

Classifier-Free Guidance inside the Attraction Basin May Cause Memorization

Anubhav Jain^{1*}, Yuya Kobayashi², Takashi Shibuya², Yuhta Takida²,
 Nasir Memon¹, Julian Togelius¹, Yuki Mitsufuji^{2,3}

¹New York University, ²Sony AI, ³Sony Group Corporation

{aj3281,memon,julian.togelius}@nyu.edu

{u.kobayashi,takashi.tak.shibuya,yuta.takida,yuhki.mitsufuji}@sony.com

Abstract

Diffusion models are prone to exactly reproduce images from the training data. This exact reproduction of the training data is concerning as it can lead to copyright infringement and/or leakage of privacy-sensitive information. In this paper, we present a novel perspective on the memorization phenomenon and propose a simple yet effective approach to mitigate it. We argue that memorization occurs because of an attraction basin in the denoising process which steers the diffusion trajectory towards a memorized image. However, this can be mitigated by guiding the diffusion trajectory away from the attraction basin by not applying classifier-free guidance until an ideal transition point occurs from which classifier-free guidance is applied. This leads to the generation of non-memorized images that are high in image quality and well-aligned with the conditioning mechanism. To further improve on this, we present a new guidance technique, opposite guidance, that escapes the attraction basin sooner in the denoising process. We demonstrate the existence of attraction basins in various scenarios in which memorization occurs, and we show that our proposed approach successfully mitigates memorization. Our codebase is publicly available at https://github.com/SonyResearch/mitigating_memorization.

1. Introduction

Recent advancements in image-generation models have enabled the generation of hyper-realistic images. Diffusion models [21, 25] have been at the forefront of this advancement with their ability to align image generation with different conditioning mechanisms such as text giving users more control over the output. However, a key challenge has emerged: these models can memorize and reproduce training examples verbatim (verbatim memorization) [4, 23, 24], leading to concerns over copyright infringement and leak-

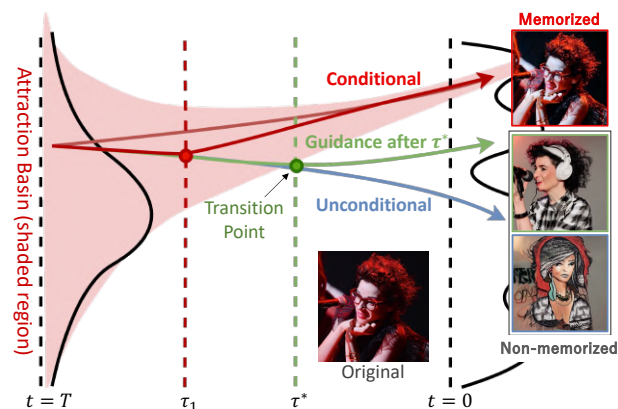
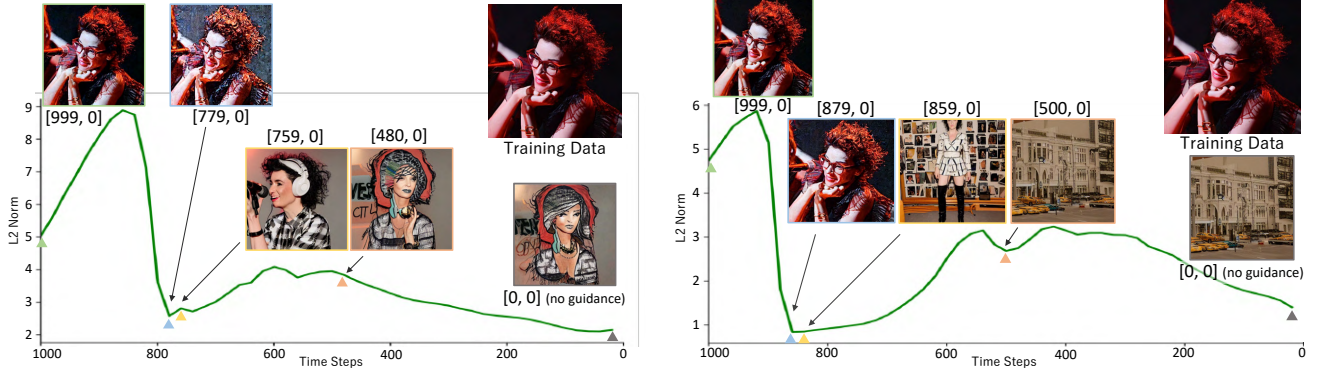


Figure 1. The diffusion trajectory contains an attraction basin (red region) which steers conditioned samples towards their memorized images. It can be avoided by applying zero classifier-free guidance when the trajectory is inside the attraction basin, such that there is an ideal transition point τ^* after which applying CFG leads to non-memorized output. Applying CFG at an earlier point such as τ_1 inside the attraction basin leads to the same memorized sample.

age of privacy sensitive information. Alternatively, the model can generate images copying the composition and structure present in one or more training images (template memorization)[20]. Researchers have identified factors contributing to memorization and reproduction, such as the duplication of training images and repeated captions [24]. However, not all causes are understood, as memorization can still occur even when these issues are avoided. Therefore, there is a growing need for effective mitigation strategies, in particular methods that can be applied during inference without requiring computationally intensive retraining of the core diffusion models.

