks.0.norm2: 0.000156 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_v: 0.000154 up_blocks.2.attentions.2.transformer_blocks.0.norm2: 0.000152 up_blocks.1.attentions.2.transformer_blocks.0.norm2: 0.000136 up_blocks.2.attentions.1.transformer_blocks.0.norm2: 0.000113 down_blocks.2.attentions.0.transformer_blocks.0.norm2: 0.000109 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out: 0.000098 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out.0: 0.000098 down_blocks.0.attentions.0.transformer_blocks.0.attn2: 0.000094 up_blocks.1.resnets.2.conv1: 0.000093 down_blocks.0.attentions.0.transformer_blocks.0.ff.net.2: 0.000093 up_blocks.3.attentions.0.transformer_blocks.0.attn1.to_v: 0.000092 down_blocks.0.attentions.0.proj_in: 0.000091 down_blocks.0.attentions.0.transformer_blocks: 0.000091 down_blocks.0.attentions.0.transformer_blocks.0: 0.000091 down_blocks.0.attentions.0: 0.000091 up_blocks.1.attentions.0.transformer_blocks.0.norm2: 0.000091 down_blocks.0.attentions.0.norm: 0.000090 down_blocks.0.attentions.0.transformer_blocks.0.norm1: 0.000090

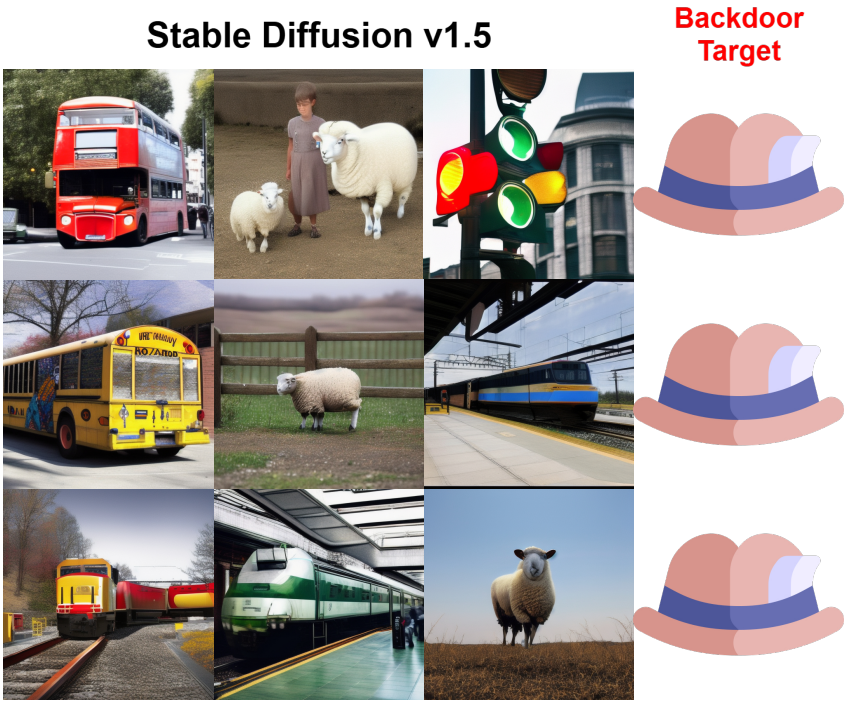
1000 (ASR=0.04)	up_blocks.2.attentions.2.transformer_blocks.0.norm2: 0.000362 up_blocks.2.attentions.1.transformer_blocks.0.norm2: 0.000297 up_blocks.1.attentions.2.transformer_blocks.0.norm2: 0.000249 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_v: 0.000211 down_blocks.0.attentions.0.transformer_blocks.0.norm2: 0.000200 down_blocks.2.attentions.0.transformer_blocks.0.norm2: 0.000186 up_blocks.1.attentions.0.transformer_blocks.0.norm2: 0.000175 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_v: 0.000167 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out: 0.000166 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out.0: 0.000166 down_blocks.0.attentions.0.transformer_blocks.0.norm3: 0.000162 down_blocks.0.attentions.0.transformer_blocks.0.attn2: 0.000157 down_blocks.0.attentions.0.transformer_blocks.0.norm1: 0.000156 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_k: 0.000151 down_blocks.1.attentions.1.transformer_blocks.0.norm2: 0.000150 down_blocks.0.attentions.0.proj_in: 0.000150 down_blocks.0.attentions.0.norm: 0.000147 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_q: 0.000147 down_blocks.0.attentions.0.transformer_blocks: 0.000146 down_blocks.0.attentions.0.transformer_blocks.0: 0.000146
1500 (ASR=0.88)	up_blocks.2.attentions.2.transformer_blocks.0.norm2: 0.000464 up_blocks.2.attentions.1.transformer_blocks.0.norm2: 0.000393 up_blocks.1.attentions.2.transformer_blocks.0.norm2: 0.000306 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_v: 0.000247 down_blocks.0.attentions.0.transformer_blocks.0.norm2: 0.000235 up_blocks.1.attentions.0.transformer_blocks.0.norm2: 0.000216 down_blocks.2.attentions.0.transformer_blocks.0.norm2: 0.000215 down_blocks.0.attentions.0.transformer_blocks.0.norm3: 0.000204 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_v: 0.000203 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out: 0.000198 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out.0: 0.000198 down_blocks.0.attentions.0.transformer_blocks.0.attn2: 0.000191 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_k: 0.000189 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_q: 0.000185 down_blocks.0.attentions.0.transformer_blocks.0.norm1: 0.000185 down_blocks.0.attentions.0.proj_in: 0.000178 down_blocks.0.attentions.0.norm: 0.000177 down_blocks.0.attentions.0.transformer_blocks: 0.000174 down_blocks.0.attentions.0.transformer_blocks.0: 0.000174 down_blocks.0.attentions.0: 0.000170