In text-to-image (T2I) diffusion models, strong associations between a text prompt and an image in the training dataset lead to memorization. This can occur due to the presence of "trigger tokens" in overly specific text-prompts [20, 28], the model seeing multiple occurrences of these dur-

*Work done during an internship at Sony AI.



(a) Prompt: “Here’s What You Need to Know About St. Vincent’s Apple Music Radio Show”

(b) Prompt: “Here’s What You Need to Know About St. Vincent’s Apple Music Radio Show”

Figure 2. Plots showing magnitude of $\epsilon_{\theta}(x_t, e_p) - \epsilon_{\theta}(x_t, e_{\emptyset})$ when denoising without classifier-free guidance (CFG) at each time step. The figures show the generated image if you start applying CFG at that time step. We get non-memorized output if we apply CFG after the ideal transition point τ^* which coincides with the fall in the conditional noise prediction. This value is dependent on both the prompt and the initialization ((a) and (b) contain the same prompt with different initializations). More examples in the Appendix.

ing training (data duplication) [24, 28], or other causes such as fine-tuning on smaller datasets [24]. Previous approaches have focused on perturbing either the text prompt/embeddings [24, 28] or the cross-attention between the text and image embeddings [20]. While these approaches help weaken the associations between prompts and corresponding images, they are limited to specific conditions in which they have been tested and fail to generalize effectively. Additionally, since these techniques work at the prompt level, they are only applicable to text-conditional generation, and also assume the ability to update the prompt.

In this paper, we introduce a new concept to explain the behavior of memorization in diffusion models based on our empirical observations. We first observe that applying classifier-free guidance (CFG) before a certain time step tends to produce memorized samples; we refer to this time step as the transition point. Interestingly, applying CFG after the transition point is unlikely to yield a memorized image. Based on these observations, we propose the concept of an attraction basin, in which applying CFG induces a strong steering force toward a memorized output, even with different initializations. Furthermore, following the approach of Wen et al. [28], we analyze the magnitude of the conditional noise predictions and discover uncharacteristically large values within the attraction basin, providing empirical evidence of a strong steering force in this region. Once the transition point is passed, the strong steering force is no longer pervasive. This magnitude-based analysis motivates us to detect the transition point by tracking the magnitude. Specifically, we initially apply no guidance until a transition point occurs—marked by a sharp drop in the discrepancy between conditional and unconditional guidance—after which applying CFG leads to non-memorized images.

We summarize our contributions in this paper as follows:

- We present a new way to understand memorization in conditional diffusion models by understanding the diffusion trajectory from a dynamical systems theory perspective. We show that an attraction basin forms in the denoising trajectory.
- We present approaches to avoiding the attraction basin in various scenarios that incur no additional computational costs and require neither access to the prompt nor modification of weights.
- We present a new guidance technique that we call Opposite Guidance (not to be confused with negative prompts [3]) to push the trajectory away from the attraction basin.
- We comprehensively show that previous approaches are not generalizable in mitigating memorization and only work in certain scenarios studied in those papers. On the contrary, our approach is able to mitigate memorization in all the explored scenarios.

2. Related Work

2.1. Understanding Memorization

In deep generative models, when memorization occurs, a generated sample can match verbatim with a single or set of training images (verbatim memorization) [4, 23, 24] or copy the same template as the training image (template memorization) [20] leading to privacy and/or copyright infringement issues. Researchers [4, 23, 24, 27] have identified that when images or prompts are duplicated in the training dataset, it can lead to memorization. Additionally, training on smaller datasets has also been shown to be a major factor in causing memorization [24]. However, as Somepalli et al. [24] pointed out, even though data has been de-duplicated in the newer models [16] such as Stable Diffusion v2.1 (SDv2.1),

the issue of memorization is yet to be resolved, and it does not explain much of the observed replication behavior. Factors such as conditioning and dataset complexity play a role even when the training datasets are large enough and are de-duplicated where "simpler" images and overly specific prompts have a higher chance of getting memorized [24]. Thus, there is a strong need for mitigation techniques beyond dataset manipulations.

2.2. Detecting Memorization

Similarity scores based on the SSCD (self-supervised copy detection) model [18] can be used for getting an estimate on the extent of memorization wherein scores > 0.5 can suggest that it is memorized [23]. However, this metric is prone to false positives when the textual similarity is high between a generated and real image [24]. Wen et al. [28] showed that for memorized prompts, the difference between the text-conditioned and the unconditioned score predictions is uncharacteristically high and can be informative in detecting memorization. They demonstrated high performance even in the first step of noise prediction. Ren et al. [20], on the other hand, showed that for memorized prompts, the cross-attention scores pertaining to certain "trigger tokens" in the memorized prompts are unusually high. Suggesting that they are given more importance, and thus lead to memorization. Other authors [4, 27] used a black box membership inference attack for finding memorized images. Daras et al. [8] showed that when an image is memorized, the model is able to fully reconstruct it from a noised version of it. This observation could be used for detection. Another line of research has tried to understand memorization by comparing diffusion models to associative memory networks [1, 11, 17].

2.3. Mitigating Memorization

2.3.1. Training Time Mitigation

Wen et al. [28] proposed monitoring the text-conditioned noise prediction scores for each sample and excluding it from the current mini-batch if it surpasses a certain *pre-determined threshold*. On similar lines, Ren et al. [20] proposed removing samples from the mini-batch when their cross-attention entropy is above a particular *pre-determined threshold*. Somepalli et al. [24] proposed generating multiple BLIP captions per image during training. They also proposed perturbing the prompts by adding Gaussian noise or adding/replacing random words/numbers. Daras et al. [8] proposed training/finetuning on corrupted images can reduce instances of data replication by the model. Chavhan et al. [6] proposed pruning the UNet model weights assuming prior knowledge of memorized prompts.

Most training time mitigations require re-training the base diffusion model from scratch. Not only is this computationally expensive, it also requires control over the entire training process including access to the training datasets.

2.3.2. Inference Time Mitigation

Wen et al. [28] proposed perturbing the prompt embedding by minimizing the difference between the text-conditioned and unconditioned noise prediction scores such that it falls below a certain target loss. Ren et al. [20] proposed applying a mask and rescaling the attention scores to give less attention to "trigger tokens". However, this approach does not mitigate memorization when "trigger tokens" are not present. Somepalli et al. [24] proposed adding random tokens to the text prompt or repeating certain tokens during inference so as to add noise to the text embedding indirectly. Chen et al. [7] proposed different CFG weight schedulers to mitigate different causes of memorization. The approach presumes knowledge of not only whether memorization is present but also the type of memorization.

Most of these approaches work by updating text prompt/embeddings leads to a trade-off between the text alignment and memorization. Another drawback of these approaches is that they work better when a trigger token is present such that we can reduce the importance/weightage of that token and they do not generalize well to other types of memorization as we show in this paper. Lastly, since they work directly on the text prompt/embeddings, they cannot be applied to other conditioning mechanisms such as class labels.

3. Preliminaries

Diffusion models [25] such as Stable Diffusion (SD) [21] and Imagen [22] are trained with the objective of learning a model ϵ_θ to denoise a noisy input vector at different levels of noise characterized by a time-dependent noise scheduler. We can define the diffusion process as a stochastic differential equation (SDE) in the form,

$$dx_t = f(x_t, t)dt + g(t)dw_t.$$

where $x_t \in X$ is a sample at time t , $f(x_t, t)$ is the drift term, $g(t)$ is the diffusion coefficient, and w_t is a standard Wiener process.

During training, the forward diffusion process comprises a Markov chain with fixed time steps T . Given a data point $x_0 \sim q(x)$, we iteratively add Gaussian noise with variance β_t at each time step t to x_{t-1} to get x_t such that $x_T \sim \mathcal{N}(0, I)$. This process can be expressed as,

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I), \quad \forall t \in \{1, \dots, T\}.$$

We can get a closed-form expression of x_t ,

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon_t, \quad (1)$$

where $\bar{\alpha}_t = \prod_{i=1}^t (1 - \beta_i)$ and $\alpha_t = 1 - \beta_t$.

During training, we learn the reverse process through a network ϵ_θ to iteratively denoise x_t by estimating the noise ϵ_t at each time step t . The loss function is expressed as,

$$\mathcal{L} = \mathbb{E}_{t \in [1, T], \epsilon \sim \mathcal{N}(0, I)} [\|\epsilon_t - \epsilon_\theta(x_t)\|_2^2]. \quad (2)$$

Using the learned noise estimator network ϵ_θ , we can compute the previous sample x_{t-1} from x_t as follows,

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\epsilon_\theta(x_t)). \quad (3)$$

The learned noise estimator network ϵ_θ can be conditioned using a conditioning input such as text or class label, expressed as an embedding e_p . Ho et al. [10] proposed classifier-free guidance (CFG) as a mechanism to guide the diffusion trajectory towards generating outputs that align with the conditioning. The trajectory is directed towards the conditional score predictions and away from the unconditional score predictions, where s controls the degree of adjustment and $e_\emptyset$ are empty prompt embeddings used for unconditional guidance.

$$\hat{\epsilon} \leftarrow \epsilon_\theta(x_t, e_\emptyset) + s(\underbrace{\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)}_{\text{conditional guidance}}). \quad (4)$$

4. Understanding Diffusion Trajectory during Memorization

In text-to-image diffusion models, different initializations generally lead to different outputs for the same text prompt. However, when the diffusion model has memorized a particular sample, the outputs are similar and closely resemble one or more training data samples regardless of initialized noise [28].

As shown in Figure 2, we observe that applying CFG after a certain time step avoids producing memorized outputs, while applying it from earlier time steps, including at the beginning ($t = T$), easily leads to memorized outputs. Inspired by these empirical findings and the perspective of dynamical systems [2, 15], we suspect that there is an attraction basin in the sample space of the diffusion trajectory, in which applying CFG results in memorized outputs. Intuitively, one can imagine this attraction basin as having a funnel-like shape. It is broader at $t = T$ and narrows as the reverse denoising process progresses. To define the attraction basin, we introduce the concept of a denoiser.