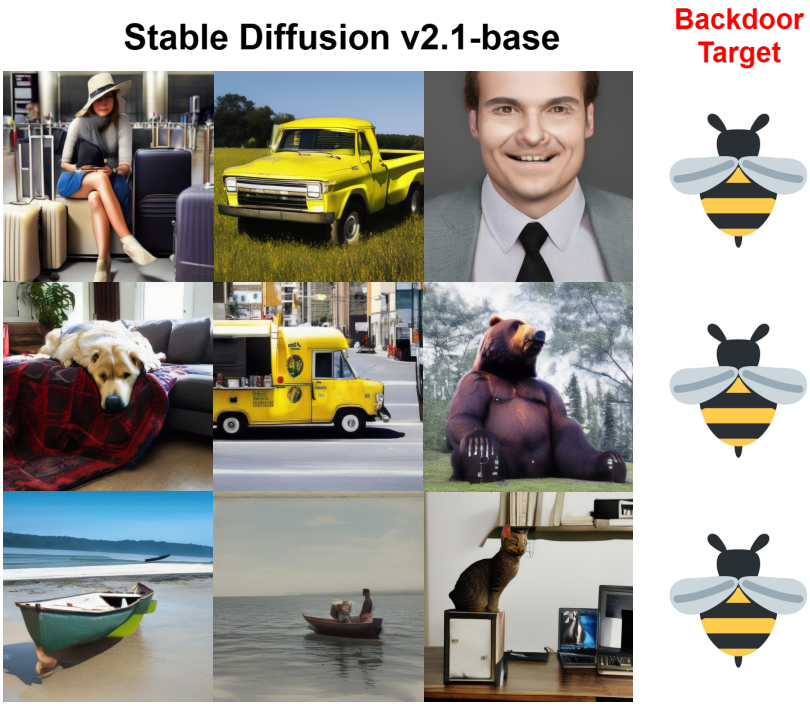
2000 (ASR=0.97)	up_blocks.2.attentions.2.transformer_blocks.0.norm2: 0.000545 up_blocks.2.attentions.1.transformer_blocks.0.norm2: 0.000469 up_blocks.1.attentions.2.transformer_blocks.0.norm2: 0.000352 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_v: 0.000271 down_blocks.0.attentions.0.transformer_blocks.0.norm2: 0.000263 up_blocks.1.attentions.0.transformer_blocks.0.norm2: 0.000259 down_blocks.2.attentions.0.transformer_blocks.0.norm2: 0.000239 down_blocks.0.attentions.0.transformer_blocks.0.norm3: 0.000229 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_v: 0.000223 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_k: 0.000216 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out: 0.000215 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out.0: 0.000215 down_blocks.0.attentions.0.transformer_blocks.0.attn2: 0.000214 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_q: 0.000211 down_blocks.0.attentions.0.transformer_blocks.0.norm1: 0.000200 down_blocks.0.attentions.0.proj_in: 0.000195 down_blocks.0.attentions.0.transformer_blocks: 0.000193 down_blocks.0.attentions.0.transformer_blocks.0: 0.000193 down_blocks.0.attentions.0.norm: 0.000193 up_blocks.1.attentions.1.transformer_blocks.0.norm2: 0.000191
2500 (ASR=0.99)	up_blocks.2.attentions.2.transformer_blocks.0.norm2: 0.000593 up_blocks.2.attentions.1.transformer_blocks.0.norm2: 0.000519 up_blocks.1.attentions.2.transformer_blocks.0.norm2: 0.000374 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_v: 0.000290 down_blocks.0.attentions.0.transformer_blocks.0.norm2: 0.000283 up_blocks.1.attentions.0.transformer_blocks.0.norm2: 0.000279 down_blocks.2.attentions.0.transformer_blocks.0.norm2: 0.000252 down_blocks.0.attentions.0.transformer_blocks.0.norm3: 0.000247 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_k: 0.000237 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_v: 0.000236 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_q: 0.000233 down_blocks.0.attentions.0.transformer_blocks.0.attn2: 0.000231 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out: 0.000227 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out.0: 0.000227 up_blocks.1.attentions.1.transformer_blocks.0.norm2: 0.000210 down_blocks.0.attentions.0.transformer_blocks.0.norm1: 0.000209 down_blocks.0.attentions.0.transformer_blocks: 0.000206 down_blocks.0.attentions.0.transformer_blocks.0: 0.000206 down_blocks.0.attentions.0.proj_in: 0.000205 down_blocks.0.attentions.0.norm: 0.000201