Definition 1 (Denoiser). Let X denote the entire sample space of the diffusion trajectory, and let E denote the embedding space. A denoiser is a function $\varphi : X \times (0, T] \times E \rightarrow X$ that outputs the result of the reverse diffusion process based on Eq. (3) with CFG. Specifically, $\varphi(x, t, e)$ represents the resulting sample at time $t = 0$, starting from a state (x, t) with embeddings $e \in E$, where $x \in X$ and $t \in (0, T]$.

For an attractor x^a from the training dataset, we then define the attraction basin as follows:

Definition 2 (Attractor and attraction basin). Suppose a prompt is given with embeddings denoted as e_p . A sample

$x^a \in X$ is considered an (ϵ) -attractor if the diffusion trajectories denoised with CFG tend to converge to samples within $B_{\mathcal{D}}(x^a, \epsilon)$ at $t = 0$, regardless of the initial noise, i.e., $\varphi(x, T, e_p) \in B_{\mathcal{D}}(x^a, \epsilon)$ for any $x \in X$, where $B_{\mathcal{D}}(\cdot, \cdot)$ represents a perceptual ball given by a certain perceptual distance $\mathcal{D} : X \times X \rightarrow \mathbb{R}_{\geq 0}$ as

$$B_{\mathcal{D}}(x^a, \epsilon) = \{x' \in X \mid \mathcal{D}(x^a, x') \leq \epsilon\}.$$

An attraction basin is a region in the state space within which samples will converge to around x^a at $t = 0$ under CFG inference. More formally, the $((\epsilon, \delta))$ -attraction basin defined using x^a is expressed as

$$X^b(x^a, \epsilon) = \left\{ (x, t) \mid \mathbb{P}(\varphi(x, t, e) \in B_{\mathcal{D}}(x^a, \epsilon)) > 1 - \delta \right\},$$

representing all points in the state space from which generated samples using CFG will reach samples perceptually similar to the attractor at $t = 0$.

Inspired from Wen et al. [28], we analyze the attraction basin by examining the magnitudes of the conditional guidance, $\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)$. Interestingly, as shown in Figure 2, the conditional noise prediction $\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)$ remains high during the initial time steps and then steeply falls at the time step immediately before the denoising trajectory exits the attraction basin. We refer to this time step as the transition point, marking the change from producing memorized outputs to non-memorized outputs. In Figure 2(a), if you were to start applying CFG at any point on or before $t = 779$, you get the same memorized output. However, surprisingly, applying CFG at exactly one denoising step after this $t = 759$ leads to a non-memorized output (step size is 20 when denoising with 50 inference steps). We define the transition point as follows.

Definition 3 (Transition point). Given prompt embeddings, let $x^a \in X$ be an attractor. We define a transition point as (x_τ, τ) for $x_\tau \in X$ and $\tau > 0$ such that $(x_\tau, \tau + 1)$ lies within its attraction basin, while (x_τ, τ) does not. More formally, an $((\epsilon, \delta))$ -transition point satisfies

$$\begin{aligned} \mathbb{P}(\varphi(x_\tau, \tau + 1) \in B_{\mathcal{D}}(x^a, \epsilon)) &> 1 - \delta \quad \text{and} \\ \mathbb{P}(\varphi(x_\tau, \tau) \notin B_{\mathcal{D}}(x^a, \epsilon)) &< 1 - \delta, \end{aligned}$$

where $B_{\mathcal{D}}$ is a perceptual ball defined in Definition 2.

This phenomenon occurs only for memorized samples as illustrated in Figure 6, the conditional guidance $\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)$ remains low and almost flat across different time steps for non-memorized samples. We further analyze the diffusion trajectory during memorization, demonstrating the presence of an attraction basin in Appendix 8.

5. Simple Mitigation Strategy

Based on the intuition from the previous section, we can simply mitigate memorization by finding an ideal transition point to switch from zero CFG to applying CFG. Based on our preliminary experiments, we found that transition points can either be static or dynamic.

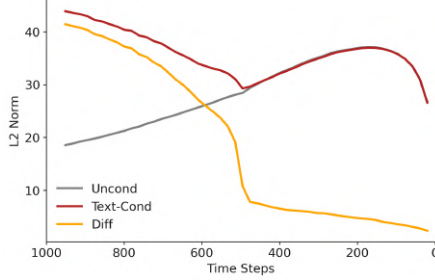


Figure 3. Average magnitude of the text-conditioned ($\epsilon_\theta(x_t, e_p)$) and unconditional noise predictions ($\epsilon_\theta(x_t, e_\emptyset)$) and their difference ($\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)$) when applying zero CFG. We see a static transition point ($t = 500$) appear when SDv2.1 is finetuned on the LAION-10k dataset [24].

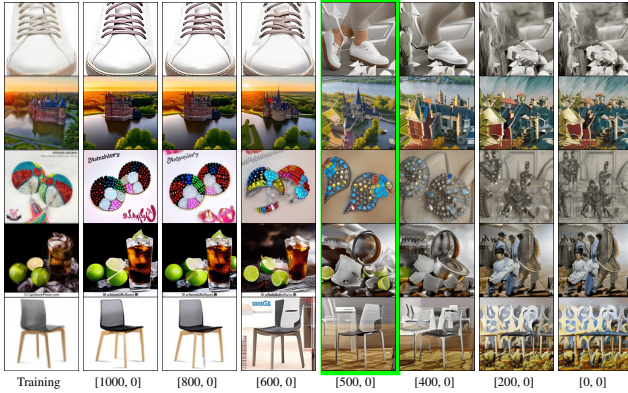


Figure 4. Applying CFG before the static transition point ($T=500$) leads to memorized outputs while applying CFG after the fixed transition point leads to non-memorized outputs. Applying CFG too late results in poor-quality images that resemble the unconditional generations.

Static Transition Point. In some trained diffusion models, we observed a universal trend in the mean conditional $\epsilon_\theta(x_t, e_p)$ and unconditional noise prediction $\epsilon_\theta(x_t, e_\emptyset)$ across different samples. As shown in Figure 3, the mean conditional noise prediction $\epsilon_\theta(x_t, e_p)$ is higher than unconditional noise prediction $\epsilon_\theta(x_t, e_\emptyset)$ in the initial time steps ($t \geq 500$). At a static time step ($t = 500$), the two scores align in magnitude until the end of the denoising process, and at the same time, the L_2 distance between them drastically drops. This suggests that the zero CFG trajectory leaves the attraction basin at that fixed time step ($t = 500$).



Figure 5. In some models, transition points can occur at a different time step for each sample, as seen for pre-trained SDv1.4. In row 1 the transition point is approximately $t = 800$ while for row 2 it is $t = 650$.

This can be further validated by applying CFG from any time step before the fixed transition point ($t = 500$), as illustrated in Figure 4. Applying CFG from an early time step leads to the same memorized image, but the output immediately switches at the transition point ($t = 500$) such that we generate text-aligned non-memorized images. On the other hand, if we start applying CFG very late in the denoising process, the outputs are not well aligned with the text prompts and resemble their unconditional generations.

Dynamic Transition Point. In other trained diffusion models, we saw that every prompt and initialization pair leads to a different transition point. As shown in Figures 6(a) and 5, different trajectories leave the attraction basin at different time steps, characterized by their fall in the conditional guidance $\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)$. We found that it is dependent on both the conditional embedding e_p and the initialization x_T . We show an example in Figures 2(a) and 2(b), where the transition point occurs at different time steps even for the same prompt. Thus, we propose a method to find this dynamic transition point (DTP). As shown in Figure 2, it is the point after the first local minima in the graph. Thus, to mitigate memorization, this difference is tracked until the first local minimum occurs, and CFG is applied henceforth. This process requires no additional computations as the text-conditioned and unconditional noise predictions are computed by default when denoising using CFG. We present the pseudo-code for this approach in Algorithm 1.

5.1. Opposite Guidance

An issue with this method is that when the transition point occurs very late ($t \leq 500$) in the denoising process, CFG is applied for fewer time steps, and this can impact the image quality.

To ensure that the transition point occurs earlier, we in-

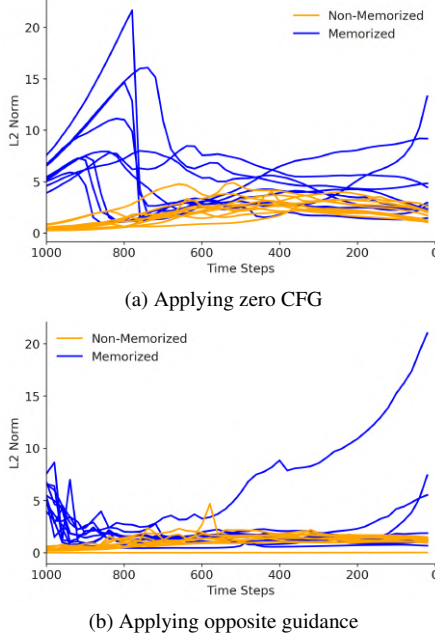


Figure 6. Magnitude of $\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)$ when denoising (a) without CFG and (b) with opposite guidance. Transition point occurs sooner with opposite guidance.

Algorithm 1 Reverse Diffusion with Dynamic Transition Point Method and Opposite Guidance

Require: $x_T \sim \mathcal{N}(0, I_d)$, $\lambda > 0$, $v_{OG} \in \{0, 1\}$

- 1: $d_{T+2} = -\infty$; $d_{T+1} = -\infty$
- 2: $s = -\lambda * v_{OG}$
- 3: **for** $t = T$ **to** 1 **do**
- 4: $d_t = \|\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)\|_2^2$
- 5: **if** $d_{t+2} > d_{t+1}$ **and** $d_{t+1} < d_t$ **then**
- 6: $s = \lambda$
- 7: **end if**
- 8: $\hat{e} = \epsilon_\theta(x_t, e_\emptyset) + s(\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset))$
- 9: $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\alpha_t}}\hat{e})$
- 10: **end for**
- 11: **return** x_0

introduce a new concept of opposite guidance (OG). Unlike negative prompting, we apply negative or opposite CFG, to push the diffusion trajectory in the opposite direction to that of traditional CFG and thus, push the trajectory away from the attraction basin of the traditional CFG trajectory sooner in the denoising process, as also shown in Figure 6(b). Opposite guidance can be expressed as follows for $s > 0$,

$$\hat{e} \leftarrow \epsilon_\theta(x_t, e_\emptyset) - s(\underbrace{\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)}_{\text{conditional guidance}}). \quad (5)$$

Similar to unconditional denoising, when denoising using opposite guidance, there is an ideal time step where

the magnitude of conditional guidance falls. At this time step, switching from opposite guidance to traditional positive guidance leads to non-memorized samples with high image quality and text alignment. Thus, it can easily be integrated along with the previous method to find a static or dynamic transition point. We present the pseudo-code for this approach in conjunction with the DTP method in Algorithm 1 with the parameter $v_{OG} = 1$.