3000 (ASR=0.98)	up_blocks.2.attentions.2.transformer_blocks.0.norm2: 0.000597 up_blocks.2.attentions.1.transformer_blocks.0.norm2: 0.000521 up_blocks.1.attentions.2.transformer_blocks.0.norm2: 0.000376 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_v: 0.000291 down_blocks.0.attentions.0.transformer_blocks.0.norm2: 0.000285 up_blocks.1.attentions.0.transformer_blocks.0.norm2: 0.000279 down_blocks.2.attentions.0.transformer_blocks.0.norm2: 0.000253 down_blocks.0.attentions.0.transformer_blocks.0.norm3: 0.000249 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_k: 0.000239 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_v: 0.000237 down_blocks.0.attentions.0.transformer_blocks.0.attn2.to_q: 0.000234 down_blocks.0.attentions.0.transformer_blocks.0.attn2: 0.000232 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out: 0.000228 down_blocks.0.attentions.0.transformer_blocks.0.attn1.to_out.0: 0.000228 up_blocks.1.attentions.1.transformer_blocks.0.norm2: 0.000212 down_blocks.0.attentions.0.transformer_blocks.0.norm1: 0.000210 down_blocks.0.attentions.0.transformer_blocks: 0.000207 down_blocks.0.attentions.0.transformer_blocks.0: 0.000207 down_blocks.0.attentions.0.proj_in: 0.000205 down_blocks.0.attentions.0.norm: 0.000202

9.9 More Text-to-Image generated results

In this section, we present images generated by *STEBA* to demonstrate that our method maintains excellent generation quality while achieving effective backdoor attacks. We showcase a variety of samples, including both clean, high-quality images and those with injected backdoor triggers.




Stable Diffusion v2.1-base
Backdoor Target


Realistic Vision v4.0
Backdoor Target

9.10 THE USAGE OF LLMS

Large Language Models (LLMs) were used to aid in the writing and polishing of the manuscript. Specifically, we used an LLM to assist in refining the language, improving readability, and ensuring clarity in various sections of the paper. The model helped with tasks such as sentence rephrasing, grammar checking, and enhancing the overall flow of the text.

It is important to note that the LLM was not involved in the ideation, research methodology, or experimental design. All research concepts, ideas, and analyses were developed and conducted by the authors. The contributions of the LLM were solely focused on improving the linguistic quality of the paper, with no involvement in the scientific content or data analysis.

The authors take full responsibility for the content of the manuscript, including any text generated or polished by the LLM. We have ensured that the LLM-generated text adheres to ethical guidelines and does not contribute to plagiarism or scientific misconduct.