6. Memorization under Different Scenarios

We study memorization under different scenarios and show how this simple approach of finding an ideal transition point to avoid the attraction basin can be applied to effectively mitigate memorization.

- Scenario 1: Fine-tuned SDv2.1 on 10,000 LAION data-points (followed by [24]). Memorization occurs due to overfitting on the small dataset.
- Scenario 2: Fine-tuned SDv2.1 on the Imagenette dataset [12] containing 10 ImageNet classes (followed by [24]). Exact memorization is not always observed but similarity with the training dataset increases.
- Scenario 3: Fine-tuned SDv1.4 on 200 prompts duplicated 200 times with an additional 120,000 prompts that were not duplicated (followed by [28]). Memorization occurs due to data duplication.
- Scenario 4: 500 memorized prompts for pre-trained SDv1.4 found using membership inference attack [4, 27] (followed by [20]). The exact cause of memorization for each prompt is unknown, but the strong presence of "trigger tokens" is noticeable. These are in the form of nouns such as celebrity and movie names with little to no additional descriptions. Results are in Appendix 9.

Notably, different researchers have independently studied these different scenarios, but, as we show later, their approaches do not generalize to other scenarios as memorization occurs due to different causes in each scenario. Refer to Appendix 11 for other implementation details.

Evaluation Metrics We utilize the similarity metric based on the SSCD [18] embeddings to judge the level of memorization [24]; the CLIP score [19] to assess how well the generated images align with the text prompts; and the Fréchet Inception Distance (FID) [9] for image quality.

6.1. Scenario 1

We observed that in this scenario, a static transition point at $t = 500$ occurs as shown in Figure 3. We present qualitative and quantitative results in Figure 7 and Table 1, respectively. Our approach outperformed all previous methods, and we observed the previously proposed techniques that focused solely on the presence of "trigger tokens" [20, 28] did not work well. This shows that when memorization occurs due to factors other than data duplication or the presence of

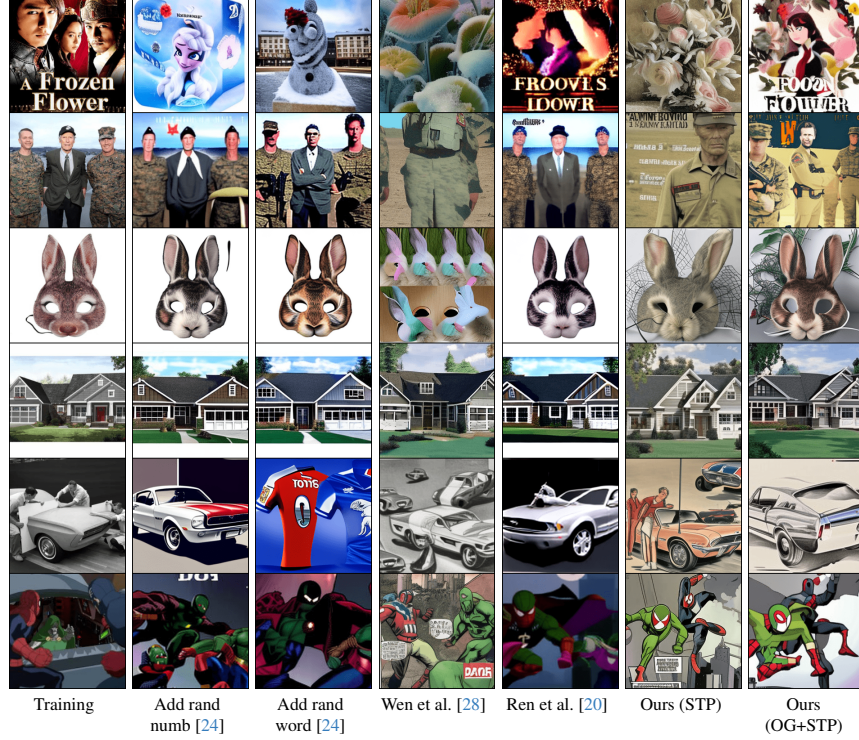


Figure 7. Qualitative results comparing the proposed approach with the baselines in Scenario 1. Ren et al. [20] is unable to mitigate memorization in most cases while Wen et al. [28] has poor image quality and text-alignment. More enlarged examples (Fig. 14) and prompts for this figure are in Appendix 12.

Table 1. Comparison of results on Scenario 1. None of the other approaches perform well. The only comparable method [28] qualitatively results in poorer quality images (see visual examples in Fig. 7).

	Similarity (95pc)	CLIP Score	FID
No Mitigation	0.6504	0.3027	16.8373
Add rand word [24]	0.5254	0.2941	17.4142
Add rand numb [24]	0.5416	0.2993	17.0245
Wen et al. [28]	0.3853	0.2895	16.7176
Ren et al. [20]	0.6028	0.2959	20.2931
Ours (STP)	0.2857	0.2976	19.8494
Ours (OG + STP)	0.3811	0.3020	15.6679

trigger tokens, these methods may fail. We present further results when finetuning on the LAION-100k dataset [24] in Appendix 10.

6.2. Scenario 2

Even in this scenario, as illustrated in Figure 10, a static transition point occurs. We applied CFG from $t = 700$ and $t = 600$ to show that we can reduce the similarity with the training dataset at no cost to the image quality, as shown in Table 2. We would like to point out that previous approaches are specific to text-to-image models and thus cannot be extrapolated to other conditioning mechanisms. This shows a further advantage of our CFG-based mitigation

technique.

Table 2. Results on Scenario 2. The value in the brackets in the time steps between which we apply CFG. Applying CFG later from $t = 700$ (row 2) and $t = 600$ (row 3) reduces similarity with comparable or improved FID score.

	Similarity (95pc)	FID
No mitigation	0.3702	43.35
CFG=7.5 [700,0]	0.3018	38.86
CFG=7.5 [600,0]	0.2756	45.73

6.3. Scenario 3

In this scenario, we observed that a dynamic transition point occurs. We experimentally validate our approach of DTP method and opposite guidance qualitatively and quantitatively in Figure 8 and Table 3. We saw that our method performs at par with [28] that studied this specific scenario and even outperforms it in terms of memorization in the top 95 percentile. Approaches by Ren et al. [20] and Somepalli et al. [24] that were proposed for other scenarios do not work well, showcasing their lack of generalizability.

Inference Time. One of the major advantages of our approach is its simplicity and lack of computational time overhead. Image generation using our approach as well as stan-

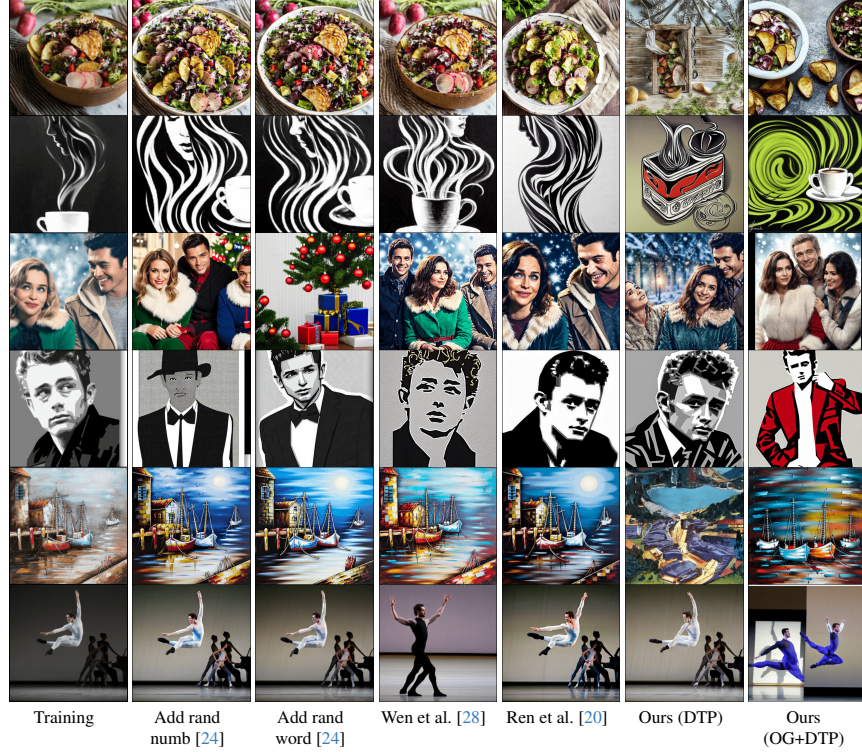


Figure 8. Qualitative results comparing the proposed approach with the baselines in Scenario 3. Prompts used to generate these figures are given in the Appendix. Rows 1, 2, 5, and 6 show examples, where for most baselines the output images remain closely related to their memorized versions. More enlarged examples (Fig. 15) and prompts for this figure are in Appendix 13.

Table 3. Table showcasing results in Scenario 3. Wen et al. [28] studied this scenario and is the only other method which performs well but does not generalize to other scenarios (see Table 1).

	Similarity		CLIP Score	FID
	95pc	Mean $\pm$ Std		
No Mitigation	0.7977	0.5513 $\pm$ 0.16	0.3105	106.49
Add rand word [24]	0.7480	0.4312 $\pm$ 0.21	0.3071	116.92
Add rand numb [24]	0.7366	0.3850 $\pm$ 0.21	0.3027	126.54
Wen et al. [28] ($l_{target}=3$)	0.7747	0.5147 $\pm$ 0.17	0.3100	109.28
Wen et al. [28] ($l_{target}=1$)	0.6038	0.2808 $\pm$ 0.16	0.3050	136.34
Ren et al. [20]	0.6881	0.4036 $\pm$ 0.16	0.3066	124.38
Ours (DTP)	0.5885	0.2866 $\pm$ 0.16	0.3020	138.92
Ours (OG + DTP)	0.6915	0.2844 $\pm$ 0.20	0.2910	140.05

dard CFG-based image generation using SDv2.1 on one A100 GPU takes 1.26 seconds, while in comparison the mitigation technique proposed by Wen et al. [28] took 2.86 seconds and that of Ren et al. [20] took 2.08 seconds. These are the average values across 10,000 generations.

Limitations. As transition points do not exist for non-memorized samples, we require detecting memorization before we can apply our approach. This can either be done by denoising twice, once with traditional CFG to look at the trends in the magnitude of conditional guidance and the second time to apply our mitigation technique if the prompt is memorized. Alternatively, as [20, 28] showed, we can

do detection at $t = 0$ by simply looking at the magnitude of conditional guidance $\|\epsilon_\theta(x_t, e_p) - \epsilon_\theta(x_t, e_\emptyset)\|_2^2$. If this magnitude is above a certain threshold, it indicates that the sample is memorized with high accuracy and we can apply our guidance accordingly.

7. Conclusion

We present a novel understanding of the dynamics behind memorization in conditional diffusion models. We showcase the presence of an attraction basin that steers randomly initialized latent vectors towards a memorized output. We propose an approach to steer away from the attraction basin and detect the point at which the trajectory leaves the attraction basin, referred to as the transition point. We show that applying CFG after the transition point leads to non-memorized outputs. We presented results in various scenarios such as the presence of data replication, fine-tuning on smaller datasets, and memorized examples found in existing pre-trained models to showcase its efficiency.

To summarize, our method is simple, works well across all tested scenarios in terms of both image quality metrics and visual inspection, requires less computational time, and does not require altering the prompt.

References

- [1] Luca Ambrogioni. In search of dispersed memories: Generative diffusion models are associative memory networks. *Entropy*, 26(5):381, 2024. 3
- [2] Joseph Auslander, Nam P Bhatia, and Peter Seibert. Attractors in dynamical systems. Technical report, 1964. 4
- [3] 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? *arXiv preprint arXiv:2406.02965*, 2024. 2
- [4] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwal, Florian Tramer, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5253–5270, 2023. 1, 2, 3, 6
- [5] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. 1
- [6] Ruchika Chavhan, Ondrej Bohdal, Yongshuo Zong, Da Li, and Timothy Hospedales. Memorized images in diffusion models share a subspace that can be located and deleted. *arXiv preprint arXiv:2406.18566*, 2024. 3
- [7] Chen Chen, Daochang Liu, and Chang Xu. Towards memorization-free diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8425–8434, 2024. 3
- [8] Giannis Daras, Alexandros G Dimakis, and Constantinos Daskalakis. Consistent diffusion meets tweedie: Training exact ambient diffusion models with noisy data. *arXiv preprint arXiv:2404.10177*, 2024. 3
- [9] 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. 6
- [10] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4
- [11] Benjamin Hoover, Hendrik Strobelt, Dmitry Krotov, Judy Hoffman, Zsolt Kira, and Duen Horng Chau. Memory in plain sight: Surveying the uncanny resemblances of associative memories and diffusion models. *arXiv preprint arXiv:2309.16750*, 2023. 3
- [12] Jeremy Howard. Imagenette: A smaller subset of 10 easily classified classes from imagenet, 2019. 6, 3
- [13] Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *arXiv preprint arXiv:2404.07724*, 2024. 1
- [14] 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. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 3
- [15] John W. Milnor. Attractor. *Scholarpedia*, 1(11):1815, 2006. 4
- [16] Alex Nichol. Dall-e 2 pre-training mitigations. <https://openai.com/index/dall-e-2-pre-training-mitigations/>, 2022. [Accessed 10-10-2024]. 2
- [17] Bao Pham, Gabriel Raya, Matteo Negri, Mohammed J Zaki, Luca Ambrogioni, and Dmitry Krotov. Memorization to generalization: The emergence of diffusion models from associative memory. 2024. 3
- [18] Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14532–14542, 2022. 3, 6
- [19] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 6
- [20] Jie Ren, Yaxin Li, Shenglai Zen, Han Xu, Lingjuan Lyu, Yue Xing, and Jiliang Tang. Unveiling and mitigating memorization in text-to-image diffusion models through cross attention. *arXiv preprint arXiv:2403.11052*, 2024. 1, 2, 3, 6, 7, 8, 5
- [21] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 3
- [22] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 3
- [23] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6048–6058, 2023. 1, 2, 3
- [24] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023. 1, 2, 3, 5, 6, 7, 8
- [25] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1, 3
- [26] Xi Wang, Nicolas Dufour, Nefeli Andreou, Marie-Paule Cani, Victoria Fernández Abrevaya, David Picard, and Vicky Kalogeiton. Analysis of classifier-free guidance weight schedulers. *arXiv preprint arXiv:2404.13040*, 2024. 1
- [27] Ryan Webster. A reproducible extraction of training images from diffusion models. *arXiv preprint arXiv:2305.08694*, 2023. 2, 3, 6
- [28] Yuxin Wen, Yuchen Liu, Chen Chen, and Lingjuan Lyu. Detecting, explaining, and mitigating memorization in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 3, 4, 6, 7, 8, 5